TELECOM INFRA PROJECT

MUST IP – SDN Controller SBI / Router NBI Technical Requirements

Version 1.1

Date: December 2021

Document version: v1.1

Confidentiality Level: Green [Public Access]

Authors

Oscar González de Dios

IP Transport and SDN Expert, Telefonica CTIO.

oscar.gonzalezdedios@telefonica.com

Luis-Ángel Muñoz

IP & SDN Network Architect, Vodafone

luis-angel.munoz@vodafone.com

Maria Vázquez

IP & SDN Network Architect, Vodafone

maria.vazquez@vodafone.com

Ndifor Luc-Fabrice Ngwa

Senior Engineer, Fixed Networks and Technology Management, MTN.

luc-fabrice.ndifor@mtn.com

Lloyd Mphahlele

General Manager, Transport, Transport, MTN.

Lloyd.Mphahlele@mtn.com

Renzo Díaz

Network Architect, Transport and Fixed Access, Telia Company

renzo.z.diaz@teliacompany.com

Philippe Niger

Network architect at Orange Labs (the R&D branch of Orange)

philippe.niger@orange.com

Esther Lerouzic

Orange

esther.lerouzic@orange.com

Samier Barguil

IP Transport and SDN Expert, Telefonica CTIO

samier.barguilgiraldo.ext@telefonica.com

Carlos Rodríguez

IP Transport and SDN Expert, Telefonica CTIO

carlos.rodriguezcampos.ext@telefonica.com

Version Control

Date	Revision	Author(s)	Comment
23/02/2021	1.0	Oscar Gonzalez de Dios Esther Lerouzic Ndifor Luc-Fabrice Ngwa Lloyd Mphahlele Philippe Niger Samier Barguil Carlos Rodriguez	First MUST IP Deliverable
01/12/2021	1.1	All Authors	Third MUST IP Deliverable Changes: - Renaming of old section 5, 6, 7, - Section 4 – Feature Bundles - Section 5 – UC 1.2, 1.3, 1.4, 1.5 - Section 6 – Category: L2VPN Services Provisioning - Section 10 – Category Network Creation - Section 11- Category support to performance monitoring - Section 12 – Category Telemetry - Section 13 – Category Support to Fault Management

Table of Contents

List of Figures

List of Tables

Scope

Implementing a complete standard specification is a time-consuming process. To accelerate the adoption of the specifications, the TIP MUST OOPT subgroup has compiled the needs of multiple operators, selecting the most relevant common use cases …

1 Scope

Implementing a complete standard technical requirement document is a time-consuming process. To accelerate the adoption of these requirements, the TIP MUST OOPT subgroup has **compiled the needs of multiple operators**, selecting the **most relevant common use cases** [1]. Based on this information, technical requirements are prepared and shared with the industry in an open manner. The produced technical requirement documents (TRDs and detail-TRDs) will serve as the baseline to qualify products and solutions (e.g., SDN Controllers) as compliant products to TIP MUST requirements.

This document provides the requirements of the SDN Interfaces mandatory to be exposed by IP/MPLS Network Elements (routers) and consumed by the South bound of IP SDN Domain Controllers. In this document these set of interfaces is shortened as SBI. The IP/MPLS equipment must support, in addition to the data and control plane protocols:

- NETCONF (IETF RFC 6241) [2] transport protocol, supporting the set of Yang data models defined in this document.
- Path Computation Element Protocol - PCEP (IETF RFC 5440) [3] to interact with a stateful PCE function with the minimum support defined in the document.
- BGP-LS (IETF RFC 7752) [4] to export topology and TE with the minimum support defined in the document.
- gRPC and gNMI [5] for configuration, state monitoring and streaming telemetry.

1.1 Use Cases

MUST follows an **incremental approach based on use cases**. For each use case, the needs in terms of interfaces are described in order to facilitate the implementations by Equipment and Controller vendors. The use cases in MUST are classified into several categories, as depicted in the next figure (see MUST Open Transport SDN Architecture Whitepaper[1]):

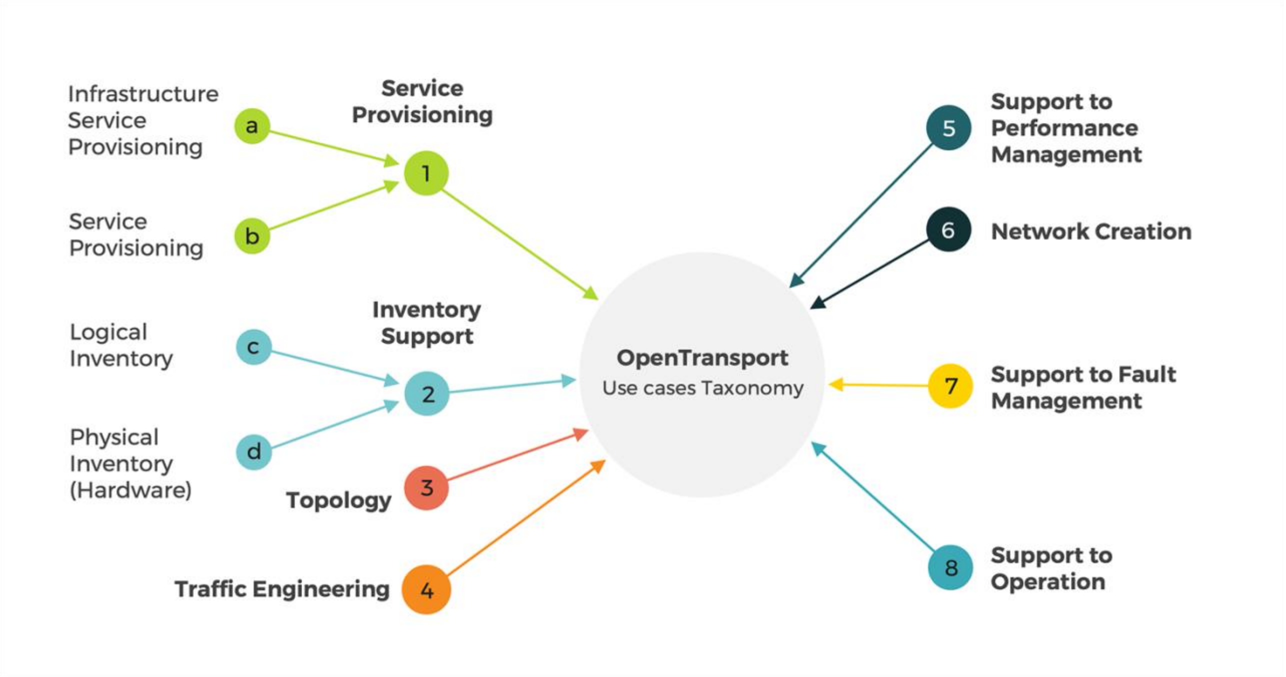


Figure 1. Uses Cases Categories

The purpose of this document is to identify, describe and compile the set of use cases as well as define the requirements, flows and details of each use cases in Network Element (router) and IP Domain controller. MUST Deliverables are incremental in terms of specified use cases.

Within MUST, all member operators have agreed on a first prioritization about the different use cases. The result is the selected use cases included in the following table and covered in D1.1:

	Use Cases	
ID	L3VPN Service Provisioning	Section

1.1	L3VPN for 3G/4G Services [L3VPN/Dot1Q/None/None]	5.3
1.2	L3VPN for Enterprises	5.3.2
1.2.1	L3VPN for Enterprise [L3VPN/[None|Dot1Q|QinQ]/BGP/Overwritting]	
1.2.2	L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/BGP [multi-hop, static]/Overwritting]	
1.2.3	L3VPN for Enterprise - [L3VPN/Loopback/None/None] – VRF Lite	
1.2.4	L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/OSPF/Overwritting]	
1.2.5	L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/None/Overwritting]	
1.2.6	L3VPN for Enterprise - [L3VPN/null/null/null]	
1.2.7	L3VPN for Enterprise - [L3VPN/Epipe/BGP[i]/None]	
1.2.8	L3VPN for Entepise - [L3VPN/GRE/None/None]	
1.3	L3VPN Double link in single PE in active/backup	5.3.3
1.4	L3VPN Double link in single PE in load balance	5.3.4
1.5	L3VPN Single access to double PE in backup mode	5.3.5
ID	**L2 VPN Service Provisioning**	**Section**
1.6.1	L2VPN for Enterprise - Hub & Spoke VPLS (H-VPLS)	6.3.1
1.6.2	L2VPN for Enterprise – Full Mesh VPLS	6.3.2
1.6.3	L2VPN for Enterprise – Point-to-Point VPLS (VLL)	0
1.5	EVPN based L2VPN Service Provisioning	6.3.4
ID	**Inventory**	**Section**
2.1	Retrieve HW [Physical] Inventory	7.3.1
2.2	Retrieve logical [Interfaces] Inventory	7.3.2
ID	**Network Topology**	**Section**
3.1	Obtain and export of end to end ip topology using ip domain controllers (IGP Topology)	8.2
3.2	Obtain and export of L2 topology using ip domain controllers (Ethernet links between routers)	8.2.2
3.3	Export potential service end points in IP topology (UNI Topology)	8.2.3
ID	**Traffic Engineering**	**Section**
4.1	LSP create, modify and delete (no constraints) and RSVP-TE signaling	9.4.1
4.2	LSP create, modify and delete (no constraints) and SR	9.4.2
4.3	LSP create, modify and delete with constraints (delay, bandwidth and hop count) – SIGNALLING: (RSVP & SR)	9.4.3
4.4	LSP create, modify and delete with constraints (delay, bandwidth and hop count) – SIGNALLING: EXPLICIT ROUTE (strict and loose hops) (RSVP & SR)	9.4.4
4.5	LSP create, modify and delete with constraints (delay, bandwidth and hop count) Protection: Redundancy 1+1 (RSVP & SR)	9.4.5
ID	**Network Creation**	**Section**

5.1	Device Commissioning	10.2.1
5.2	Network infrastructure configuration templates	10.2.2
ID	**Support to Performance Monitoring**	**Section**
6.1	Reporting performance information [Node: CPU, Temp, Memory]	11.3.1
6.2	Reporting performance information [Interface: Counters, Packets, Errors]	11.3.2
ID	**Telemetry**	**Section**
7.1	Reporting telemetry information [Links: Occupation]	12.1.1
ID	**Telemetry**	**Section**
8.1	Sending Network events/alarms to Fault Management OSS [Node Alarms]	13.1.1
8.2	Sending Network events/alarms to Fault Management OSS [Interface Alarms]	13.1.2

1.2 Network architecture

The proposed IP network architecture is distributed in five levels defined by the following roles:

- **HL1**: Interconnection Routers (RIs)
- **HL2**: P Routers (RNs) - LSR services are not expected to start or terminate on this layer at any time.
- **HL3**: PE routers are aggregators (GWC, SWC) of HL4 / 5 layers, there are variants that include assigning BRAS, CGNAT and SecGW functionalities in this layer
- **HL4**: Access Routers (GWD, SWD) - Oriented to receive corporate clients, fixed services and can add mobile services
- **HL5**: Access Routers (GWT, SWT) - Oriented to receive mobile clients and in special cases fixed services

The HL5-4 from a region are usually concentrated in the HL3. This aggregation device has the *Autonomous System Border Router* (ASBR) role as well as Inline Router reflector for its Region. Thus, to forward the traffic from the L3VPN services, the ASBR routers from each region establish an *eBGP* session against the core routers. This session exports the Router ID plus Label information of all the routers in the region using BGP-LU. Additionally, there is another *eBGP* session between the HL3 of the region and the core Router-Reflectors to export the VPNv4 routes from each VPN service. This *eBGP* session requires a mandatory a Next-Hop-Unchanged configuration to avoid network loops or misconfigured paths. All of this control plane setup allows the creation of an end-to-end LSP from the access HL5 to the platforms without changes in the configuration during the service provisioning.

Additionally, to deploy any of this service the network has to fulfil the following basic requirements:

- IGP connectivity established between HL3 and HL5.
- LDP / RSVP session between HL3 and HL5.
- MPLS enable session between HL3 and HL5.
- MP-BGP (family vpnv4, ipv4)

Please note that this nomenclature based on HLx roles will be spread across this document to improve readability.

2

SBI Communication Protocols

The Southbound Interface (SBI), is the interface, based on standards, between the SDN Domain Controller and the Network Elements. The MUST project has several objectives ...

2 SBI Communication Protocols

The Southbound Interface (SBI), is the interface, based on standards, between the SDN Domain Controller and the Network Elements. The MUST project has several objectives. One of them includes Southbound standardization. The standardization must include a communication protocol that will depend on the use case. Thus, for the use cases described in this document the standard communication protocols requested are:

- For Service L3 and L2 Service Provisioning: **NETCONF** [2]
- For Inventory support: **NETCONF** [2]
- For Network Topology:
  - **BGP-LS** [3]
  - NETCONF to collect **LLDP information**. Note that LLDP will run between the network elements and not with the controller.
- For Traffic Engineering: **PCEP** [3]
- For telemetry and statistics**: gRPC** [5], **NETCONF**[2]

In this section a description of each the protocols and Its requirements is provided:

2.1 NETCONF

The NETCONF protocol defined in RFC 6241 [2] propose a simple mechanism through which a network device can:

- be managed,
- be queried (configuration retrieved)
- be uploaded and manipulated.

NETCONF uses a simple Remote Procedure Call RPC-based mechanism to facilitate communication between a client and a server. The client can be a script, application running as part of a network domain controller. The server is typically a network device.

The protocol allows the device to expose a full, formal application programming interface (API). MUST has selected SSH as the transport protocol and XML as the encoding mechanism, for this purpose. Applications can use this straightforward API to send and receive full and partial configuration data sets.

The support of each of the use cases described in this document implies de following actions:

Creation
Modification
Deletion

Accordingly, in order to be supported as the MUST project the network equipment MUST be compliant with the standard RFC 6241 "Network Configuration Protocol (NETCONF)", including the complete set of base protocol operations (NOTE: Apply to all NE to be deployed regardless of the level in hierarchical architecture)

get
get-config
edit-config
copy-config
delete-config
lock
unlock
close-session
kill-session

Furthermore, NETCONF defines the existence of one or more configuration datastores and allows configuration operations on them. A configuration datastore is defined as the complete set of configuration data that is required to get a device from its initial default state into a desired operational state. The configuration datastore does not include state data or executive commands.

Thus, the network element MUST support the following datastores (in case some of the datastores are not supported, the vendor should provide the alternative workflow to achieve the use cases):

Start-Up
Running
Candidate

Any equipment MUST advertise its NETCONF capabilities by sending them during an initial capability exchange as defined in RFC 6241. The equipment MUST implement the following NETCONF capabilities:

writable-running
candidate configuration
confirmed-commit
rollback-on-error
xpath
(*) The “validate” capability support shall be considered optional but valuable

Finally, the equipment MUST provide mechanisms (e.g., RADIUS) to support mutual authentication and authorization with the SDN controller.

2.2 BGP-LS

BGP-LS [3] shall be used to extract the layer 3 topology from the designated node, and also report the Traffic Engineering (TE) information according to RFC8571 [6], in particular latency, and it shall deliver Sub TLV present in the IS-IS TE according to RFC8570 (same as RFC7810, or equivalent in OSPF-TE metric extensions RFC7471), as documented in

- Unidirectional average latency (IS-IS/BGP-LS) SubTLV 33 /1114

- MIN/MAX average latency SubTLV 34 /1115
- Delay variation SubTLV 35 /1116
- Unidirectional Residual Bandwidth SubTLV 37 /1118
- Unidirectional Available Bandwidth SubTLV 38 /1119
- Unidirectional Utilized Bandwidth SubTLV 39 /1120

Parameters in use will depend on the specific use case. Value calculation is subject to vendor implementation.

2.3 PCEP

PCEP (Path Computation Element (PCE) Communication Protocol), as described in RFC4657 and RFC5440 will be the protocol used as a way to exchange candidate paths between the PCE and the nodes (also referred to as PCC(Path Computation Client)).
The PCE component of the controller shall be stateful as per RFC8051 definition. There are two main ways of definition and modification of paths for a Stateful PCE, with different use cases, that will be required in MUST.

- **Delegation**: an operation to grant a PCE temporary rights to modify a subset of LSP parameters on one or more LSPs of a PCC ("delegated" LSPs). The (unique) PCC that owns the PCE state for the LSP has the right to delegate it. For intra-domain LSPs, this PCC should be the LSP head end.
- **PCE Initiation**: assuming LSP delegation granted by default, a PCE can issue recommendations to the network.

A stateful PCE shall comply with PCEP protocol extensions as described in RFC8231: Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE. In case the Network uses Segment routing (SR) as transport (RFC8402, Segment Routing Architecture), specific PCEP extensions will be needed as well according to RFC8664. PCEP extensions for automatic bandwidth adjustment when employing an Active Stateful PCE for both PCE-Initiated and PCC-Initiated LSPs are described in RFC8733

The way the PCE shall optimise the paths is described at RFC8232 Optimizations of Label Switched Path State Synchronization Procedures for a Stateful PCE.

A summary of the MUST requirements within this set of recommendations: The PCE, maintains two sets of information DB for use in path computation. Stateful condition requires reliable state synchronization mechanisms between the PCE DB and the network spokes, the PCCs.

1. **Traffic Engineering Database (TED)**: topology and resource state in the network. This information can be obtained by a stateful PCE from the network using mechanisms like BGP-LS, or gRPC Telemetry (in this case a parameter mapping to the topology shall be needed)
2. **LSP State Database (LSP-DB)**, attributes of all active LSPs in the network, such as their paths through the network, bandwidth/resource usage, switching types, and LSP constraints

On top of stateful, MUST is selecting an **Active Stateful PCE**: a PCE that may issue recommendations to the network. For example, an Active Stateful PCE may use the Delegation mechanism to update LSP parameters in those PCCs that delegate control over their LSPs to the PCE

Note that a Passive Stateful PCE would not actively update LSP state.

RFC 8281 Describes the creation and deletion of PCE-initiated LSPs under the stateful PCE model using additional extensions to PCEP in a Stateful PCE Model (or latest version).

The security implementation of PCEP is critical and shall depend on every operator requirement, but one of the options under consideration is the use of TLS. RFC8253 PCEPS: Usage of TLS to Provide a Secure Transport for PCEP

The controller shall be able to steer traffic according to policies across candidate paths that can be either MPLS tunnels or SR paths according to (draft-ietf-spring) Segment Routing Policy Architecture. The candidate paths and the policies association can be done in different ways but PCEP is a good option to make it dynamic and active-stateful (draft-ietf-pce-segment-routing-policy-cp-05). LSP paths association extensions in PCEP are described in (RFC 8697).

3

OpenConfig Data Models

As mentioned, one pillar of the Southbound Interface (SBI) is the use of NETCONF protocol with Device Models based on OpenConfig (OC).

3 OpenConfig Data Models

As mentioned, one pillar of the Southbound Interface (SBI) is the use of NETCONF protocol with Device Models based on OpenConfig (OC) [7] .

OpenConfig is an informal working group of network operators sharing the goal of moving our networks toward a more dynamic, programmable infrastructure by adopting software-defined networking principles such as declarative configuration and model-driven management and operations. OpenConfig models are vendor-neutral data models defined in YANG. They cover actual operational needs from use cases and requirements from multiple network operators.

The main modules used in this IP specification are:

- **bgp**: This set of modules describe the BGP protocol configuration. They are used in the service-related use cases to handle the BGP protocol (in CE-PE routing).
- **interfaces**: Model for managing network interfaces and subinterfaces. For the use cases that are currently defined, it is used to configure the interfaces of the VPNs
- **lldp**: This module defines configuration and operational state data for the LLDP protocol. Used for Topology discovery.
- **lacp**: This module describes configuration and operational state data for Link Aggregation Control Protocol (LACP) for managing aggregate interfaces.   It works in conjunction with the OpenConfig interfaces and aggregate interfaces models. In this specification, it is used in case of Link Aggregation (LAG) in the CE-PE connection plus LACP protocol.

- **local-routing**: This module describes configuration and operational state data for routes that are locally generated, i.e., not created by dynamic routing protocols. These include static routes, locally created aggregate routes for reducing the number of constituent routes that must be advertised, summary routes for IGPs, etc. In this specification, local-routing is imported in network-instance to configure the static routes.
- **mpls**: This module provides data definitions for configuration of Multiprotocol Label Switching (MPLS) and associated protocols for signaling and traffic engineering. In this specification it is used to create PCC-Initiated LSPs.
- **network-instance**: An OpenConfig description of a network-instance. This may be a Layer 3 forwarding construct such as a virtual routing and forwarding (VRF) instance, or a Layer 2 instance such as a virtual switch instance (VSI). Mixed Layer 2 and Layer 3 instances are also supported. In this specification this is the main module for the service instantiation. The openconfig-network-instance imports additional models to fulfill the VPN instance requirements:

    - Interfaces
    - VLANs
    - Protocols

- **ospf**: An OpenConfig model for Open Shortest Path First (OSPF) version 2. In this specification, it is used in case OSPF is used as PE-CE protocol.
- **isis**: This module describes a YANG model for ISIS protocol configuration. In this specification this model is used to configure commissioning parameters on network creation use cases.
- **platform**: This module defines a data model for representing a system component inventory, which can include hardware or software elements arranged in an arbitrary structure. The primary relationship supported by the model is containment, e.g., components containing subcomponents.

The platform module is the base for the Hardware Inventory use case and it is composed of the following models:

  - PLATFORM - openconfig-platform.yang – It constitutes the main model to define the hardware components of a network device.
  - CPU - It augments the platform model to add specific parameters of a CPU component.
  - FAN - openconfig-platform-fan.yang – It augments the platform model to add specific parameters of a FAN component.
  - LINECARD - It augments the platform model to add specific parameters of a Linecard component.
  - PORT - It augments the platform model to add specific parameters of a port component.
  - PSU - It augments the platform model to add specific parameters of a PSU (Power Supply Unit) component.

- **policy**: This module describes a YANG model for routing policy configuration. It is a limited subset of all of the policy configuration parameters available in the variety of vendor implementations but supports widely used constructs for managing how routes are imported, exported, and modified across different routing protocols. The routing policy is used in this specification for the distribution of routes within MP-BGP in the VPN related use cases.
- **qos**: This module defines configuration and operational state data related to network quality-of-service. In this configuration it is used for setting the use case to overwrite the QoS in a VPN setup.
- **rib**: Defines a data model for representing BGP routing table (RIB) contents.
- **segment-routing**: Configuration and operational state parameters relating to the segment routing. This module defines a number of elements which are instantiated in multiple places throughout the

OpenConfig collection of models. In this specification it is used to create SR LSPs.

- **types**: This module contains a set of general type definitions that are used across OpenConfig models. It can be imported by modules that make use of these types.
- **vlan**: This module defines configuration and state variables for VLANs.in addition to VLAN parameters associated with interfaces
- **system**: This module describes the managing system-wide services and functions on network devices such as NTP server, DNS, AAA, etc.
- **telemetry**: This module creates the configuration for the telemetry systems and functions on the device, including destinations, protocols, encodings, sensors and subscriptions.
- **acl**: This module defines configuration and operational state data for network access control lists (i.e., filters, rules, etc.).

These modules and the rest of the OpenConfig initiative yang models can be found at https://github.com/openconfig/public, under the folder /release/models.

All of those modules are used as a whole to configure a device depending on every aspect concerned as depicted in the following diagram:

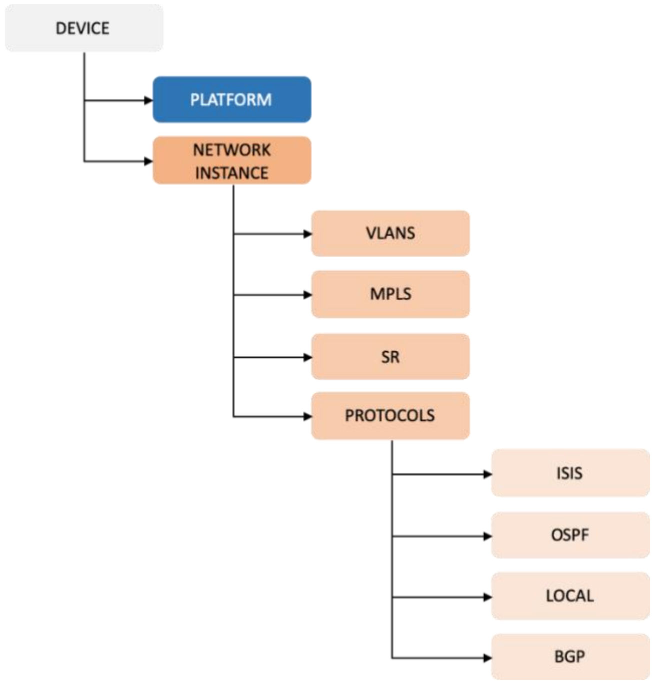


Figure 2. OpenConfig Modules Example

Furthermore, the support of certain OpenConfig modules is highly related with the use case. For example, in the case of a configuration of a L3VPN just certain modules must be supported by the network device. As depicted in this example, the Network instance module has cross-relations with Interfaces and QoS.

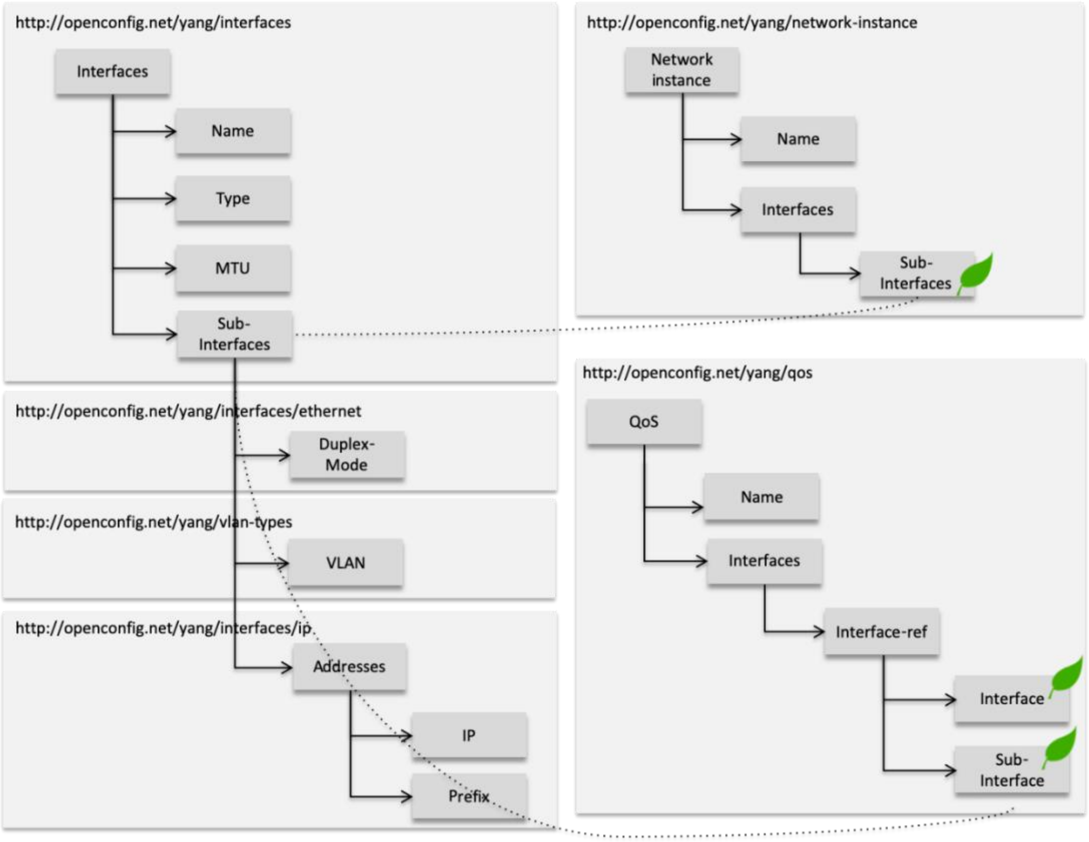

http://openconfig.net/yang/interfaces
Interfaces
Name
Type
MTU
Sub-Interfaces
http://openconfig.net/yang/interfaces/ethernet
Duplex-Mode
http://openconfig.net/yang/vlan-types
VLAN
http://openconfig.net/yang/interfaces/ip
Addresses
IP
Prefix
http://openconfig.net/yang/network-instance
Network instance
Name
Interfaces
Sub-Interfaces
http://openconfig.net/yang/qos
QoS
Name
Interfaces
Interface-ref
Interface
Sub-Interface

Figure 3. OpenConfig module relationship

OpenConfig models are currently the preferred options based on the implementation maturity. Nevertheless, the MUST operators will keep monitoring IETF standards and will include them when the level of maturity reaches the same level.

4

Feature Bundles

In this Technical Requirement Document (TRD) MUST operators have considered that the best way to gather the information about the proposed implementation based on OpenConfig data models, is to group the required set of XPaths by functional blocks, called Feature Bundles.

4 Feature Bundles

In this Technical Requirement Document (TRD) MUST operators have considered that the best way to gather the information about the proposed implementation based on OpenConfig data models, is to group the required set of XPaths by functional blocks, called Feature Bundles.

The Feature Bundles are sets of Open Config XPaths intended to cover co-related management functions exposed by the target SDN Controller SBI / Routers NBI.

4.1 BASIC_L3VPN_MANAGEMENT

Feature Bundle	Basic L3VPN Management
Id	BASIC_L3VPN_MANAGEMENT
HL Description	Provisioning of L3VPN services through the configuration and management of a network-instance (VRF) and its related parameters. It includes basic VRF definition with RD, BGP import/export routing policy for VPN routes and binding interfaces to VRF (IPv4 only, IPv6 not included).
Covered Functions	▪ Provide a name and a description for the new L3VPN service. ▪ Define type and data plane encapsulation for network-instance. In this category, only BGP based L3VPNs (type=L3VRF) over MPLS (encapsulation-type=MPLS) will be allowed regardless of type parameter has more options. Besides, an special network-instance. type=DEFAULT_INSTANCE should be used to refer to Routing Global Table on the device. ▪ Set the route distinguisher that should be used for the VRF when it is signaled via BGP. ▪ Configure a router-id to identify the routing device and used by BGP and OSPF to function in a routing instance. Depending on vendor implementation, if you do not configure a router identifier, the IP address of the first interface to come online is used. This is usually the loopback interface. Otherwise, the first hardware interface with an IP address or the IP address of interface first binding to the L3VPN instance. ▪ Enable/disable the configured network-instance on the network-element (Note: Some vendors implementation could not allow disable it so that the network instance will be enable immediately after creation and keep enable until it is deleted). ▪ Define the mode of label allocation to be used for L3VPN route entries in the network instance, that is, per prefix, per next-hop or a single label per VRF. ▪ Enable the Address Families (AF) supported within the L3VPN (only IPv4 Unicast supported in this version of feature-bundle) ▪ Configure multiple Route Target export and import through the application of inter-instance-policies. Note that, at the moment, in "**openconfig-network-instance"** yang model there isn't a place where configuring directly import/export-route-targets as with RD. Instead, first you have to create an export and import policy with **"openconfig-routing-policy"** module , referring RT as matching Extended Communities, and then use them within **"openconfig-network-instance"** as inter-instance-policies. ▪ Configure and enable interface(s)/subinterface(s) to be bound to the L3VPN at: L2 (port speed/duplex, MTU, single or double tagging, Lag members if apply), L3 (IP address, Loopback type interface). In case

	of using GRE Tunnel to associate to the L3VPN other parameters as source and destination IP addresses, MTU or TTL of the interface Tunnel must be provided. ▪ Associate/Bind interfaces previously configured to the network-instance. Interfaces could be of any type either physical single/aggregated (LAG) or logical subinterfaces/interface tunnel. ▪ Retrieve the list (names) of all L3VPN configured on the device. ▪ Retrieve params configuration of a specific L3VPN on the device (description, RD, RT policies, AF, interface bounds). ▪ Delete a L3VPN completely.
Not covered	▪ IPv6 VRF support ▪ Data Plane encapsulation other than MPLS ▪ Signaling and routing protocols configuration (LDP, RSVP, ISIS, OSPF, BGP,...) on the device ▪ Routing policies others than VPNv4 routes RT import/export ▪ L2-L3 stitching (terminate VLL or VPLS to VRF service) ▪ Multicast VPN support
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1424916611/1.+BASIC+L3VPN+MANAGEMENT

4.2 INTERFACE_MANAGEMENT

Feature Bundle	**INTERFACE_MANAGEMENT**
Id	INTERFACE_MANAGEMENT
HL Description	Parametrization needed to configure at L2 and L3 (IPv4 only, IPv6 not included) Ethernet interfaces (UNI/NNI), including sub-interfaces and link-aggregation (LAG static & LACP), as well as GRE logical tunnels. Configuration involves also reading related fields.
Covered Functions	▪ Basic physical interface configuration (name, description) including interface type as per **ietf-if:interface-type**. Enable/disable. Status and attributes, even the ones system-assigned (HW-mac-address, admin-status) can also be read ▪ Focus in Ethernet interfaces (speed configuration, auto-negotiation, duplex mode). VLAN (oc-vlan:tpid) subinterfaces can be configured, and IPv4 addresses can be assigned to those, either explicitly or as dhcp-clients. IPv4 MTU can be defined per subinterface. ▪ LAG assignment to logical aggregate-id. Minimum link number definition for the LAG interface to stay available. Includes LACP messaging speed and active/passive, with control node assignment via priority values. ▪ Interface configuration required to deliver transport of Ethernet frames or IP datagrams (typically for L2VPN or L3VPN), meaning the attachment of those VPN services will be performed in other feature bundles, but can be done in ports or virtual interfaces (dot1q, QinQ) already configured in this bundle. VLAN-ID parameters to be match at incoming frames can be configured for single-tagged, and double-tagged (inner and outer VLAN-ID) variants.

	▪ GRE tunnels can be defined and enabled with explicit destination and source IPv4 address, and IPv4 prefixes are attached to the tunnel, with its MTU. ▪ IPv4 neighbor addresses to link layer at sub-interface level can be read (to be static entries in ARP cache).
Not covered	▪ Non-Ethernet interface details. ▪ IPv6 addresses for sub-interfaces ▪ IPv6 GRE tunnels, IPv6 traffic over GRE tunnels ▪ BNG to fixed customer interfaces are not included in this release (PPPoE IPoE. L2TP) ▪ P bits, ACL, QoS ▪ performance statistics are at INTERFACE_PERFORMANCE_STATISTICS ▪ LAG maximum number of members. system-id-mac. Load balancing algorithms within LAG. LAG recovery setting as revertive/non revertive ▪ VRRP routed-VLAN interfaces (IPv4 and IPv6)
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1469120630/2.+INTERFACE+MANAGEMENT

4.3 BASIC_TRAFFIC_ENGINEERING

Feature Bundle	**Basic Traffic Engineering**
Id	BASIC_TRAFFIC_ENGINEERING
HL Description	This involves the provisioning of an MPLS tunnel to carry services between 2 points in a network running an IGP (can be ISIS or OSPF) using either SR or RSVP as signaling protocols.
Dependencies / pre-requisites	▪ **13. CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT** ▪ OSPF/ISIS routing (defined in another feature bundle) ▪ MPLS Configuration (defined in another feature bundle)
Covered Functions	▪ Tunnel setup based on: RSVP-TE, LDP and Segment Routing (SR). ▪ In addition to enabling MPLS TE on interface, we enable RSVP TE on every interface ▪ Configuring CSPF (constrained Shortest Path First) to calculate the shortest path destined for a specified node. ▪ Configure IGP TE (OSPF or ISIS). Done for all LSRs in an MPLS domain, a TEDB is generated on each LSR including: o Cost style o Traffic Engineering level o Configuring TE attributes for a link o Link bandwidth o Adjust-interval

  - Adjust-threshold
  - Enable
  - Max-bandwidth
  - Min-bandwidth
  - Set-bandwidth
  - Specification-type
  - Signaled-bandwidth
- Configuring an explicit path (Define the nodes through which the MPLS TE tunnel must pass or the nodes that are excluded from the MPLS TE tunnel)
  - Explicit path name
  - Address
  - Hop-type
  - Index
- Configuring the Tunnel interface
  - Admin-status
  - Description
  - Hold-priority
  - Metric
  - Metric-type
  - Name
  - Preference
  - Protection-style-requested
  - re-optimize-timer
  - Setup-priority
  - Signaling-protocol
  - Soft-preemption
  - Source
  - Type
  - MTU
- Configuration of P2P-Tunnel-attributes
  - destination
  - Primary / Secondary path
    - candidate-secondary-path (priority, secondary-path): For primary path
    - name
    - CSPF-tiebreaker
    - explicit-path-name
    - Hold-priority
    - path-computation-method
    - path-computation-server
    - preference
    - retry-timer
    - setup-priority
    - use-CSPF
    - path-constraints (metric-upper-bound, preference, type)

	▪ Discovery of state of segment routing (SR) te-policies configured through PCEP or BGP protocols). Including: o color o candidate-paths o segment-lists o SIDS
Not covered	▪ Manually configured tunnel (static CR-LSP) - for nodes with low performance. o Ingress: outgoing label and next-hop address o Transit node : inbound interface name, next-hop address and outgoing label o Egress : incoming label and inbound interface name. ▪ BFD for MPLS LSP and BFD for Tunnel interface ▪ Configuration of segment routing (SR) te-policies (TE-policies configuration is only available through PCEP or BGP protocols).
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511326062/3.+BASIC+TRAFFIC+ENGINEERING

4.4 POLICY_FORWARDING_FOR_TUNNELS

Feature Bundle	**Policy Forwarding for Tunnels**
Id	POLICY_FORWARDING_FOR_TUNNELS
HL Description	Policy-based forwarding provides the possibility of applying user defined actions when forwarding packets (including data-plane operations such as encapsulation or decapsulation) based on criteria (rules) according to L2, L3, L4 header values, and steering matched packets through a selected set of MPLS Tunnels (RSVP-TE LSP).
Covered Functions	▪ Matching Rules definition for different packet header values: ▪ L2 ▪ source-mac ▪ destination-mac ▪ ethertype ▪ L3 (IPv4) ▪ source-address ▪ destination-address ▪ DSCP ▪ Protocol number (payload type) ▪ L4 ▪ source-port ▪ destination-port ▪ Action to be applied for packets matching the rule

	■ path-selection-group: Steer traffic through selected set of RSVP-TE tunnels (In order to select between the LSPs within the path-selection-group, the system should determine which LSP provides the best path to the next-hop for the routed packet). ■ Apply forwarding-policy to ingress packets coming from interface(s), sub-interface(s) ■ Packets ingress on the referenced interface should be compared to the match criteria within the specified policy, and in the case that these criteria are met, the forwarding actions specified applied.
Not covered	■ Rule matching criteria for IPv6 stack headers ■ Rule matching criteria for MPLS label header ■ Rule matching criteria associated with Segment Routing Traffic Engineering policies. ■ Action steering traffic through Segment-Routing Paths or SR-TE policies ■ Steer traffic to IPSec Tunnels. ■ Actions not covered: ■ Discard ■ Next-hop ■ Encapsulate/Decapsulate GRE ■ Matching Rules definition for different packet header values: ■ L3 (IPv4) ■ TTL Hop count ■ L4 ■ TCP flag
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511194923/4.+POLICY+FORWARDING+FOR+TUNNELS

4.5 L3VPN_CE_PE_ROUTING

Feature Bundle	**L3VPN CE-PE ROUTING**
Id	L3VPN_CE_PE_ROUTING
HL Description	Configuration of routing protocols OSPFv2, BGP, Static, routes between PE and CE, including basic routing policies for redistribution (only for IPv4 stack, not IPv6 stack). Read state of neighbor s.
Covered Functions	■ New "network instance" BGP basic global configuration: AFI/SAFI name, type & enable, AS number and Router-id configuration as dotted quad). Multiple paths for the same NRLI can be allowed (set maximum) or not. ■ BGP neighbor 's configuration, description, enable of peer (cease/start) including configuration of local-as for session with the neighbor if different from global AS. Neighbor IPv4 address and peer-AS number, config, including peer group name and type (iBGP/eBGP). Multi-hop enabling of eBGP neighbor s. Read state of config and also operational session-state BGP neighbor s ■ OSPFv2 instance configuration at PE-CE interfaces: area identifier, router-id for the protocol in dotted quad format, list of interfaces enabled in the area configuration (interface identifiers,

	network and authentication types, passive or not, BFD enabling, including sub-interfaces). Status read of interface metric. ▪ Static routes: Configuration of destination prefix for the static route, either IPv4 or IPv6 (inet:ip-prefix). List of next hops to be utilized for the static route being specified: IP address, interface, sub-interface, local defined (e.g., drop). ▪ Propagation connections between forwarding or routing tables (FIB/RIB), leaking of entries used to redistribute routes, using import policies. These fields are also in use by other bundles like PE_PE_ROUTING_MPBGP_MANAGEMENT and BASIC_L3VPN_MANAGEMENT to propagate VPN related information. The bundle specifies Source-protocol plus address-family and the destination table, as well as the import policy reference (name) to be used (default is reject route) ▪ Routing policies definition, policy chains for accepting or rejecting routes, including statements for condition and action. Either tags or prefixes with their mask length can be used as attributes to match. Statements included as conditions are to check the protocol/method used to install the route into the local routing table-type (and OSPF area as match condition). Policy result statement (actions) include to set the specific OSPF area into which to import a prefix, and the metric. ▪ Configuration of BGP peering attributes in the CE-PE sessions. It Includes the internal and external BGP forwarding policy parameters as well as some advanced loop prevention mechanisms like the Autonomous System (AS) overwrite. Finally, inbound and outbound filters can be applied to all updates that ensure conformance to local policies.
Not covered	▪ IPv6 router address for BGP peers ▪ Other PE-CE protocols like OSPFv3, RIP, IS-IS (now only as IGP) ▪ OSPFv2 metric configuration. ▪ Static routes in IPv6 are admitted by the data model format used in this bundle, but IPv6 is not included in some other feature bundles at this stage on an MPLS data plane.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519845413/5.+L3VPN+CE+PE+ROUTING

4.6 ACL_MANAGEMENT

Feature Bundle	ACL Management
Id	ACL_MANAGEMENT
HL Description	Configuration of network access control lists (i.e., filters, rules, etc.). ACLs are organized into ACL sets, with each set containing one or more ACL entries. ACL sets are identified by a unique name, while each entry within a set is assigned a sequence-id that determines the order in which the ACL rules are applied to a packet. Individual ACL rules specify match criteria based on fields in the packet, along with an action that defines how matching packets should be handled. Entries have a type that indicates the type of match criteria, e.g., MAC layer, IPv4, IPv6, etc.
Covered Functions	▪ Create the ACL o Configure the name or number for the ACL

	o Provide a description o Define the type (IPv4, IPv6, L2, MPLS, Mixed,) o Provide the ID (user defined identifier) ▪ Define the rules used to match the packet o destination address o DSCP o protocol o source address o destination port o source port ▪ Define the action for the matched packets o Forwarding-action (drop, accept, reject) o Log-action (log-syslog, log-none) o define the sequence to apply the ACL (sequence-id) ▪ Apply the ACL to an interface (interface / sub-interface) o Specify the direction to apply the ACL on the interface (ingress / egress) o Set-Name o Type
Not covered	
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511194937/6.+ACL+MANAGEMENT

4.7 QOS_MANAGEMENT

Feature Bundle	**QOS MANAGEMENT**
Id	QOS_MANAGEMENT
HL Description	Parametrization related to QoS. It includes the different stages of the QoS traffic processing: Classification, Policing, Marking, Queuing and Scheduling.
Covered Functions	▪ Ingress traffic classifier conditions based on different packet header layers: ▪ L2 ▪ source-mac ▪ destination-mac ▪ ethertype ▪ L3 ▪ source-address ▪ destination-address ▪ DSCP ▪ protocol number (payload type)

	■ hop-limit (TTL) ■ L4 ■ source-port ■ destination-port ■ tcp-flags ■ MPLS ■ EXP bits ■ TTL ■ Label values ■ Remark traffic matching classifier: ■ Set DSCP ■ Set CoS ■ Set EXP bits ■ Define queues (A queue may be explicitly configured, or implicitly created by the system based on default queues that are instantiated by a hardware component) ■ red-queues ■ minth ■ maxth ■ Actions ■ Assign traffic matching classifier to queues ■ Set DSCP ■ Set dot1q ■ Set mpls-tc ■ Define Scheduler-Policies for queues and its parametrization: ■ Associate input/output classifiers, queues and scheduler-policies to interfaces
Not covered	■ IPv6 stack ■ Hardware-specific QoS vendor implementation ■ Hierarchical QoS
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519845377/7.+QOS+MANAGEMENT

4.8 L2VPN_VPLS_VLL_SERVICES_MANAGEMENT

Feature Bundle	**L2VPN_VPLS_VLL_SERVICES_MANAGEMENT**
Id	L2VPN_VPLS_VLL_SERVICES_MANAGEMENT
HL Description	Provisioning of point-to-point (VLL) and multipoint-to-multipoint (VPLS) L2VPN services through the configuration and management of a network-instance (VSI) and its related parameters. It includes basic VSI definition along with configuration for PW (T-LDP Pseudowires) and MAC Tables.

Covered Functions	▪ Definition of connection points within a Layer 2 network instance. Each connection point consists of a set of interfaces only one of which is active at any one time. ▪ List of local and remote endpoint that can be used for each connection point and their configuration without distinction in their parameters: ▪ Endpoint id ▪ Precedence (active one is lowest in number) ▪ Type ▪ Remote system parameters to identify the host of remote endpoints for the PWE3: ▪ IP address of the device (remote-system) ▪ Virtual-circuit-identifier ▪ site-id (CE identifier) ▪ Forwarding database configuration of the VSI, MAC tables ▪ MAC learning enabling ▪ MAC aging time, ▪ maximum entries ▪ MAC mobility and duplication check
Not covered	▪ active-active set of interfaces ▪ BGP signaled L2 services ▪ L2VPN termination on L3VPN scenario. Only l2vpn at both sides. ▪ Spanning tree and any other loop prevention mechanisms.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519813009/8.+L2VPN+VPLS+VLL+SERVICES+MANAGEMENT

4.9 BGP_AND_PBB_MPLS_BASED_EVPN_MANAGEMENT

Feature Bundle	**BGP_AND_PBB_MPLS_BASED_EVPN_MANAGEMENT**
Id	BGP_AND_PBB_MPLS_BASED_EVPN_MANAGEMENT
HL Description	This involves the provisioning of Provider Backbone MPLS based VPN using BGP in the control plane for distributing MAC address reachability information over the MPLS network operations in the core provides significant advantages, including lower control-plane scale, simpler control-plane operation, and faster convergence compared to the regular EVPN, making it a superior solution for Data Center Interconnection (DCI) and E-LAN offerings.
Dependencies / pre-requisites	▪ **13. CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT** ▪ OSPF/ISIS routing (defined in another feature bundle) ▪ MPLS Configuration (defined in another feature bundle)

Covered Functions	■ Configuration network instance ■ Configuration of the forwarding database of the network instance ■ anycast-gateway-mac ■ arp-proxy ■ arp-suppression ■ flood-unknown-unicast-supression ■ nd-proxy ■ nd-suppression ■ Configuration of MAC Mobility ■ duplicate-IP-detection-interval ■ IP-mobility-threshold ■ MAC-mobility ■ MAC-mobility-threshold ■ MAC-mobility-window ■ Interface configuration for the network instance ■ IRB-anycast-gateway ■ MAC-pinning ■ Configuration of PBB-EVPN instances and B-MAC addresses on each PE. ■ EVPN instance ■ auto-rd ■ encapsulation-type ■ evi ■ multicast group ■ multicast mask ■ P2MP replication ■ replication mode ■ route-distinguisher ■ service type ■ import/export policy ■ auto-rt ■ import route-target ■ export route-target ■ Provider Backbone b-component (backbone-facing PBB core instance) and i-component (customer or access facing interface or routing instance) ■ backbone-src-mac ■ b-component name ■ i-sid ■ Configuration of a PBB-EVPN source address on each PE ■ EVPN source address ■ Configuration of an ESI for each PE interface that connects to a CE ■ esi

	▪ interface ▪ name ▪ redundancy-mode ▪ sub-interface ▪ Configuration of a BGP EVPN peer relationships between PEs ▪ L2VPN EVPN family prefix limit ▪ max-prefixes ▪ prevent-teardown ▪ restart-timer ▪ warning-threshold-pct
Not covered	▪
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511194951/9.+BGP+AND+PBB+MPLS+BASED+EVPN+MANAGEMENT

4.10 BFD_MANAGEMENT

Feature Bundle	**BFD Management**
Id	BFD_MANAGEMENT
HL Description	Bidirectional Forwarding Detection (BFD) provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, forwarding and control planes. BFD is enabled at the interface and routing protocol levels.
Covered Functions	▪ Enabling BFD on interfaces (link failure detection), on physical and logical interfaces. ▪ BFD parametrization on local interfaces (discriminator, desired-minimum-tx-interval, required-minimum-receive, detection-multiplier). ▪ Viewing parameters relating to the BFD peers which are seen over this interface. ▪ Enabling BFD for static routes. ▪ Enabling BFD for OSPF session (on interfaces). ▪ Enabling BFD for ISIS session (on interfaces). ▪ Enabling BFD for PIM session (on interfaces). ▪ Enabling BFD for BGP neighbor s.
Not covered	▪ Enabling BFD over LSPs. Not covered by the models yet. ▪ Authentication.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511194965/10.+BFD+MANAGEMENT

4.11 DEVICE_INTERNAL_COMPONENTS_MANAGEMENT

Feature Bundle	**Device Internal Components Management**
Id	DEVICE_INTERNAL_COMPONENTS_MANAGEMENT
HL Description	Device inventory including hardware or software elements arranged in an arbitrary structure.
Covered Functions	▪ Currently device components are covered in this feature bundle for the following components of the platform: ▪ Line card state o Power admin state (config/state) o Slot_id ▪ Transceiver o Transceiver data state: ▪ Connector type ▪ Enabled (also config) ▪ ethernet-pmd ▪ fault-condition ▪ form factor ▪ Present ▪ serial-no ▪ Vendor ▪ Port config/state o channel-speed o num-channels ▪ Power Supply o Power supply data ▪ enabled (config/state) ▪ Capacity ▪ Common details and statistics for all components are included: ▪ Details of the component feature: name, valid, description, hardware-version, id, location, operational status, parent serial number, software version, properties (name, value) ▪ Configuration/state of subcomponent name (which denotes the hardware model hierarchy).
Not covered	▪ Removable - info about whether the HW component is replaceable.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511194983/11.+DEVICE+INTERNAL+COMPONENTS+MANAGEMENT

4.12 LLDP_MANAGEMENT

Feature Bundle	**LLDP Management**

Id	LLDP_MANAGEMENT
HL Description	This shows the configuration of the Link Layer Discovery Protocol (LLDP) to obtain details about the network topology (as defined in IEEE 802.1ab), changes in the topology, and detect incorrect configurations on the network.
Covered Functions	▪ Enable device to send LLDP packets with local system information to neighbors and parse LLDP packets received from neighbors ▪ Enabled ▪ System level state of the LLDP protocol ▪ Configure the LLDP timers between the devices. ▪ hello-timer - System level hello timer for the LLDP protocol ▪ Configuration of the local and remote interfaces ▪ Enable ▪ Name ▪ State discovery information: ▪ local: ○ system-name ○ Chassis-id ▪ Neighbor s: ○ system-name ○ chassis-id ○ port-id ○ port-description ○ ttl
Not covered	▪ Configuration of the following parameters: ▪ suppress-TLV-advertisement ▪ system-name ▪ system-description ▪ Chassis-id ▪ Chassis-id-type ▪ TLV neighbor information parameters: ▪ Age ▪ Last-update ▪ LLDP packet counters
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511326102/12.+LLDP+MANAGEMENT

4.13 CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT

Feature Bundle	Core Routing Signalling Protocols Management
Id	CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT
HL Description	Advanced configuration of signalling and routing protocols used in the Core, including IGP (ISIS) and those used for MPLS, RSVP/LDP (only for IPv4 stack, not IPv6 stack).
Covered Functions	▪ ISIS routing configuration on the device including: ▪ Instance ID ▪ ISO address (NET) ▪ Level capability ▪ Metric style ▪ Interface attributes (metric, passive, circuit type, level, hello-interval, enable bfd) ▪ AFI/SAFIs ▪ LDP Synchronization ▪ Authentication ▪ Graceful restart and timers ▪ Maximum paths (ECMP) ▪ Enable IGP shortcuts ▪ Manage LSP (Link-State PDUs) overload-bit and attached-bit (set, suppress, ignore) ▪ Inter-level propagation policies ▪ Enable Traffic-Engineering support ▪ Enable Segment-Routing (SR-MPLS) extensions including definition of Global Block (SRGB) and Local Block (SRLB) that are to be used by the ISIS instance. Additionally, prefix and adjacency SIDs assignment at interface level. ▪ MPLS configuration on the device including: ▪ Enable on interfaces ▪ Global label block allocation. ▪ RSVP protocol configuration on the device including: ▪ Enable on interfaces ▪ Graceful restart ▪ Session state ▪ LDP protocol configuration on the device including: ▪ Enable on interfaces ▪ Neighbor authentication ▪ Graceful restart ▪ Targeted LDP attributes ▪ OSPFv2 routing configuration on the device including: ▪ OSPFv2 instance configuration (Area identifier, router-id for the protocol in dotted quad ▪ format) ▪ OSPFv2 Multi-Area (Area configuration interface identifiers, network and authentication types, passive or not, BFD enabling, including sub-interfaces). ▪ IGP-LDP Synchronization (Globally, and at Interface level).

	- Passive Interfaces - LSA Filtering - Graceful-Restart - Virtual Links - Routing Policies - The status read of the interface metric.
Not covered	- ISIS for IPv6. - LDP or RSVP for IPv6. - OSPFv3. - Static Routes (see L3VPN_CE_PE_ROUTING feature-bundle). - IS-IS extensions for Segment Routing operating on IPv6 Data Plane.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519878145/13.+CORE+ROUTING+SIGNALLING+PROTOCOLS+MANAGEMENT

4.14 TELEMETRY_MANAGEMENT

Feature Bundle	Telemetry Management
Id	TELEMETRY_MANAGEMENT
HL Description	Configuration for the telemetry function on the device.
Covered Functions	- It’s considered the usage of dynamic and persistent subscription. - Dynamic subscription definition: A dynamic subscription is typically configured through an RPC channel, and does not persist across device restarts, or if the RPC channel is reset or otherwise torn down. - Persistent subscription definition: A persistent telemetry subscription is configured locally on the device through configuration, and is persistent across device restarts or other redundancy changes. - Creation of destination group-id, destination groups allow the reuse of common telemetry destinations across the telemetry configuration, group id identify the destination group. - Configuration of IP address of the telemetry stream destination group. - Configuration of port number for protocol (UDP or TCP) for the telemetry stream destination - Configuration of sensor group-id. - Configuration of path to a section of operational state of interest (the sensor). Configuration value for dynamic subscription: - Group_id: Unique identifier for the destination group. State values for dynamic subscription: - reference to the path of interest state. - IP address of the telemetry stream destination. - Protocol (UDP or TCP) port number for the telemetry stream destination. - Specific encoding or RPC framework for telemetry messages to and from the network element

	▪ Heartbeat-interval, Maximum time interval in seconds that may pass between updates from a device to a telemetry collector ▪ id of the System generated identifier of the telemetry subscription. ▪ Selected transport protocol for the telemetry stream. ▪ Originated-QoS-marking, DSCP marking of the packets generated by the telemetry subsystem on the network device. ▪ Sample-interval, Time in milliseconds between the device's sample of a telemetry data source. ▪ suppress-redundant flag: Boolean flag to control suppression of redundant telemetry updates to the collector platform. If this flag is set to TRUE, then the collector will only send an update at the configured interval if a subscribed data value has changed. Otherwise, the device will not send an update to the collector until expiration of the heartbeat interval. Configuration/state values for persistent subscription: ▪ Encoding, Selection of the specific encoding or RPC framework for telemetry messages to and from the network element. ▪ local-source-address, IP address which will be the source of packets from the device to a telemetry collector destination. ▪ name, identifier of the telemetry subscription. ▪ originated-qos-marking, DSCP marking of packets generated by the telemetry subsystem on the network device. ▪ protocol, selected transport protocol for the telemetry stream. ▪ group-id, destination group id references a configured group of destinations for the telemetry stream. ▪ heartbeat-interval, Maximum time interval in seconds that may pass between updates from a device to a telemetry collector. ▪ sample-interval, Time in milliseconds between the device's sample of a telemetry data source. ▪ sensor-group, Reference to the sensor group which is used in the profile. ▪ suppress-redundant: Boolean flag to control suppression of redundant telemetry updates to the collector platform. If this flag is set to TRUE, then the collector will only send an update at the configured interval if a subscribed data value has changed. Otherwise, the device will not send an update to the collector until expiration of the heartbeat interval
Not covered	▪ Exclude filters.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1511195001/14.+TELEMETRY+MANAGEMENT

4.15 PE_PE_ROUTING_MPBGP_MANAGEMENT

Feature Bundle	**Pe-PE Routing MP-BGP Management**
Id	PE_PE_ROUTING_MPBGP_MANAGEMENT
HL Description	This involves the provisioning of a BGP/MPLS IP VPN (L3VPN) between PE devices using MPBGP to advertise VPN routes and MPLS to forward VPN packets on the IP backbone networks of the service

	providers. The network can provide VPN services for users so that multiple private networks can communicate across the backbone network of the carrier. VPN routes are isolated from the public network routes on the backbone network, and the routes of VPN instances are isolated from each other.
Covered Functions	▪ Configuration of the network-instance on the PE to manage VPN routes o name o address-family o default-import-policy o dst-protocol o import policy o src-protocol o route-limit ▪ afi ▪ alarm-threshold ▪ maximum ▪ Configuration of route-policy o Prefix set ▪ mode ▪ name ▪ ip-prefix ▪ maxlength-range o tag-set ▪ name ▪ tag-value o Policy definition ▪ name ▪ actions ▪ policy-result ▪ bgp-actions ▪ set-local-preference ▪ set-med ▪ set-next-hop ▪ set-route-origin ▪ set-as-path-prepend ▪ asn ▪ repeat-n ▪ set-community ▪ method ▪ options ▪ communities ▪ community-set-ref ▪ set-ext-community ▪ method ▪ options ▪ communities ▪ community-set-ref ▪ set-tag ▪ mode ▪ tag ▪ tag-set ▪ conditions ▪ call-policy ▪ install-protocol-eq

        - match-prefix-set
          - match-set-options
          - prefix-set
        - match-tag-set
          - match-set-options
          - tag-set
        - bgp-conditions
          - afi-safi-in
          - community-set
          - ext-community-set
          - local-pref-eq
          - med-eq
          - next-hop-in
          - origin-eq
          - route-type
          - match-as-path
            - as-path-set
            - match-set-options

- Configure route exchange between PEs. On each PE, the following will be done in the global mode
  - asn
  - router-id
  - graceful-restart
    - enable
  - route-selection-options
    - enable
  - use-multiple-paths
    - enabled
    - ebgp
      - maximum-paths
    - ibgp
      - maximum-paths
  - address-family identifier (AFI) and Subsequent address-family identifier (SAFI)
    - afi-safi-name
    - enabled
    - Add-paths
      - receive
      - send
      - send-max
  - Neighbors
    - description
    - auth-password
    - enabled
    - local-as
    - neighbor-address
    - peer-as
    - peer-group
    - peer-type
    - remove-private-as
    - route-flap-damping
    - send-community
    - ebgp-multihop
      - enabled
      - multihop-ttl

	▪ route-reflector ▪ route-reflector-client ▪ route-reflector-cluster-id o peer-group ▪ peer-group name ▪ afi-safi ▪ afi-safi-name ▪ add-paths ▪ receive ▪ send ▪ send-max ▪ apply-policy ▪ export-policy ▪ import-policy ▪ enabled ▪ apply-policy ▪ default-import-policy ▪ export-policy ▪ import-policy ▪ ipv4-labeled-unicast ▪ prefix-limit ▪ prevent-teardown ▪ restart-timer ▪ warning-threshold-pct ▪ ipv4-unicast ▪ prefix-limit ▪ max-prefixes.
Not covered	
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519910917/15.+PE+PE+ROUTING+MPBGP+MANAGEMENT

4.16 INTERFACE_PERFORMANCE_STATISTICS

Feature Bundle	**Interface Performance Statistics**
Id	INTERFACE_PERFORMANCE_STATISTICS
HL Description	Collection of counters for interface performance measurements. The counters can be reseted.
Covered Functions	▪ Per interface counters including: ▪ The total number of packets received and transmitted on the interface, including all unicast, multicast, broadcast and bad packets etc. ▪ The total number of octets received and transmitted on the interface, including framing characters. ▪ Per interface Ethernet specific counters, including: ▪ The number of received errored frames due to a carrier issue. ▪ The total number of frames received with bad Frame Check Sequence (FCS) (fragmented, that is with length less than 64 octets, and between 64 and 1518).

	▪ The number of received errored frames due to interrupted transmission issue. ▪ Time when the counters of an interface where last reseted.
Not covered	▪ Reset counters.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519943681/16.+INTERFACE+PERFORMANCE+STATISTICS

4.17 HARDWARE_PERFORMANCE_STATISTICS

Feature Bundle	**Hardware Performance Statistics**
Id	HARDWARE_PERFORMANCE_STATISTICS
HL Description	Collection of counters for device hardware performance measurement, no static values.
Covered Functions	▪ Currently statistics are covered in this feature bundle for the following components of the platform: o **CPU** o **Transceiver** o **Power Supply** o **FAN** ▪ Common statistic for all components o **Memory** For components that have associated memory, these values report information about available and utilized memory. ▪ Available - The available memory physically installed, or logically allocated to the component. ▪ Utilized - The memory currently in use by processes running on the component, not considering reserved memory that is not available for use. o **Temperature -** Temperature in degrees Celsius of the component. Values include the instantaneous, average, minimum, and maximum statistics. If avg/min/max statistics are not supported, the target is expected to just supply the instant value. ▪ Interval - Time interval over which the min/max/average statistics are computed by the system. ▪ Avg – Average value ▪ Instant value ▪ Max - the maximum value of the statistic over the time interval ▪ Max-time- The absolute time at which the maximum value occurred. ▪ Min - the minimum value of the statistic over the time interval ▪ Min-time - The absolute time at which the minimum value occurred. ▪ Description of functionalities: o **CPU component.** ▪ **U**tilization statistic for the CPU component: ▪ Interval over which the min/max/average statistics are computed by the system. ▪ Average value of the percentage measure of the statistic over the time interval. ▪ The instantaneous percentage value. ▪ Maximum value of the percentage measure of the statistic over the time interval. ▪ Max-time - The absolute time at which the maximum value occurred.

	▪ Min - The minimum value of the percentage measure of the statistic over the time interval. ▪ Min-Time - The absolute time at which the minimum value occurred. o **Transceiver component** ▪ Transceiver statistic for client port transceiver data ▪ **Input power** statistics: The input optical power of a physical channel in units of 0.01dBm, which may be associated with individual physical channels, or an aggregate of multiple physical channels (i.e., for the overall transceiver). ▪ Interval The arithmetic mean value of the statistic over the time interval. ▪ Instant - The input optical power instant ▪ Max - The maximum value of the statistic over the time interval. ▪ Min - The minimum value of the statistic over the time interval. ▪ **Laser-bias-current** statistics: The current applied by the system to the transmit laser to achieve the output power. The current is expressed in mA with up to two decimal precisions ▪ Instant value ▪ Maximum value ▪ Minimum value ▪ **Output-power** statistics: The output optical power of a physical channel in units of 0.01dBm, which may be associated with individual physical channels, or an aggregate of multiple physical channels (i.e., for the overall transceiver). For an aggregate, this may be a measurement from a photodetector or a a calculation performed on the device by summing up all of the related individual physical channels ▪ Instant value. ▪ Interval value. ▪ Max- the maximum value of the statistic over the time interval. ▪ Min - The minimum value of the statistic over the time interval. o **Power supply component** ▪ Input Current - The input current draw of the power supply. ▪ Input voltage - Input voltage to the power supply. ▪ Output-current - The output current supplied by the power supply. ▪ Output-power - Output power supplied by the power supply. ▪ Output-voltage - Output voltage supplied by the power supply. o **FAN** ▪ Speed.
Not covered	▪ No included statistic for the following components: o Integrated-circuit o Transceiver – FEC performance and error counters. ▪ Coherent optics HW is not yet considered. ▪ There are not further statistics in the models for these specific HW components, o Chassis* o Port* o Fabric*

	o Storage* o Backplane* o Software-module* o Linecard*
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1470332966/17.+HARDWARE+PERFORMANCE+STATISTICS

4.18 ALARM_MANAGEMENT

Feature Bundle	Alarm Management
Id	ALARM_MANAGEMENT
HL Description	This feature bundle defines operational state data related to the management of alarms risen by the device.
Covered Functions	▪ This feature bundle includes the generic alarm structure including the following elements ▪ Alarm id (unique identifier of the alarm in the device within the list of alarms) ▪ Impacted resource (Xpath to the affected resource, for example /platform/component="component_name" or /interfaces/interface="interface_name") ▪ Alt resource (used to correlate with MIB counter). ▪ Alarm description (text information to understand the problem) ▪ Severity (determines the impact of the alarm) ▪ Type id (category of the alarm) ▪ Timestamp (time when the alarm was created) ▪ With this feature bundle the active alarms can be queried and streamed.
Not covered	▪ Administrative alarm lifecycle management. ▪ View cleared alarms. ▪ last change. ▪ operator state changes. ▪ The structure does not include all fields from recommendation X.733, for example: ▪ Alarm code (well know code to identify similar type of alarms). ▪ Probable cause of the alarm. ▪ Alarm name (human readable name to identify similar type of alarms). ▪ The feature bundle does not indicate which elements trigger alarms.
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1519845399/18.+ALARMS+MANAGEMENT

4.19STP_PROTOCOL_MANAGEMENT

Feature Bundle	STP Protocol Management
Id	STP_PROTOCOL_MANAGEMENT
HL Description	STP, defined by IEEE 802.1D, is used to prevents loops . Devices running STP exchange information with one another to discover loops on the network, and then block certain redundant links to eliminate loops. If an active link fails, STP activates a running standby link with the highest priority to ensure network connectivity. For this feature bundle, we will focus on the basic STP, then MSTP, RSTP and rapid-pvst as supported in the OpenConfig model.
Covered Functions	▪ For the basic STP, the following parameters are considered for configuration ▪ In the global mode, o enabled-protocol o bridge-assurance o etherchannel-misconfig-guard o bgpguard-timeout-recovery o loop-guard o bpdu-guard o bpdu-filter ▪ On the interfaces we need to participate in STP o name o edge-port o link-type o guard o bpdu-guard o bpdu-filter ▪ For RSTP o hello-timer o max-age o forwarding-delay o hold-count o bridge-priority o On the interfaces we need to participate in RSTP ▪ name ▪ cost ▪ port-priority ▪ For MSTP o name o revision o max-hop o hello-timer o max-age

	o Forwarding-delay o hold-count o For the mstp-instance ▪ mst-id ▪ vlan ▪ bridge-priority o For the interfaces to participate in mstp ▪ name ▪ cost ▪ port-priority ▪ For rapid pvst o vlan-id o hello-timer o max-age o forwarding-delay o hold-count o bridge-priority o For the interfaces to participate in rapid-pvst ▪ name ▪ cost ▪ port-priority
Not covered	▪ Per VLAN Spanning tree (pvst) ▪ PVST+ ▪ Rapid-pvst+ ▪
External references	https://telecominfraproject.atlassian.net/wiki/spaces/OOPT/pages/1540393052/19.+STP+PROTOCOL+MANAGEMENT

5

L3VPN Service Provisioning Use cases (Category 1)

The L3VPNs are widely used to deploy 3G/4G, fixed and enterprise services mostly due to the fact that several traffic discrimination policies can be applied in the network to transport and guarantee the right SLAs to the mobile customers ...

5 Category 1: L3VPN Service Provisioning

5.1 L3VPN Structure and Classification

The L3VPNs are widely used to deploy 3G/4G, fixed and enterprise services mostly since several traffic discrimination policies can be applied in the network to transport and guarantee the right SLAs to the mobile customers. The L3VPN creates a virtual routing network instance (VRF) in each of the nodes involved in service deployment. This routing instance allows the routing information propagation between the sites involved in the service.

The L3VPN services can be classified using its operational characteristics, thus the following structure have been defined to identify each possible variation of the VPN service. Additionally, it has been included as an identifier of the Use case **(Use case Definition)** to map the relationship between the commercial offer and the real deployment:

- ***Use case Definition***: Type of services deployed in the network.
- ***Functional Parameters***: Used to find common structures in the configured Services. This classification has the following structure:

  a. **VPN Service Type**: Type of service configured
     i) L3VPN
  b. **End Point Connection Type**: Encapsulation details used in the CE-PE connection.
     i) None
     ii) Dot1q
     iii) QinQ
     iv) L2VPN (ipipe/epipe/VLL)
     v) Loopback
  c. Routing Protocol used to connect the CE: Routing details used in the CE-PE connection.

  i) Direct
  ii) Static
  iii) BGP
  iv) OSPF
  v) ISIS
  vi) RIP

d. QoS policies applied in the service:

  i) None
  ii) QoS Overwriting
  iii) CIR / PIR policies

5.2 Atomic Operations

In general, the parameters needed to configure devices via SBI (Netconf) through standard OpenConfig model will have the following six steps:

1. Create VPN (VRF instance definition)
2. Add VPN import/export BGP policies
3. Configure interfaces and subinterfaces to access L3VPN
4. Bind access interfaces/subinterfaces to VRF (L3VPN End Points)
5. Redistribute routing protocols on client connection (CE) inside VRF
6. Append additional configuration related to QoS Policy

Thus, the corresponding general workflow, valid for all VPNs, is as follows:

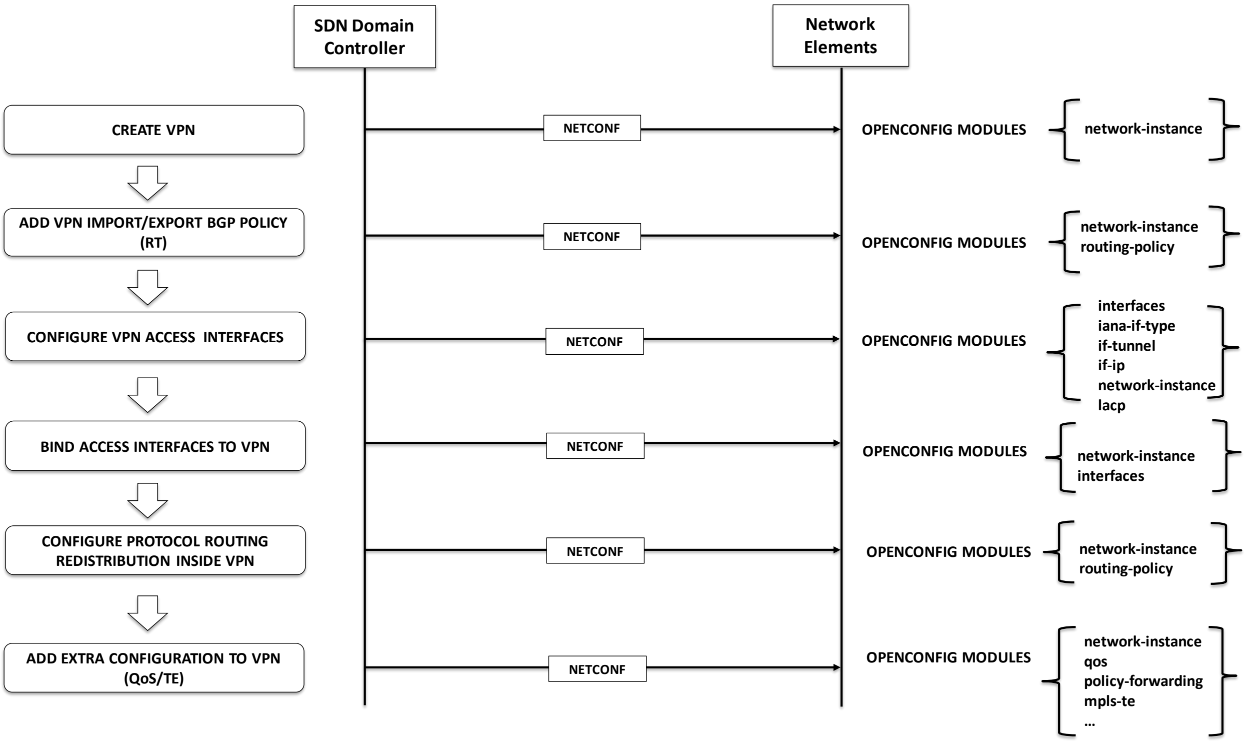


Figure 4. Workflow for L3VPN creation using OpenConfig yang model + Netconf in the SBI

In the following subsections, all the parametrization needed for L3VPN will be described in detail.

5.2.1 Create VPN (VRF instance definition)

The set of parameters required for the creation of the new L3VPN service on Netconf SBI are taken from "openconfig-network-instance" module. The information includes a name, description, and RD, together with address-family and label-allocation mode for the VPN/VRF.

5.2.2 Add VPN import/export BGP policies (VRF route target)

Furthermore, configuration related to RT and policies to be applied to routes imported and exported to/from VRF are added. Generally, first you have to create an export and import policy within "**openconfig-routing-policy**" module and then use it within "**openconfig-network-instance**" module.

In the following table are detailed the features of each parameter:

Apart from these parameters, in some cases, it could be necessary to complete VPN configuration with information related to MP-iBGP policies for a specific neighbor, in addition to those define for the import/export (RT) within VRF as detailed above. For example, to add/delete/modify a BGP attribute (community, as-path, MED, local-pref, etc) to the routes exchanged with neighbors. To do so, again, we first create a policy by using “routing-policy” module, and then we assign that policy to address-family VPNv4 for the corresponding MP-iBGP neighbors.

5.2.3 Configure Interfaces to access to VPN

Before to be associated to VRF, the PE router access interface connecting node B or Enterprise CPE on client side shall be configured, as described in the figure below. The interface could be a physical Ethernet port or a logical interface, like a port+Vlan subinterface, or an aggregation of several physical ports (LAG). Also, in some scenarios, the interface used to access and be associated to the VRF is not a physical interface, but a “logical” interface like a GRE tunnel or a PseudoWire (VLL). In these cases, first the GRE tunnel or PW hast to be defined and then associated to the VRF as if there was a traditional physical interface. The configuration includes L2/L3 parameters like vlan-id (single or double tagging) or IP-address for interface, subinterface, LAG or GRE tunnel and PW, using yang modules **“iana-if-type**”, “**openconfig-interfaces**“, “**openconfig-if-ip**”, **“openconfig-if-tunnel**”, “**openconfig-lacp**”.

5.2.4 Bind interfaces/subinterfaces to VRF (VPN End Points)

The interface and subinterface configured in previous step are not yet associated to a L3VPN service and, therefore, not included into a VRF in the network element. Now, as depicted below, we should bind the interface or sub-interface (or LAG or tunnel or PW) previously defined to the network instance as client interface by using "**openconfig-network-instance**" and "**openconfig-interface**" modules.

5.2.5 Redistribute routing protocols on client connection (CPE) inside VRF

Additionally, route redistribution inside VRF from different protocols (and associated policies if apply) can be configured. OpenConfig provides a set of tables within a VPN/VRF. The tables represent various per-protocol RIBs that can exist in a network instance. Further, OpenConfig models allow for the interconnection of RIBs through an explicit table-connection list within the network-instance model. Thus, to redistribute routes within protocol A to protocol B, an explicit connection between the tables corresponding to protocol A and protocol B should be created.

Accordingly, within OpenConfig "network-instance" module, first it would be necessary to define the protocols involved in the redistribution (direct connections, static routes, OSPF or BGP). Then, associate a table per each protocol and finally create table-connections as required from a source protocol to destination protocol (note: when a BGP instance is created with IPv4 and IPv6 address families enabled, the protocol=BGP, address-family=IPv4 table is created by the system). Although "ACCEPT_ROUTE" is set as default-import-policy, if needed, a policy could be also added to explicit routes being imported into the destination protocol's RIB.

5.2.6 Append additional configuration for QoS

Apart from previous steps, in some cases, it could be necessary to complete VPN configuration with complementary parametrization related QoS policy to apply to inbound/outbound traffic within VPN. Furthermore, could also be needed to add TE constraint in order to steer VPN traffic for a specific path.

Thus, for QoS the corresponding OpenConfig parametrization is detailed in the next table. It includes the different stages of the QoS traffic process:

- **Classification**: Organizes traffic on the basis of whether or not it matches a specific criterion by identifying all key packet fields: CoS, DSCP, IP precedence, or MPLS EXP field of the incoming packets.
- **Policing**: Determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer CIR and PIR. A set of actions should be performed on packets that conform to the CIR and PIR (conform-action), packets that conform to the PIR, but not the CIR (exceed-action), and packets that exceed the PIR value (violate-action).
- **Marking**: All packets that belong to a classification can be remarked. Furthermore, when you configure a policer, packets that meet or exceed the permitted bandwidth requirements (CIR/PIR) can be conditionally passed through, dropped, or marked.
- **Queueing**: Differentiates traffic classes and regulates the queue size based on the classification to avoid network bottlenecks. It uses some algorithms for congestion avoidance such as tail drop, RED or WRED by providing a drop precedence.
- **Scheduling**: Manages congestion by providing a guaranteed bandwidth to a particular class of traffic while also serving other traffic in a fair way. It uses different mechanisms (traffic shaping, WFQ, priority queueing) to limit the maximum bandwidth that can be consumed by a particular class of traffic and ensure that delay-sensitive traffic in a low-latency queue is sent before traffic in other queues.

This section has described the general procedure. In the next sections, the details for each L3VPN use case is described.

5.3 Use Cases

5.3.1 Use case 1.1: L3VPN for 3G/4G Services. [L3VPN/Dot1Q/None/None]

5.3.1.1 Definition

ID	**1.1**
Name	**L3VPN for 3G/4G Services. [L3VPN/Dot1Q/None/None]**
Brief description	Backhaul mobile network represents the interconnection between the data network and the mobile network. L3VPN services are widely deployed in the IP/MPLS networks. It supports the L3 transport of 3G and 4G services, offering specific traffic treatment through QoS policies applied within L3VPN. These services consume several logical resources (RD, RT, VLANs, IP Address, etc.) to be deployed correctly. The maintenance is done in the network daily and several areas get involved.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT QOS_MANAGEMENT L3VPN_CE_PE_ROUTING INTERFACE_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT

Table 1 Definition of Use Case 1.1

5.3.1.2 Description

The scenario that would apply for this use case definition is outlined in the following table.

Name	**1.1. L3VPN for 3G/4G Services. [L3VPN/Dot1Q/None/None]**
Service Type	L3VPN
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	Direct
QoS policies applied in the service	None

Table 2 Use case 1.1 VPN scenario

In the **L3VPN for 3G/4G Services** the nodeB/eNodeB are directly connected to the cell site gateway layer. The cell site gateway acts as a first access layer for nodes that shares the same geographical location. The L3VPN can be implemented over a multiple IGP network, in the example two IGP for backhaul and the one core. IP network is therefore structured in Hierarchy Levels where the first border router receives traffic from the cell site gateway collecting rings of the same geographical location and aggregates them towards the border router to connect with the Core layer.

From the backhaul network point of view, a L3VPN is created on cell site gateway where, as a general rule, and depending on the operator, several sub-interfaces are created, each one with its specific IP Address and VLAN range. Node B interfaces can be assigned to the same or different VPN depending on the specific traffic constraints, such as:

- **S1 Interface**: Control plane and user plane Traffic (S1-MME & S1-U respectively)
- **Sync Interface**: Synchronization Plane Traffic (clock signaling)
- **O&M Interface**: Management Plane Traffic

Finally, the L3VPN created reach the core IGP (HL3 routers) where 3G/4G Core platforms are allocated. Accordingly, the following diagram depicts a generic network scenario to be used as reference for the configuration of this kind of services:

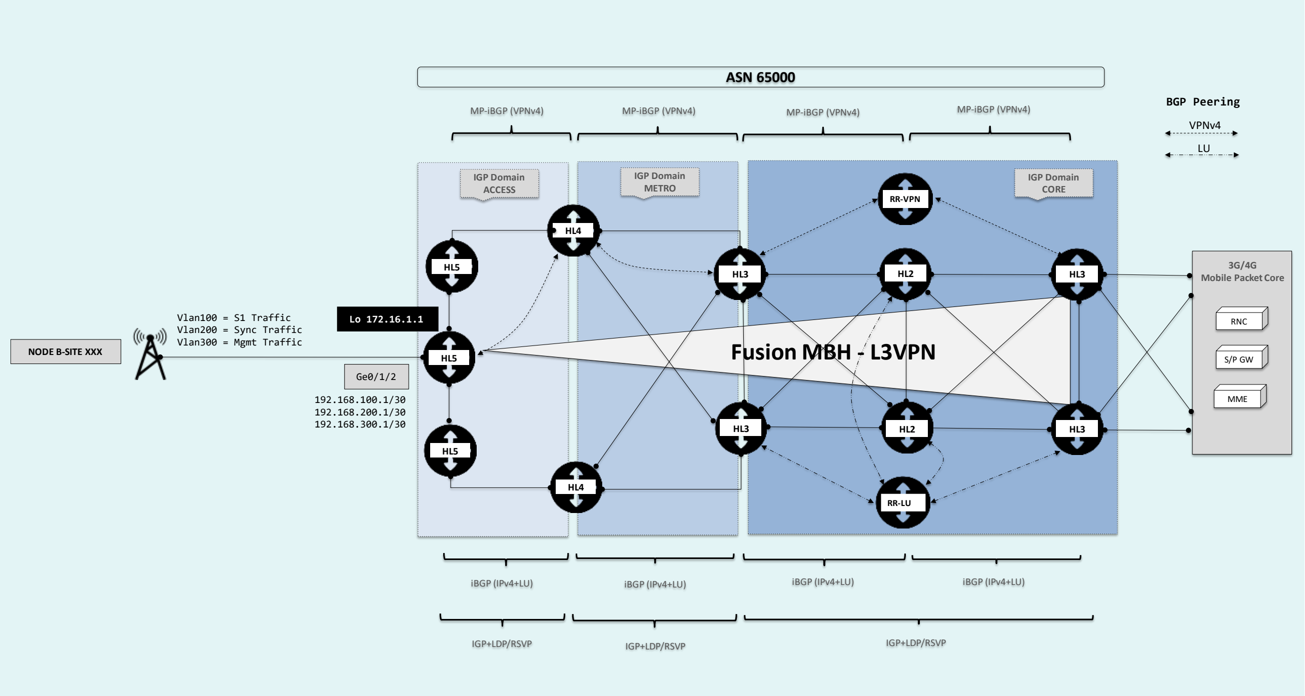


Figure 5. Generic network topology for L3VPN 3G/4G service

To deploy this service several conditions should be evaluated, each of them triggering processes and related configurations, each consuming a subset of the data L3VPN related models, following a flow diagram like this:

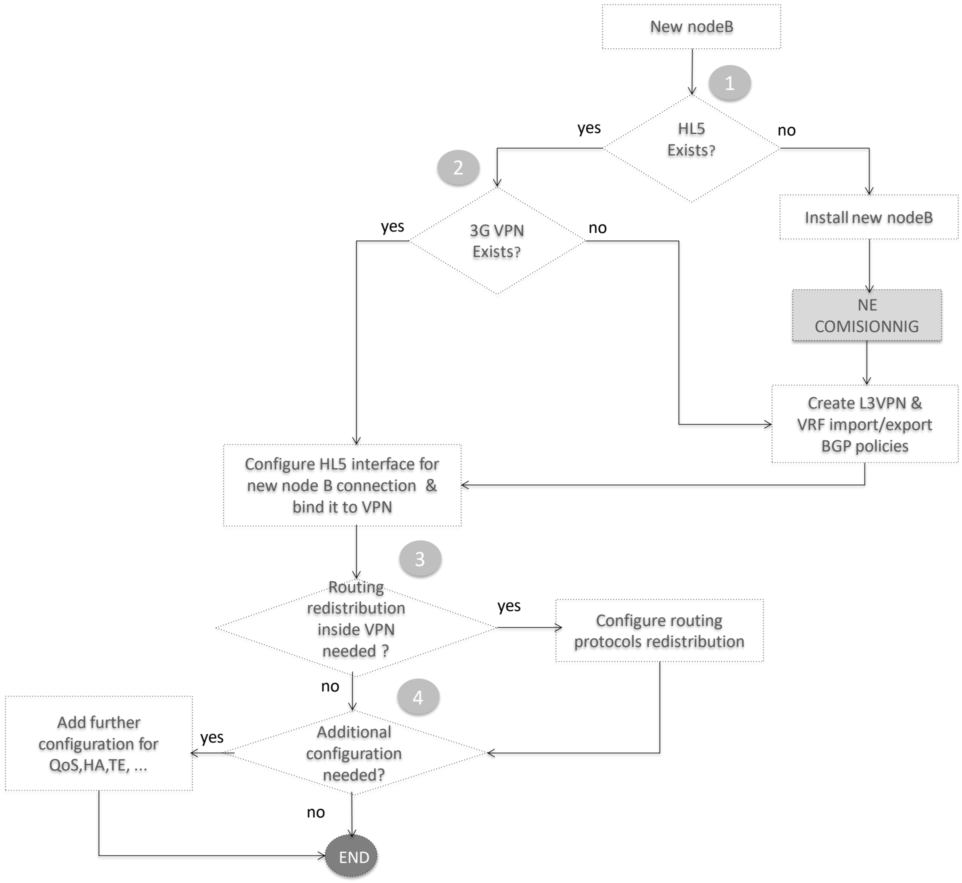

New nodeB
1
HL5 Exists?
yes
no
Install new nodeB
NE COMISIONNIG
Create L3VPN & VRF import/export BGP policies
2
3G VPN Exists?
yes
no
Configure HL5 interface for new node B connection & bind it to VPN
3
Routing redistribution inside VPN needed ?
yes
Configure routing protocols redistribution
no
4
Additional configuration needed?
yes
Add further configuration for QoS,HA,TE, ...
no
END

Figure 6. Configuration process flow for L3VPN 3G/4G service

1 🗆 Is the HL5 (Cell Site gateway) already deployed in the network?

- ***Yes***. *Continue to the next step.*
- ***No***. *Some commissioning process may be done. This Includes Physical/Logical activation of the device, interfaces, and protocols.*

2 🗆 Is the L3VPN already configured at the HL5 Cell Site gateway) receiving the nodeB?

- ***Yes.*** *Just the interface parameters and additional configuration must be included*
- ***No.*** *L3VPN initial parametrization must be done.*

3 🗆 Routing protocols redistribution needed?

- ***Yes.*** *Append corresponding configuration*
- ***No.*** *Continue with the next step*

4 Do the L3VPN need additional configuration?

- ***Yes.*** *Applicable new parameters for QoS, HA, TE, etc. should be included in the configuration*
- ***No.*** *End*

5.3.2 Use case 1.2: L3VPN for Enterprises

5.3.2.1 Definition

ID	**1.2**
Name	**L3VPN for Enterprises**
Brief description	L3VPN for enterprises are customer fitted solutions able to fulfill connectivity and SLAs compliancy between sites using a common infrastructure. This kind of services support changes in topology as well as in parameters configuration regarding End Point Connection Type, Routing Protocol used to Connect the CPE and QoS policies applied in the service.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT QOS_MANAGEMENT L3VPN_CE_PE_ROUTING INTERFACE_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT

5.3.2.2 Description

In the **L3VPN for Enterprise** the Client CPE router in company site is connected to HL4 (or HL5 in some cases, for example where HL4 footprint is not available in the zone or is sharing site with a Node B). The HL4 receives traffic from the Enterprise client CPE and connect the site with the company headquarters routers, typically connected to a HL3 router through the L3VPN in a single or dual homed configuration (other scenario, could be also a connection between two branch company sites, both connected to HL4 respectively).

Furthermore, on HL4/HL5, several sub-interfaces could be created for different services, each one with its specific IP Address and VLAN. For example, they can be assigned to the same or different VPN depending on the use, such as:

- Voice traffic
- Internal Data traffic

- Internet traffic

Accordingly, the following diagram depicts a generic network scenario to be used as reference for the configuration of this kind of services:

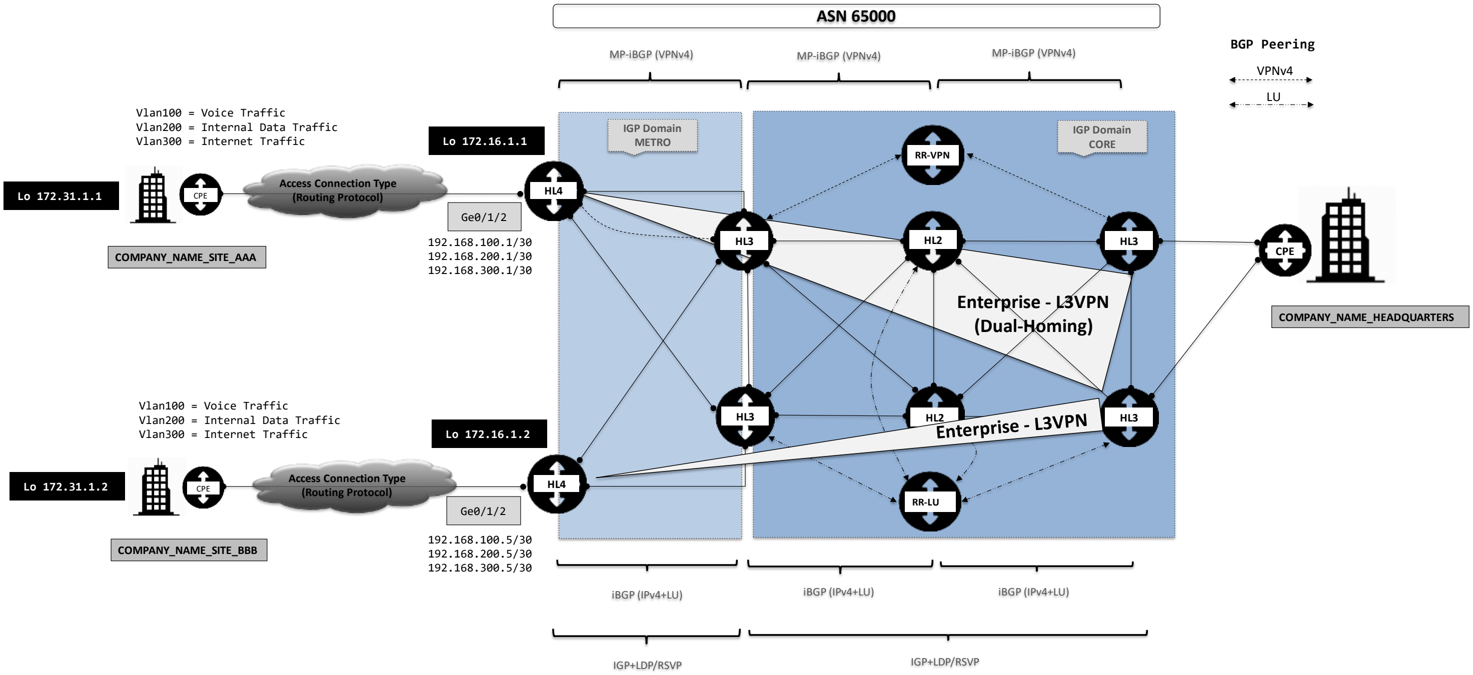


Figure 7. Generic network topology for L3VPN Enterprise Service

As of this general scenario, different L3VPN for Enterprise uses cases are defined, depending on the option selected for:

- End Point Connection Type (plain Ethernet, single Vlan tagging, double Vlan tagging (QinQ), GRE Tunnel, VLL/Epipe)
- Routing Protocol used to Connect the CPE (direct connection, static/default route, OSPF, E-BGP)
- QoS policies applied in the service.

5.3.2.3 Sub use cases – specific configurations

All identified sub use cases applying to L3VPN for Enterprise would be the following:

Use Case 1.2.1 - L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/BGP/Overwritting]

Name	1.2.1 - L3VPN for Enterprises - [L3VPN/[None|Dot1Q|QinQ]/BGP/Overwritting]
Service Type	L3VPN
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	BGP
QoS policies applied in the service	Overwriting

Use Case 1.2.2 - L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/BGP [multi-hop, static]/Overwritting]

Name	1.2.2 - L3VPN for Enterprises - [L3VPN/[None|Dot1Q|QinQ]/BGP [multi-hop, static]/Overwritting]
Service Type	L3VPN
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	BGP [multi-hop, static]
QoS policies applied in the service	Overwriting

Use Case 1.2.3 - L3VPN for Enterprise - [L3VPN/Loopback/None/None] – VRF Lite

Name	1.2.3 L3VPN for Enterprises – VRF Lite
Service Type	L3VPN
End Point Connection Type	Loopback
Routing Protocol used to Connect the CPE	None
QoS policies applied in the service	None

Use Case 1.2.4 - L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/OSPF/Overwritting]

Name	1.2.4 L3VPN for Enterprises - [L3VPN/[None|Dot1Q|QinQ]/OSPF/Overwritting]
Service Type	L3VPN
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	OSPF
QoS policies applied in the service	Overwriting

Use Case 1.2.5 - L3VPN for Enterprise - [L3VPN/[None|Dot1Q|QinQ]/None/Overwritting]

Name	**1.2.5 L3VPN for Enterprises - [L3VPN/[None|Dot1Q|QinQ]/None/Overwritting]**
Service Type	L3VPN
End Point Connection Type	Dot1Q / Q-in-Q
Routing Protocol used to Connect the CPE	Null
QoS policies applied in the service	Overwriting

Use Case 1.2.6 - L3VPN for Enterprise - [L3VPN/null/null/null]

Name	**1.2.6 L3VPN for Enterprise - [L3VPN/null/null/null]**
Service Type	L3VPN
End Point Connection Type	Null
Routing Protocol used to Connect the CPE	Null
QoS policies applied in the service	Null

Use Case 1.2.7 - L3VPN for Enterprise - [L3VPN/Epipe-VLL/BGP/None]

Name	**1.2.7 L3VPN for Enterprise - [L3VPN/Epipe-VLL/BGP/None]**
Service Type	L3VPN
End Point Connection Type	Epipe/VLL (T-LDP PW)
Routing Protocol used to Connect the CPE	BGP
QoS policies applied in the service	None

Use Case 1.2.8 - L3VPN for Enterprise - [L3VPN/GRE/None/None]

Name	**1.2.8 L3VPN for Enterprise - [L3VPN/GRE/None/None]**
Service Type	L3VPN
End Point Connection Type	GRE
Routing Protocol used to Connect the CPE	None
QoS policies applied in the service	None

5.3.3 Use case 1.3: L3VPN Double link in single PE in active/backup

5.3.3.1 Definition

ID	**1.3**
Name	**L3VPN Double link in single PE in active/backup**
Brief description	In this use case, the main objective is creating a L3VPN service with primary and secondary paths working like an active and passive links to support services availability, this UC cover the configuration in a single PE.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT QOS_MANAGEMENT L3VPN_CE_PE_ROUTING INTERFACE_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT

Table 3 Definition of Use Case 1.3

5.3.3.2 Description

The figure below depicts the L3VPN service setting in which there are two links to guaranty the service availability and all customer traffic should be routed throughout only active link while the passive link should be ready to receive all customer traffic in case to have any failure in the active link.

The route mechanism used to create this setting active/backup is the static route protocol and is necessary specify in each next-hop the metric parameter to define the preference of the next hop, the lower metric is more preferable for reachable the network prefix and the higher metric is less preferable and used for passive links. The next-hop could be an IP Address or there are many cases in which is specified interface and sub-interfaces, according to solution design.

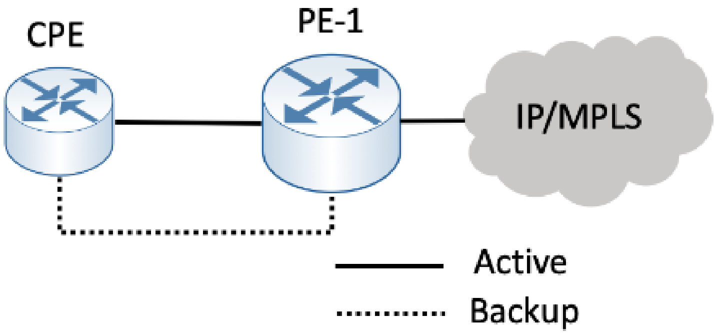


Figure 8. Double link in single PE in active/backup

The list of parameters needed is exclusively to L3VPN services with double access link active and passive mode configuration in the same PE Router, all parameters described below are necessary to implement. The static route protocol is used for this type of solution choosing per link the preference to define the links access roles according to the solution design of high availability.

5.3.4 Use case 1.4: L3VPN Double link in single PE in load Balance

5.3.4.1 Definition

ID	**1.4**
Name	**L3VPN Double link in single PE in load Balance**
Brief description	In this use case, the main objective is create a L3VPN service with two paths, distributing the traffic load per packet and guaranteeing services availability, this UC cover the configuration in a single PE.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT QOS_MANAGEMENT L3VPN_CE_PE_ROUTING INTERFACE_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT

Table 4 Definition of Use Case 1.4

5.3.4.2 Description

The Figure 9. Double Link in single PE in load balance. depicts the L3VPN service setting in which there are two links to guaranty the service availability and all customer traffic should be routed per packet throughout both link at the same time and if any of them has a failure all traffic should be routed by available link.

There are two mechanisms used to create this load balance setting the first one is the static route protocol in this case the static-route has several next-hops with the same metric according for design topology and use the method of send per packet throughout each link. The second is using BGP routing protocol is necessary set up two BGP instances, one per link and the Multipath parameter should be enabled to achieve load balance traffic.

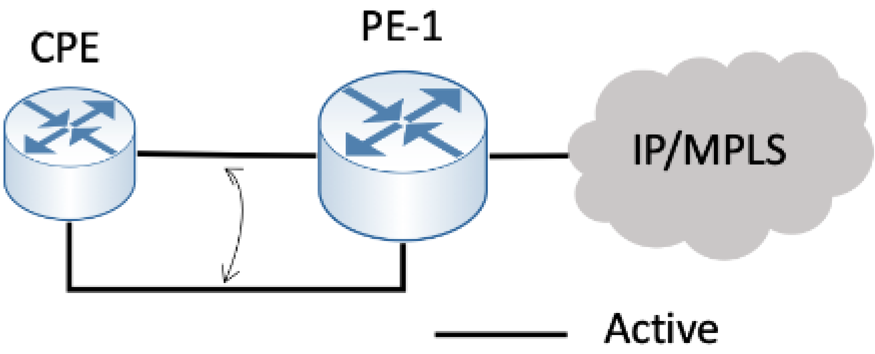


Figure 9. Double Link in single PE in load balance.

The list of parameters needed is exclusively to L3VPN services with double access link in load balance mode configuration in the same PE Router, the static route protocol is used for this type of solution distributing the traffic per packet in each link available with the same preference.

5.3.5 Use case 1.5: L3VPN Single access to double PE in backup mode

5.3.5.1 Definition

ID	1.5
Name	L3VPN Single access to double PE in backup mode

Brief description	In this use case, the main objective is create a L3VPN service with primary and secondary paths working like an active and passive links to support services availability, this UC cover the configuration in two PE's (Primary and Secondary).
Feature Bundles required	BASIC_L3VPN_MANAGEMENT QOS_MANAGEMENT L3VPN_CE_PE_ROUTING INTERFACE_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT

Table 5 Definition of Use Case 1.5

5.3.5.2 Description

The Figure 10. Single access to double PE in backup mode. depicts the L3VPN service setting in which there are two links and two router's PE to guaranty the service availability, all customer traffic should be routed only in a primary PE throughout their access link, the secondary PE with their link access must in passive mode to route all traffic if any failure occurs and the primary path with their PE is down.

The mechanism used to create this backup mode setting is using BGP routing protocol, is necessary set up two BGP instances, one per PE and establish these of BGP sessions against to CE Router in which has BGP settings with both session per Primary and Secondary PE.

There are two mechanisms used to create this backup mode setting the first one is using BGP routing protocol, is necessary set up two BGP instances, one per PE and establish these of BGP sessions against to CE Router in which has BGP settings with both session per Primary and Secondary PE.

And the second mechanism used is a static route protocol in this case using Primary and Secondary PE each one connected to CE, is necessary set a local preference to define the active and passive Router. When the prefix is defined must set a number Tag and it should be associate with a local preference to advertised in BGP Session for another peers.

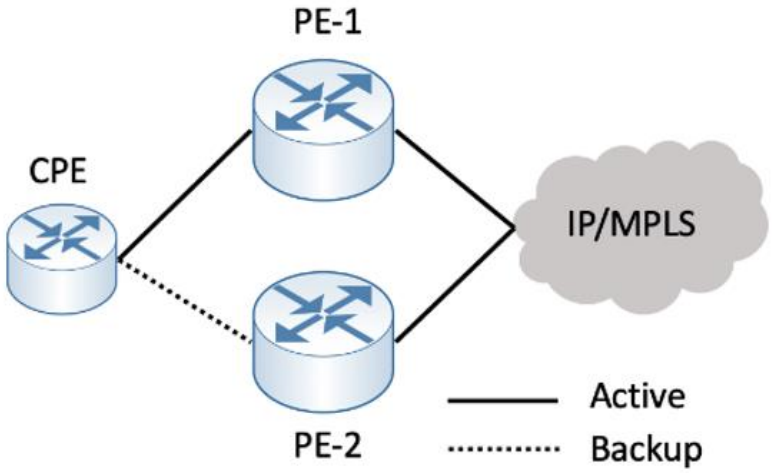


Figure 10. Single access to double PE in backup mode.

The list of parameters needed is exclusively to L3VPN services with double access link active and passive mode configuration in two PE Routers, the BGP route protocol is used for this type of solution choosing per link the metric to define the links access roles according to the solution design of high availability.

6

L2VPN Service Provisioning Use Cases (Category 2)

This L2VPN services are part of the wholesale catalog of services deployed in all of the Service Provider network aimed at providing Point-to-point (P2P) or Point-to-Multipoint (P2M) Ethernet connections. In order to create a L2VPN in the IP network, it is needed to create a virtual switching instance (VSI) in each of the nodes involved in service deployment...

6 Category 2: L2VPN Services Provisioning

6.1 L2VPN Structure and Classification

This L2VPN services are part of the wholesale catalog of services deployed in all of the Service Provider network aimed at providing Point-to-point (P2P) or Point-to-Multipoint (P2M) Ethernet connections. In order to create a L2VPN in the IP network, it is needed to create a virtual switching instance (VSI) in each of the nodes involved in service deployment. This switching instance allows the ethernet information propagation between the sites involved in the service. Besides, in the absence of an auto-discovery mechanism, the identities of all the remote PE routers that are part of a VPLS instance must be configured manually on each PE router. Further, the split horizon rule prevents from loops happening at L2.

The L2VPN services can be classified using its operational characteristics, thus the following structure have been defined to identify each possible variation of the VPN service. Additionally, it has been included an identifier of the use case to map the relationship between the commercial offer and the real deployment:

- ***Use case Definition***: Type of services deployed in the network.
- ***Functional Parameters***: Used to find common structures in the configured Services. This classification has the following structure:

a. **VPN Service Type:** Type of service configured
   i. L2VPN-Full Mesh (VPLS)
   ii. L2VPN-Hub & Spoke (H-VPLS)
   iii. L2VPN-Point-to-Point (VLL)

b. End Point Connection Type: Encapsulation details used in the CE-PE connection.
   i. None

    ii. Dot1q
    iii. QinQ

c. **Routing Protocol used to connect the CE**: N/A

d. **QoS policies applied in the service**:

    i. None
    ii. QoS Overwritting
    iii. CIR / PIR policies

6.2 Atomic Operations

In general, for this specific use case, the parameters needed to configure devices via SBI (Netconf) through standard OpenConfig model will have the following four steps:

1. Create L2VPN (VPLS/VLL instance definition).
2. Configure Interfaces and Sub Interfaces to access L2VPN service.
3. Bind interfaces/sub-interfaces to L2VPN (VPLS/VLL End Points).
4. Append additional configuration related to:
    a. QoS Policy
    b. Redundancy/HA or Load Balancing mechanisms inside L2VPN (PW Redundancy)

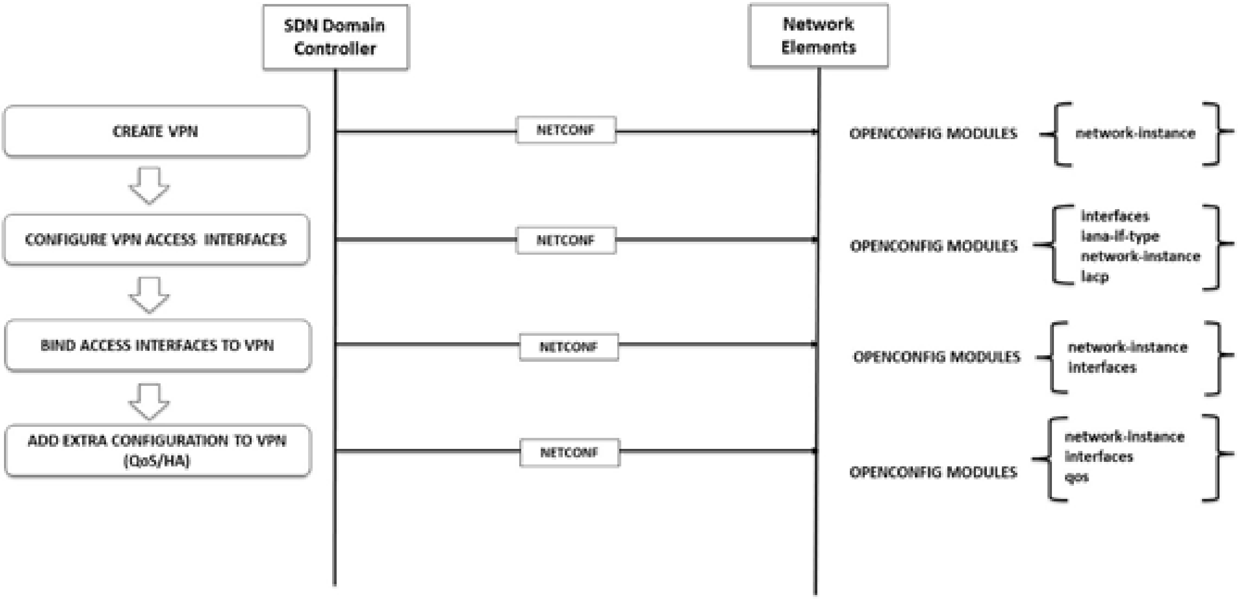


Figure 11. Workflow for L2VPN creation using OpenConfig yang model + Netconf in the SBI

6.2.1 Create VPN (VPLS/VLL instance definition)

The set of general parameters required for the creation of the new L2VPN service on Netconf SBI are taken from “openconfig-network-instance” module. The information includes a name, description, and peers remote address for T-LDP adjacency in order to establish PW for VPLS data-forwarding plane.

6.2.2 Configure Interfaces to access to VPN

Before to be associated to VPLS, the Fusion HL4/HL5 router access interface connecting client side shall be configured. The interface could be a physical Ethernet port or a logical interface like a port+Vlan subinterface, or even an aggregation of several physical ports (LAG). The xpaths used are within “openconfig-interfaces“ and “openconfig-lacp” data models within the INTERFACE_MANAGEMENT and BASIC_L3VPN_MANAGEMENT feature bundles.

6.2.3 Bind interfaces/sunbinterfaces to VPN (VPLS/VLL End Points)

The interface and subinterface configured in previous step are not yet associated to a L2VPN service and, therefore, not included into a VPLS in the network element. Now, we should bind the interface or subinterface (or LAG) previously defined to the network instance as client interface by using “openconfig-network-instance” and “openconfig-interface” modules.

6.2.4 Append additional configuration

Apart from previous steps, in some cases, it could be necessary to complete L2VPN configuration with complementary parametrization related QoS policy to apply to inbound/outbound traffic within VPN and High Availability or Load Balance enhancements in case of dual-homing scenario (PW Redundancy).

For QoS configuration would apply OpenConfig parametrization as detailed in the equivalent section for L3VPN Services Provisioning.

As regards HA and Load Balancing options, in order to configure PW redundancy for L2P2P (VLL) circuits, “Connection-Points” within “network-instance” module should be used.

6.3 Use Cases

6.3.1 Use case 1.6.1: L2VPN for Enterprise - Hub & Spoke VPLS (H-VPLS)

6.3.1.1 Definition

ID	**1.3.1**
Name	**L2VPN for Enterprise - Hub & Spoke VPLS (H-VPLS)**
Brief description	LAN to LAN services are services mainly focused on enterprise customers contracting high volume traffic. This service has a critical impact in terms of SLAs and service cost for Service Providers. The implementation of this service can be achieved through an H-VPLS, where HL3s in the core network are the provider edge (N-PE) router, with full mesh/hub PW connectivity with other HL3, whereas HL5/HL4 routers, distributed over the Seamless MPLS domain on the same metro region or other regions, are the user provider edge (U-PE) or MTU (Multi-Tenant Unit). To reach SLA with clients, spoke PWs are created between HL5/HL4 and HL3, terminating on the full mesh VSI instances. Optionally, Dual-Homed with PW primary/backup to switch to an alternative destination in case of failure of the primary HL3 device can be configured.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT L2VPN_VPLS_VLL_SERVICES_MANAGEMENT ORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT INTERFACE_MANAGEMENT

Table 6 Definition of Use Case 1.6.1

6.3.1.2 Description

The scenario that would apply for this use case definition is outlined in the following table.

Name	**L2VPN for Enterprise**
Service Type	L2VPN - Hub & Spoke VPLS (H-VPLS)
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	None
QoS policies applied in the service	Overwriting

Table 7 Use case 1.6.1 VPN scenario

The end-to-end scenario is depicted in the following picture with the hierarchical VPLS (H-VPLS) scheme in which, instead of a PE router being fully meshed with

LDP sessions, a two-level hierarchy is created involving hub PE routers and spoke PE routers. The hub PE routers are fully meshed with T-LDP sessions, whereas the spoke PE routers has either a pseudo-wire to a single hub PE router or a pair of hub PE routers for redundancy (Dual-Homed). In our case, HL4 or HL5 in the metro network would be spoke routers, where client CPE is connected, with only PWs against a single or a pair of HL3 in the core network.

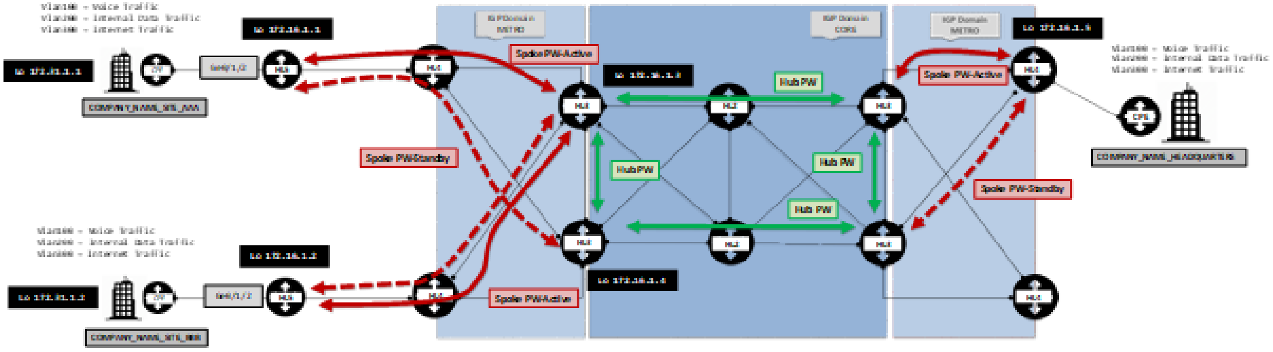


Figure 12. Generic network topology for L2VPN Enterprise service

To deploy these set of services several conditions should be evaluated by means of a flow diagram like the following:

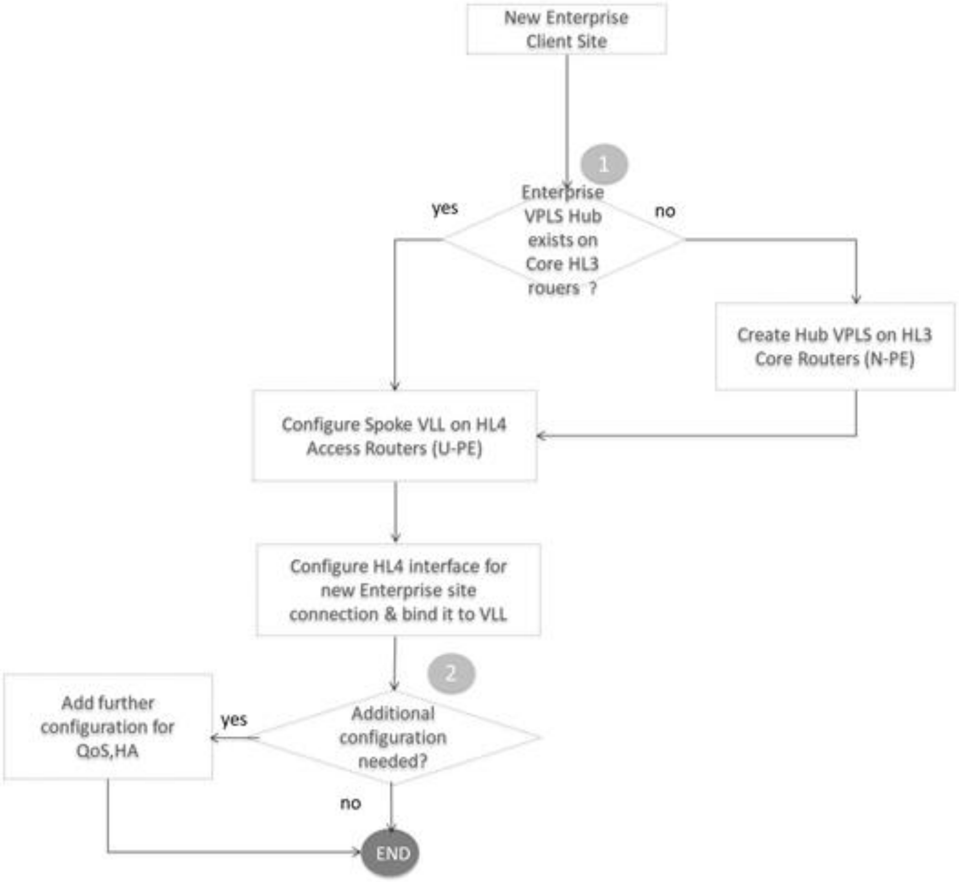

New Enterprise Client Site
1
Enterprise VPLS Hub exists on Core HL3 rouers ?
yes
no
Create Hub VPLS on HL3 Core Routers (N-PE)
Configure Spoke VLL on HL4 Access Routers (U-PE)
Configure HL4 interface for new Enterprise site connection & bind it to VLL
2
Additional configuration needed?
yes
Add further configuration for QoS,HA
no
END

Figure 13. Configuration process flow for L2VPN Hub & Spoke VPLS Enterprise Service

Does it exist a Hub L2VPN configuration in the Core HL3 routers yet?

- Yes. Continue with the next step to configure Spoke VLL part on the Access HL4 routers (U-PE) receiving the client CPE
- No. L2VPN Hub part parametrization on the Core HL3 routers (N-PE) must be done first

Does the L2VPN need additional configuration?

- Yes. Applicable new parameters for QoS, HA should be included in the configuration
- No. End

6.3.2 Use case 1.6.2: L2VPN for Enterprise - Full Mesh VPLS

6.3.2.1 Definition

ID	1.3.2
Name	**L2VPN for Enterprise - IP/Ethernet**
Brief description	LAN to LAN services are services mainly focused on enterprise customers contracting high volume traffic. This service has a critical impact in terms of SLAs and service cost for Service Providers. Apart from previous use case, in some scenarios this service can be implemented through a multipoint-to-multipoint VPLS, where Enterprise sites are connected in mesh topology with full everyone to everyone connectivity. Thus, a mesh of PW between HL4/HL5 where CPE of a site are connected allow to reach any other site distributed over the Seamless MPLS domain on the same metro region or other regions.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT L2VPN_VPLS_VLL_SERVICES_MANAGEMENT CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT INTERFACE_MANAGEMENT

Table 8 Definition of Use Case 1.6.2

6.3.2.2 Description

The scenario that would apply for this use case definition is outlined in the following table.

Name	L2VPN for Enterprise
Service Type	L2VPN – Full Mesh VPLS
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	None
QoS policies applied in the service	None

Table 9 Use case 1.3.2 VPN scenario

The end-to-end scenario is depicted in the following picture with the hierarchical VPLS

The end-to-end scenario is depicted in the following picture. Unlike the previous, with the hierarchical VPLS (H-VPLS) scheme, this use case refers to a L2VPN service implemented through a mesh of T-LDP PW between all routers belonging to the VPLS, HL4 or HL5. In this scenario, user data traffic is forwarded through PWs based on MAC entries allocated in each router where VPLS is created, with the split horizon rule enabled to avoid loops at L2.

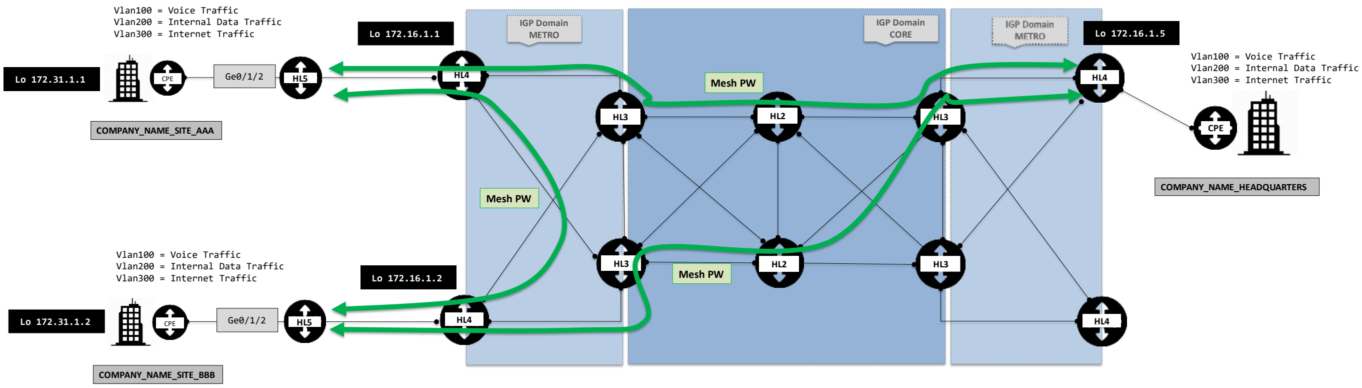


Figure 14. Generic network topology for L2VPN Full Mesh VPLS Enterprise Service

To deploy these set of services several conditions should be evaluated by means of a flow diagram like the following:

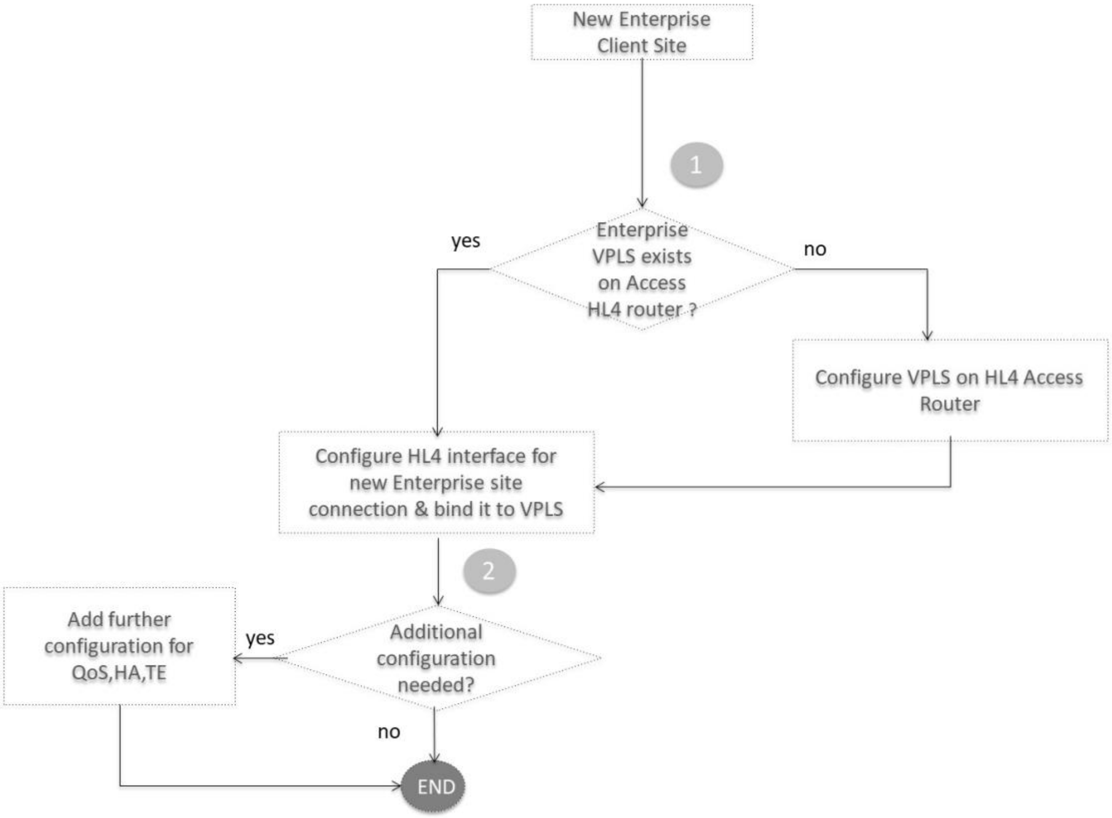

New Enterprise Client Site
1
Enterprise VPLS exists on Access HL4 router ?
yes
no
Configure VPLS on HL4 Access Router
Configure HL4 interface for new Enterprise site connection & bind it to VPLS
2
Additional configuration needed?
yes
Add further configuration for QoS,HA,TE
no
END

Figure 15. Configuration process flow for L2VPN Full Mesh VPLS Enterprise Service

1 ▯ Is the L2VPN already configured on the HL4 receiving the client CPE?

- ***Yes****. Interface parameters must be included in the configuration.*
- ***No.*** *L2VPN initial parametrization must be done*

2 ▯ Do the L2VPN need additional configuration?

- ***Yes.*** *Applicable new parameters for QoS, HA, should be included in the configuration*
- ***No.*** *End*

6.3.3 Use case 1.6.3: L2VPN for Enterprise - Point-to-Point VPLS (VLL)

6.3.3.1 Definition

ID	**1.3.3**
Name	**L2VPN for Enterprise - Point-to-Point VPLS (VLL)**
Brief description	Point-to-Point L2Circuit services are mainly focused on enterprise customers contracting high volume traffic. This service has a critical impact in terms of SLAs and service cost for Service Providers. Network deployment on Point-to-Point L2Circuit is using a VLL between two access devices distributed over the Seamless MPLS domain on other regions or the same region. HL5 or HL4 are the devices that are end-points for this service, while forwarding path can transit through Metro and Core IP network.
Feature Bundles required	BASIC_L3VPN_MANAGEMENT L2VPN_VPLS_VLL_SERVICES_MANAGEMENT ORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT INTERFACE_MANAGEMENT

Table 10 Definition of Use Case 1.6.3

6.3.3.2 Description

The scenario that would apply for this use case definition is outlined in the following table.

Name	**L2VPN for Enterprise**
Service Type	L2VPN – Point-to-Point VPLS (VLL)
End Point Connection Type	Dot1Q
Routing Protocol used to Connect the CPE	None
QoS policies applied in the service	None

Table 11 Use case 1.3.3 VPN scenario

The end-to-end L2VPN Point-to-Point VLL Enterprise Service scenario is depicted in Figure 16.

The end-to-end scenario is depicted in the following picture. This use case considers as endpoints for the VLL two HL4 or HL5 devices, where client CPEs are connected. For that, It will be necessary to create a PWs between end-points HL4 or HL5. T-LDP is used for signaling the pseudo wires.

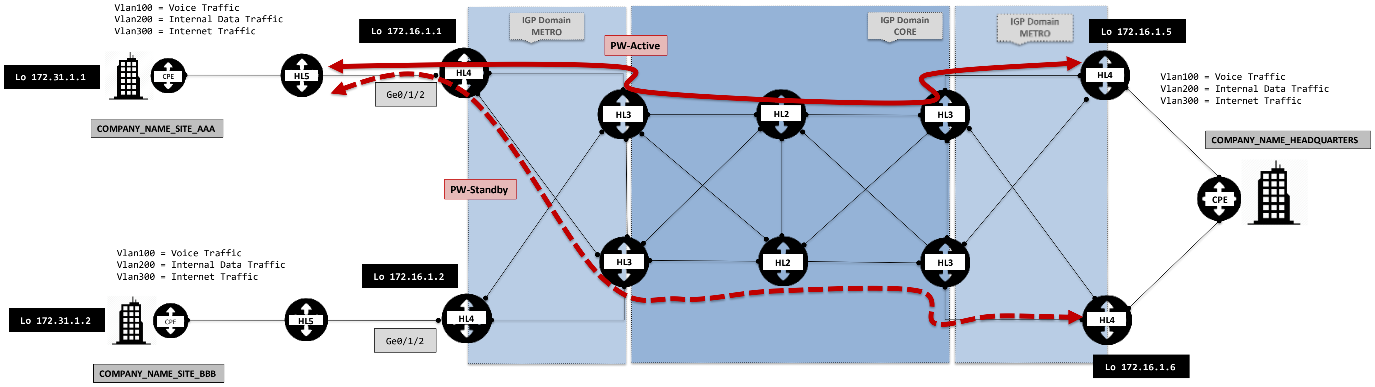


Figure 16. Generic network topology for L2VPN Point-to-Point VLL Enterprise Service

To deploy these set of services several conditions should be evaluated by means of a flow diagram like the following:

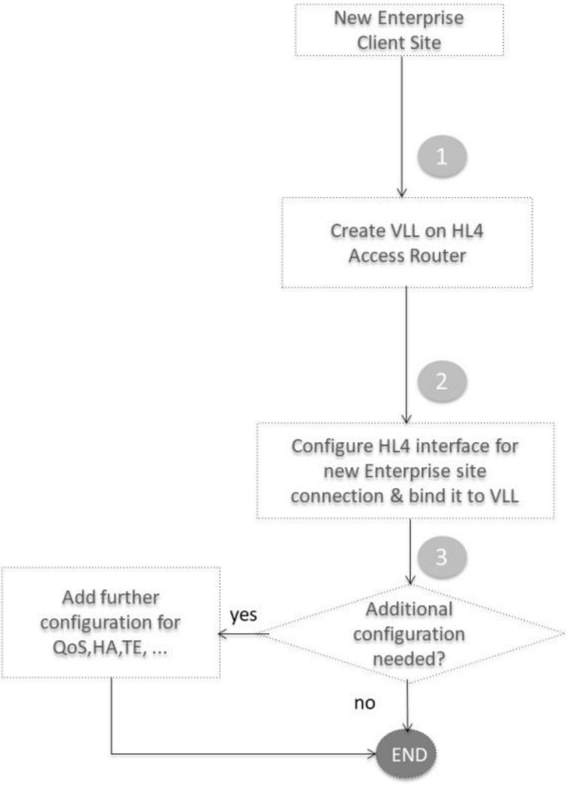

New Enterprise Client Site
1
Create VLL on HL4 Access Router
2
Configure HL4 interface for new Enterprise site connection & bind it to VLL
3
Additional configuration needed?
yes
Add further configuration for QoS,HA,TE, ...
no
END

Figure 17. Configuration process flow for L2VPN Point-to-Point VPLS VPLS Enterprise Service

1 Create VLL on HL4 Access Router receiving the client CPE

2 Continue to configure and bind HL4 client interface to VLL

3 Do the L2VPN need additional configuration ?

- ***Yes.*** *Applicable new parameters for QoS, HA should be included in the configuration*
- ***No.*** *End*

6.3.4 Use case 1.7: EVPN based L2VPN Service Provisioning

6.3.4.1 Definition

ID	**1.7**
Name	**EVPN based L2VPN Service Provisioning**
Brief description	7.2.10 Ethernet VPN (EVPN) is next generation solution that provide Ethernet multipoint services including Data Centre Interconnect (DCI) and carrier Ethernet VPLS/VLL services. EVPNs utilize the BGP control plane infrastructure while it allows for different flavours depending of the encapsulation utilized in the data plane VXLAN/MPLS/PBB.
Feature Bundles required	BGP_AND_PBB_MPLS_BASED_EVPN_MANAGEMENT

Table 12 Definition of Use Case 1.7

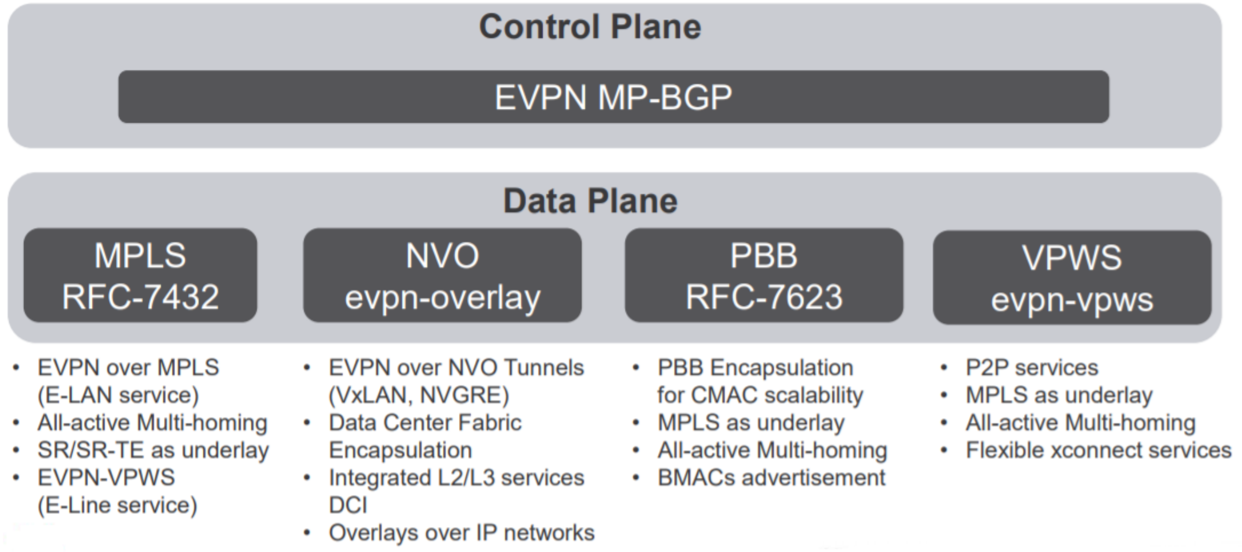
6.3.4.2 Description

EVPN [RFC 7432] stands for Ethernet VPN, which is considered as an alternative to the common implementations used for deploying L2VPN services such as VPLS. EVPN is an evolved technology more efficient than VPLS since by using extended BGP reachability information. Thus, EVPN implements MAC address learning and advertisement between Layer 2 networks at different sites on the control plane instead of on the data plane.

VPLS usually have issues regarding lack of support of load balancing and high network resource usage, but EVPN overcome the preceding problems due to the following features:

- Usage of extended BGP to implement MAC learning and advertisement on control plane.
- Not requires a fully meshed backbone between PEs.
- PEs can use ARP lo learn local MAC addresses and use MAC address advertisement routes to learn remote MAC addresses and IP addresses corresponding to these MAC addresses and store them locally.

While the control plane is based on MP-BGP enhancements, the EVPN data plane allows for different solutions or flavours depending on the use case and application scenario, as depicted below



- EVPN over MPLS (E-LAN service)
- All-active Multi-homing
- SR/SR-TE as underlay
- EVPN-VPWS (E-Line service)

- EVPN over NVO Tunnels (VxLAN, NVGRE)
- Data Center Fabric Encapsulation
- Integrated L2/L3 services DCI
- Overlays over IP networks

- PBB Encapsulation for CMAC scalability
- MPLS as underlay
- All-active Multi-homing
- BMACs advertisement

- P2P services
- MPLS as underlay
- All-active Multi-homing
- Flexible xconnect services

Figure 18. EVPN framework

7

Inventory Support (Category 3)

Traffic engineering (TE) allows the enforcement of traffic steering flows by leveraging onto MPLS tunnels or Segment Routing paths ...

7 Category 3: Inventory Support

7.1 Structure and Classification

The set of uses cases related to inventory support on the SBI interface apply to recover information and state about hardware components of NE and the logical configuration of the device.

ID	Use case Title	Section
2.1	Retrieve HW [Physical] Inventory	7.3.1
2.2	Retrieve logical [Interfaces] Inventory	7.3.2

Table 13 Inventory Use Cases covered in

7.2 Atomic Operations

On the other hand, the following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing the inventory support use cases. A deeper level detail of the operations (to be included in the MUST Test plan) would explain how to execute each specific function (RPC calls and XML).

ID	Operation	Description
Network Inventory - 1	Retrieve all device component state and config data from a given device.	Retrieve all the openconfig-components model configuration and state data of an specific device.
Network Inventory -2	Retrieve logical interfaces configuration from one node	Retrieve all the parameters related to interface configuration (description, IP addressing, vlan, etc.) of a node. In this operation information about all the interfaces is received.

Table 14 Network Inventory Atomic Operations

7.3 Use Cases

7.3.1 Use case 2.1 Retrieve HW [Physical] Inventory

7.3.1.1 Definition

ID	**2.1**
Name	**Retrieve HW [Physical] Inventory**
Brief description	Representation of a component's hierarchy using the YANG model openconfig-platform
Feature Bundles required	DEVICE_INTERNAL_COMPONENTS_MANAGEMENT HARDWARE_PERFORMANCE_STATISTICS

Table 15 Definition of Use Case 2.1

7.3.1.2 Description

The HW components hierarchy follows the next diagram:

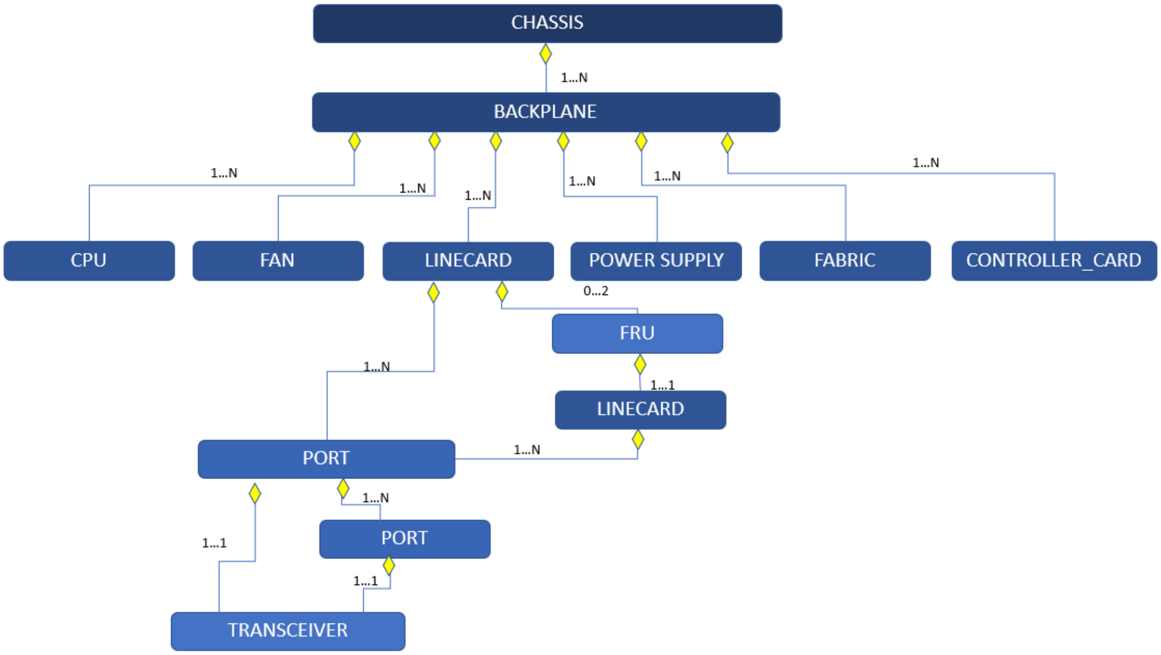


Figure 19. HW Hierarchy for a network device.

In this architecture, every blue box represents a COMPONENT of the openconfig/platform model. The description of the above components is the following:

- **CHASSIS**: is the top level component and can be understood as the main device.
- **BACKPLANE**: medium over which all the components of a device can communicate. In a cluster or stacking environment, more than one backplane can be associated to a single top device (for example, in a stacking scenario).
- **CPU**: processing unit, e.g., a management processor.
- **FAN**: cooling fan or some other heat-reduction component.
- **POWER_SUPPLY**: component that is supplying power to the device.
- **FABRIC**: interconnect between ingress and egress ports on the device (e.g., a crossbar switch).
- **CONTROLLER_CARD**: a type of card whose primary role is management or control rather than data forwarding
- **LINECARD**: component typically inserted into a chassis slot that provides interfaces to the network.
- **FRU**: replaceable hardware component. Some linecards can consist on a motherboard and a flexible subcard that is inserted into the motherboard. Those subcards are modeled in openconfig/platform yang as a FRU component.
- **PORT**: physical port, e.g., for attaching pluggables and networking cables. If the port mode is breakout, the component is composed of N ports (example: QSFP).
- **TRANSCEIVER**: pluggable module present in a port.

In case of not being able to fit with the above architecture, to establish the relationship between different levels of the hardware inventory hierarchy, a minimum set of three parameters that MUST be present for each component:

- **status/location:** refers to the **absolute location** of the component within the device hierarchy.
- **subcomponent/name:** refers to the name of each component that is contained within a upper component. For example, a linecard component

may have a list of references to port components that reside on the linecard.

- **status/parent:** refers to the name of the inmediately higher parent component

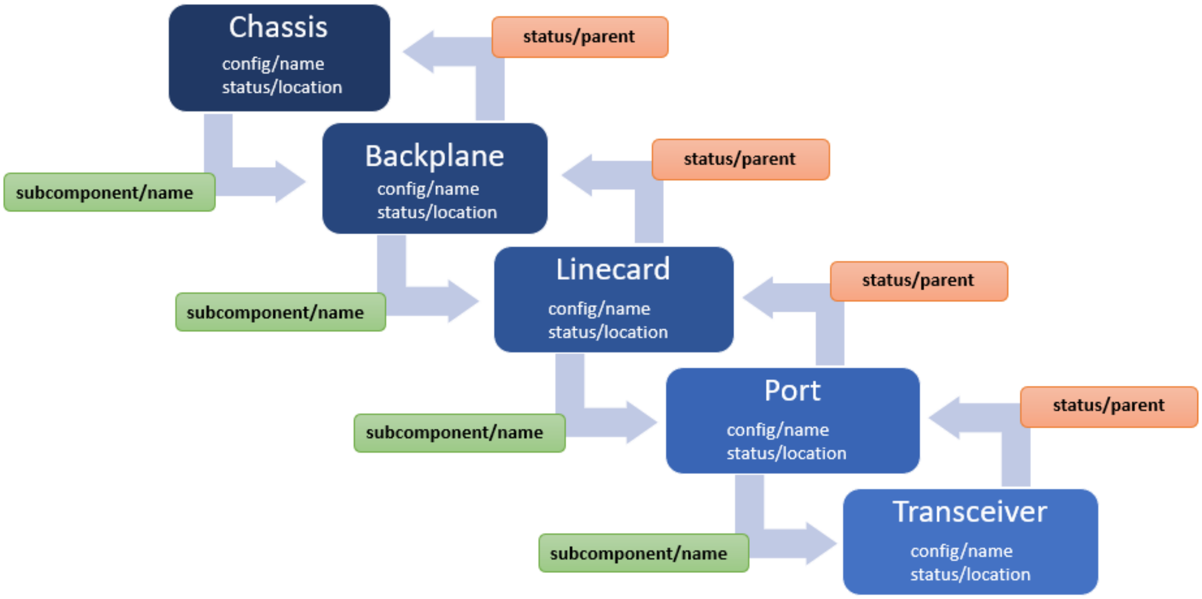


Figure 20. HW inventory hierarchical relationship.

A concrete example is shown in the next figure.

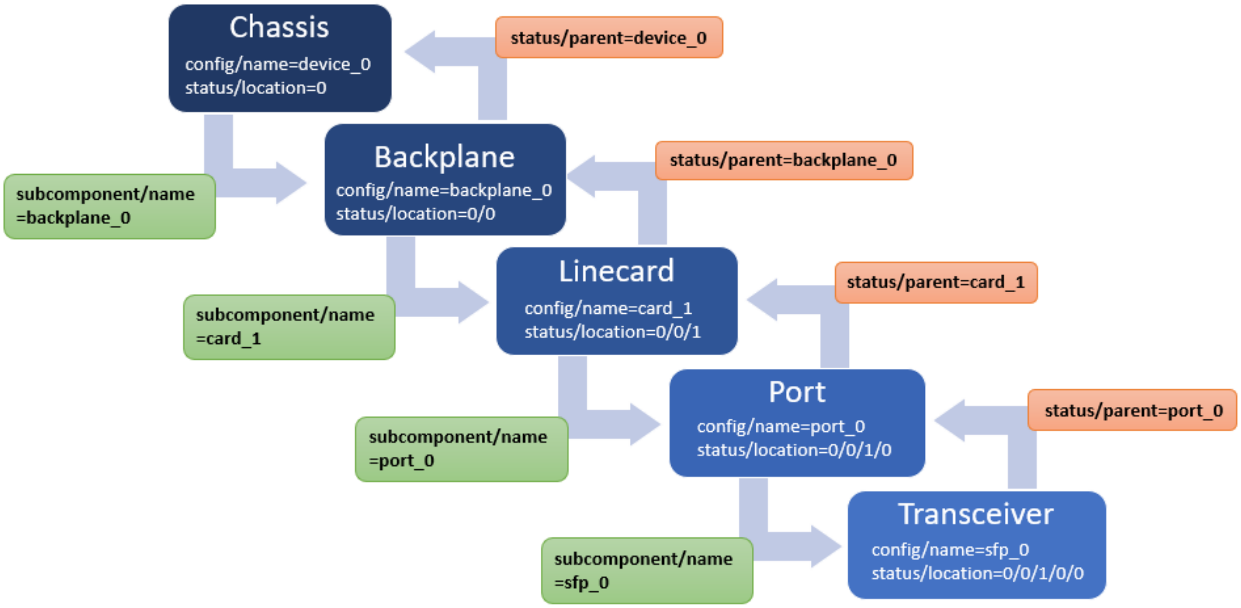


Figure 21. HW inventory hierarchical relationship example.

A device is modeled in OC by using a set of modules around **"openconfig-platform"** Yang model that are used to model specific parameters of the device subcomponents.

- **OpenConfig-platform:** This model performs a device modelling in terms of its hardware components.
- **OpenConfig-platform-linecard:** This model is used to augment the openconfig-platform model adding several parameters for a card subcomponent.
- **OpenConfig-platform-port:** This model is used to augment the openconfig-platform model adding several parameters for a port subcomponent.
- **OpenConfig-platform-transceiver**: This model is used to augment the openconfig-platform model adding several parameters for a transceiver subcomponent.
- **OpenConfig-platform-psu**: This model is used to augment the openconfig-platform model adding several parameters for a power supply subcomponent.

- **OpenConfig-platfom-cpu**: This model is used to augment the openconfig-platform model adding several parameters for a CPU subcomponent.
- **OpenConfig-platform-fan**: This model is used to augment the openconfig-platform model adding several parameters for a fan subcomponent.

All components and subcomponents that comprise a device regardless of the type CHASIS, FABRIC, BACKPLANE, LINECARD, PORT, TRANSCEIVER, FAN, CPU, or POWER-SUPPLY shall make use of the same config and state parameters so as to describe them in addition to those specific parameters related to the component under consideration.

7.3.2 Use case 2.2: Retrieve Logical Interfaces Inventory

7.3.2.1 Definition

ID	**2.2**
Name	**Retrieve logical interfaces inventory**
Brief description	The use case focuses on retrieving the information regarding all the logical or virtual interfaces of a specific equipment such as subinterfaces, vlan interfaces, tunnel interfaces and other non-physical interfaces.
Feature Bundles required	INTERFACE_MANAGEMENT

Table 16 Definition of Use Case 2.2

7.3.2.2 Description

This use case aims at inventorying logical resources and associated parameters of a previously commissioned NE through base OpenConfig SBI <get> operation. Specifically, parameters concerning logical interfaces inventory would look like those used configuring interfaces to access to VPN, as explained in section 5.2.3.

8

Topology and Discovery Use Cases (Category 4)

A set of abstractions have been defined in order to represent several views of the network topology. However, to gather this information and represent it correctly, the SDN controller must be able to collect the required information from the Network Devices. Thus, in this chapter all the SDN-Controller SBI...

8 Category 4: Topology and Discovery Use Cases

8.1 Structure and Classification

A set of abstractions have been defined in order to represent several views of the network topology. However, to gather this information and represent it correctly, the SDN controller must be able to collect the required information from the Network Devices. Thus, in this chapter all the SDN-Controller SBI requirements are defined to collect and export the network topology information.

The set of topology views defined at the controller level are:

- **UNI-Topology**: This layer should contain all the client/user side ports (used and unused) of the network elements.
- **Layer 1 Topology:** This layer contains the physical and virtual devices of the network and the interconnection between the IP and Optical domain.
- **Layer 2 Topology:** This level should represent the Link-Layer connection between network nodes in NNI side. Even though the Layer 2 network topology should be generic and applicable to Layer 2 networks built with different L2 technologies, it will be mainly used to describe both the physical and the logical (virtual) Ethernet network topologies. Accordingly, parameters of this Layer are the following:

    - MAC address information
    - Port Speed/duplex configuration
    - Port administrative state
    - Maximum Transmission Unit (MTU)
    - Link Aggregation (LAG)
    - VLAN ID (single and double tagging)

  - Pseudowires (PW) and Pseudowire terminations
  - LLDP Relationships between nodes.

- **Layer 3 Topology:** The L3 layer represents the Network at IP layer. As in the L2, at L3 the following information can be found:

  - Links IP address in both sides
  - Router-id
  - Link IGP metric
  - Maximum Bandwidth / Available Bandwidth
  - Latency

Inside this layer several protocols views can be generated. For example, IGP-topology or BGP topology.

In addition to pure network topology, logical topologies related to BGP, Multicast or Service/LSP to be exported to SDN Controller are also covered in this section.

The set of uses cases related to topology support considered on the SBI interface in this deliverable are listed in the following table:

ID	Use case Title	Section
3.1	Obtain and export of end to end ip topology using ip domain controllers (IGP Topology)	8.2
3.2	Obtain and export of L2 topology using ip domain controllers (ethernet links between routers)	8.2.2
3.3	Export potential service end points in IP topology (UNI Topology)	8.2.3

8.2 Use Cases

8.2.1 Use case 3.1 Obtain and export of end to end IP topology using IP domain controllers (IGP Topology)

8.2.1.1 Definition

Number	**3.1**
Name	**Obtain and export of end to end IP topology using IP domain controllers (IGP Topology)**
Brief description	L3 topology must represent the IP links in the network, including specific parameters to support IP Addressing, Metrics and IGP information. The L3 (nodes and Links) are supported in L2 topology (nodes and links).
Feature Bundles required	INTERFACE_MANAGEMENT BASIC_L3VPN_MANAGEMENT CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT L3VPN_CE_PE_ROUTING

Table 17 Definition of Use Case 3.1

8.2.1.2 Description

The L3 topology information can be used by different OSS/BSS applications to draw the network topology and relate the network services with the lower layers. Moreover, in the case of the deployment of a central controller like PCE in the network, the L3 Topology information is fundamental to calculate end-to-end optimal paths. The topology information can be updated dynamically as the network changes occur i.e., a node or link goes down. However, this update process is out of the scope of the current specification.

The L3 topology in this use case must represent the IP links in the network, including specific parameters to support IP Addressing, Metrics and IGP information. The L3 (nodes and Links) are supported by L2 topol8.2ogy (nodes and links).

The L3 Network information MUST be gathered from the network by the SDN domain controller. This section presents the details of how the IP SDN domain controllers MUST get L3 Topology related information from the network devices and be able to build the NBI abstract representation.

8.2.1.3 L3 Network Topology Discovery

There are several options to obtain the end-to-end topology information. One way is by peering passively with an IGP node to get all the link state information, with all its drawbacks:

- A practical PCE or SDN controller code needs to support both OSPF and IS-IS.
- IGP tends to send many updates, so the controller will spend some time processing all those messages.
- In the cases where the network consists of multiple IGP domains across geographic areas then it could be a challenging on where to place the central controller which peers with IGP nodes.

The target and right alternative approach is to use BGP-LS protocol (RFC 7752) which creates a new NLRI carrying all the IGP info over BGP. In this approach we can leverage a BGP speaker that is already participating in the IGP, thus retrieving info from IGP LSDBs and distributing it to a PCE or SDN controller. The BGP speaker can apply any filter before sending the info northbound to the controller. Accordingly, compared with the IGP peering, the advantages of this approach would be:

- Controller implementation must only support BGP.
- BGP tends to produce less updates messages.
- In a network with multiple IGP domains, extending peering over BGP is much more feasible compared to IGP.

Therefore, by establishing a BGP-LS peering with one selected router (or more for redundancy) PCE or SDN controller, the topology information needed can be obtained from the real network and used it for LSP path computation or to visualize the network topology, respectively. This use case will be only focused on how SDN controller can get L3 IP network topology related information to be stored and exported over its NBI to third parties (SDN hierarchical levels or OSS/BSS).

Since IGP consists of topology and IP reachability information and if we want to reconstruct an IGP Topology view at the controller based on the data received over BGP-LS, then BGP-LS must have some way to represent Topology and IP reachability information in its database. For that, BGP-LS specification contains two parts:

- Definition of a new BGP NLRI type which is essentially sets of TLV's that define three objects:
    1. Nodes
    2. Links
    3. IP Prefixes

  So, with the combination of Node and Link objects one can construct a topology info and IP Prefix object will provide IP reachability information.

- Definition of a new BGP path attribute (BGP-LS attribute) which is optional non-transitive attribute. It encodes the properties of the objects (link, node and prefix). For instance, it could be Node-names, IGP metric, TE-metric, latency, Available BW etc.

For this specific use case, the expected L3 network topology information collected by SDN Domain controller through BGP-LS to MUST include:

1. Router-id
2. Link IP addresses in both sides
3. Link IGP metric
4. Maximum Bandwidth / Available Bandwidth
5. Link latency

A link described by the Link descriptor TLVs actually is a "half-link", a unidirectional representation of a logical link. In order to fully describe a single logical link, two originating routers advertise a half-link each, i.e. two link NLRIs are advertised for a given point-to-point link. As regards IGP further information such as IP prefixes advertised by each node, area distribution and routers roles are not represented in the topology for the current version of the document (to be discussed).

8.2.2 Use case 3.2 Obtain and export of L2 topology using IP domain controllers (ethernet links between routers)

This section presents the details of how the Network Controller MUST gather the information from the SBI to export the L2 topology (ethernet links between routers).

8.2.2.1 Definition

Number	**3.2**
Name	**Obtain and export of L2 topology using IP domain controllers (ethernet links between routers)**
Brief description	This topology stored the parameters to support L2 Ethernet network topology representation including nodes, termination points (ports and interfaces) as well as links and theirs operational state.
Feature Bundles required	LLDP_MANAGEMENT

Table 18 Definition of Use Case 3.2

8.2.2.2 L2 Network Topology Discovery

The L2 Network information MUST be gathered from the network by the SDN domain controller using LLDP. LLDP is a Layer 2 protocol that allows a network device to advertise its identity and capabilities on the local network providing topological information. The protocol is defined in the IEEE standard 802.1AB.

It suports the usage of basic and custom TLVs (type-length-values). A TLV is the basic information unit of an LLDP-Data Unit (DU). TLVs that can be encapsulated into an LLDPDU include basic TLVs, TLVs defined by IEEE 802.1 and TLVs defined by IEEE 802.3. The table below details each type of TLV where those labeled as non-mandatory could be suppressed, so that only TLVs needed for representing L2 topology will be advertised with the LLDP PDUs.

TLV Category	TLV Name	TLV Type Value	Description	Mandatory
BASIC	End of LLDPDU	0	End of an LLDPDU.	Yes
BASIC	Chassis ID	1	Bridge MAC address of the transmit device.	Yes
BASIC	Port ID	2	Number of a transmit interface of a device.	Yes
BASIC	Time To Live	3	Timeout period of local device information on neighboring devices.	Yes
BASIC	Port Description	4	String describing an Ethernet interface.	No
BASIC	System Name	5	Device name.	No
BASIC	System Description	6	System description.	No
BASIC	System Capabilities	7	Primary functions of the system and whether these primary functions are enabled.	No
BASIC	Management Address	8	Management address.	No
BASIC	Reserved	9-126	Reserved for special use.	No
Defined by IEEE 802.1	Reserved	0	Reserved for special use.	No
Defined by IEEE 802.1	Port VLAN ID	1	VLAN ID on an interface.	No
Defined by IEEE 802.1	Port And Protocol VLAN ID	2	Protocol VLAN ID on an interface.	No

Defined by IEEE 802.1	VLAN Name	3	VLAN name on an interface.	No
Defined by IEEE 802.1	Protocol Identity	4	Protocol type that an interface supports.	No
Defined by IEEE 802.3	Reserved	0	Reserved for special use.	No
Defined by IEEE 802.3	MAC/PHY Configuration/Status	1	Duplex and bit-rate capability, current duplex and bit-rate settings, and auto-negotiation status.	No
Defined by IEEE 802.3	Power Via MDI	2	Power supply capability of an interface, that is, whether an interface supports PoE and whether an interface supplies or requires power.	No
Defined by IEEE 802.3	Link Aggregation	3	Link aggregation status.	No
Defined by IEEE 802.3	Maximum Frame Size	4	Maximum frame length supported by interfaces. The maximum transmission unit (MTU) of an interface is used.	No
Defined by IEEE 802.3	Reserved	5-255	Reserved for special	No

Table 19 LLDP TLVs for Use Case 6.2

NOTE: For security reasons, in order to use LLDP, it has to be enabled only in those interfaces strictly necessary. A generic recommendation is to enable it on NNI interfaces of the Core and Aggregation routers, but never on UNI interfaces where client CPE are connected. Also, LLDP for Media Endpoint Devices (LLDP-MED) must be disabled in all interfaces of the whole network.

This information can be collected from the network by SDN Controllers through NETCONF/Yang using the "OpenConfig-LLDP" data model Configuration and state variables are defined in this module to report generic protocol information and the per interface neighbors and status. Note that the model allows to modify parameters like chassis-id or system-description.

As it is depicted in the following example, when the basic and one optional TLVs are enabled between the **NODE A** and **NODE B**:

- System Name (Optional)
- Chassis ID
- Port ID
- TTL

the SDN controller, based on the information above, could find out and represent the different links between the nodes and create the whole L2 topology.

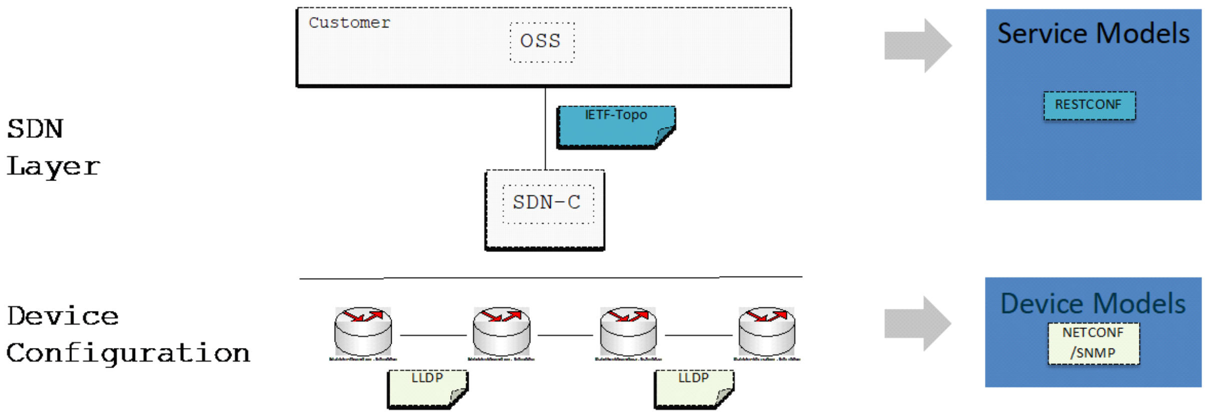


Figure 22. L2/L3 topology collection architecture

Finally, the following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case. A deeper level detail of the operations (to be included in the associated MUST test plans) would explain how to execute each specific function (RPC calls and XML).

ID	Operation	Description

Topology - 1	Retrieve interfaces in a node on which LLDP is enabled	Obtain information about which ports of a node have LLDP protocol enabled
Topology - 2	Retrieve L2 neighbor s of a node in all ports	Obtain information about L2 neighbor s of a node for all its ports
Topology - 3	Retrieve L2 neighbor s of a node in a specific port	Obtain information about L2 neighbor s of a node for a specific ports
Topology - 4	Enable LLDP in a interface	Activate LLDP protocol in a interface given the name of the interface
Topology - 5	Modify LLDP parameters	Edit global LLDP configuration on a device

Table 20 Atomic Operations for Topology Use case 6.3.2

8.2.3 Use case 3.3 UNI-Topology

In this section, the details to obtain and export potential end points in the network are the following:

8.2.3.1 Definition

Number	**3.3**
Name	**Export potential service end points in IP topology (UNI Topology)**
Brief description	The UNI topology represents the feasibility topology and has all the potential end point in the network.
Feature Bundles	INTERFACE_MANAGEMENT CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT

Table 21 Definition of Use Case 3.3

8.2.3.2 UNI Topology Discovery

The aim of this use case is to determinate a procedure to obtain those interfaces available in the NE that could be used by the user as UNI ports to be bound to LxVPN services. Such information should be exposed to the SDN Controller.

One way of doing so might be through a NETCONF get operation filtering those interfaces that not had MPLS activated and export them to SDN controller. It would mean that are susceptible to be utilized as UNI ports.

/network-instances/network-instance/mpls/global/interface-attributes/interface/config/**mpls-enabled=FALSE**

In any case the vendor is requested to come up with any alternative proposal or enhancement that address the use case objective.

The following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case. A deeper level detail of the operations (o be included in the associated MUST test plans) would explain how to execute each specific function (RPC calls and XML).

ID	Operation	Description
Topology - 6	Retrieve UNI interfaces of a node	Obtain information about which ports of a node have MPLS enabled and therefore can b considered as UNI ports.

Table 22 Atomic Operations for Use Case 6.3

9

Traffic Engineering Use cases (category 5)

Traffic engineering (TE) allows the enforcement of traffic steering flows by leveraging onto MPLS tunnels or Segment Routing paths ...

9 Category 5: Traffic Engineering Use Cases

9.1 TE and PCE for SBI

Traffic engineering (TE) allows the enforcement of traffic steering flows by leveraging onto MPLS tunnels or Segment Routing paths. This permits to increase the efficiency on the use of the network resources by properly mapping the traffic flows to the available resources, and improve network management, including troubleshooting, to overcome difficult failure situations. Increasingly complex network scenarios such as large single domain environments, multi-domain or multi-layer networks require the usage of algorithms for efficiently computing end-to-end paths.

This complexity is driving the need for a dedicated SDN controller, which will perform path computations and be adaptive to network changes Currently, only online and real-time constraint-based routing path computation is provided in an MPLS RSVP-TE or Segment Routing networks. Each router performs constraint-based routing calculations independent of the other routers in the network. These calculations are based on currently available topology information, information that is usually recent, but not completely accurate. LSP placements are locally optimized, based on current network status.

On the contrary, a PCE has a global view of the bandwidth demand in the network and maintains a traffic-engineered database to perform path computations. In order to perform the computation of a path for a TE LSP the PCE shall gather the required information (including network topology and resource information) by participating in routing protocols updates as any other

NE, usually through BGP-LS as covered in **RFC 8571**. Besides, it even could perform statistics collection from all the routers in the MPLS domain using SNMP, NETCONF or other methods, which also would provide a mechanism for offline control of the TE LSPs.

Technically speaking, a Path Computation Element (PCE) is an entity (component, application, or network node) that is capable of computing a network path or route based on a network graph and applying computational constraints. A Path Computation Client (PCC) is any client application requesting a path computation to be performed by a PCE. The Path Computation Element Protocol (PCEP) enables communications between a PCC and a PCE, or between two PCEs, as describe in figure below.

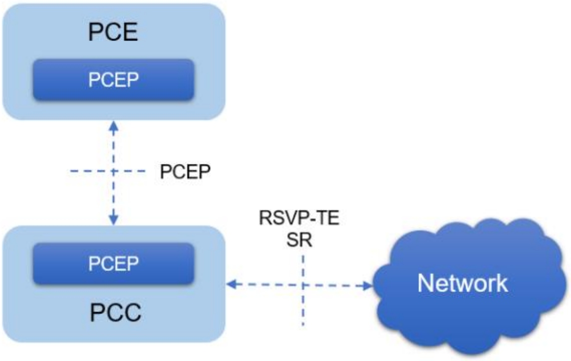


Figure 23. PCEP Architecture

<u>Note:</u> PCE is covered by IETF in different documents, the main are:

- **RFC 4655** “A Path Computation Element (PCE)-Based Architecture”
- **RFC 5440** “Path Computation Element (PCE) Communication Protocol (PCEP)”
- **RFC 8231** “Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE”
- **RFC 8281** “Path Computation Element Communication Protocol (PCEP) Extensions for PCE-Initiated LSP Setup in a Stateful PCE Model”

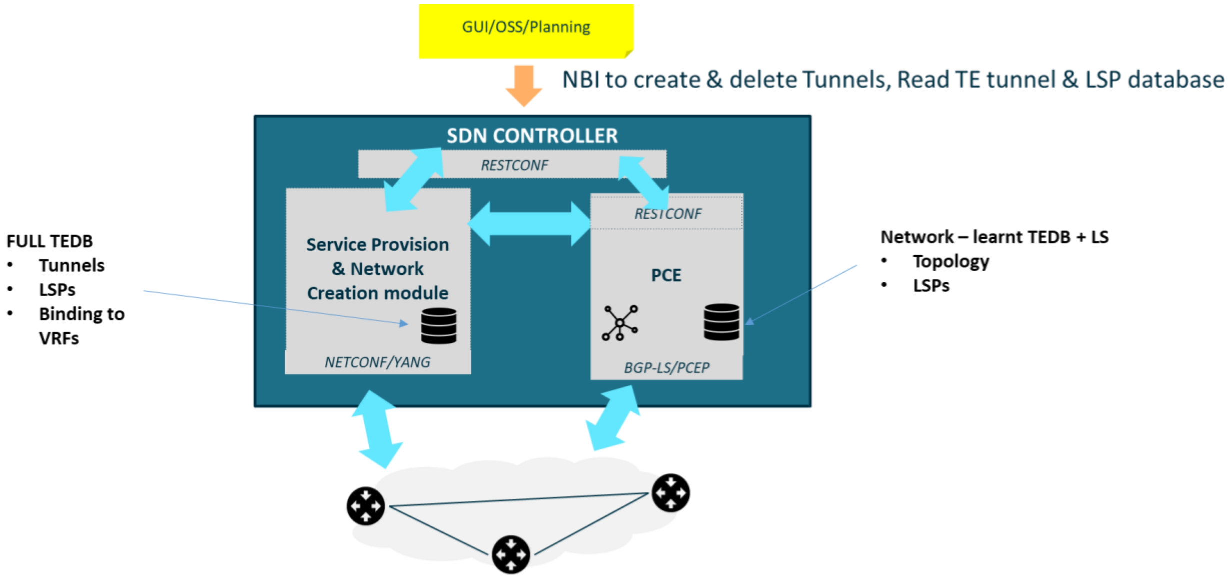

GUI/OSS/Planning
NBI to create & delete Tunnels, Read TE tunnel & LSP database
SDN CONTROLLER
RESTCONF
RESTCONF
Service Provision & Network Creation module
PCE
NETCONF/YANG
BGP-LS/PCEP
FULL TEDB
• Tunnels
• LSPs
• Binding to VRFs
Network – learnt TEDB + LS
• Topology
• LSPs

Figure 24. PCE function in the E2E IP SDN Domain Controller

The PCC initiates the PCEP session and stays connected to the PCE for the duration of the PCEP session. During the PCEP session, the PCC requests LSP parameters from the stateful PCE. On receiving one or more LSP parameters from the PCE, the PCC re-signals the TE LSP.

A PCE can either be stateful or stateless. A stateful PCE maintains strict synchronization between the PCE and network states (in terms of topology and resource information), along with the set of computed paths and reserved resources in use in the network. A stateless PCE does not remember any computed path, and each set of requests is processed independently of each other. For this specification only active stateful PCE model are covered, where PCE learns about the PCC LSP states and actively modifies the PCC LSPs.

Thus, these are the functions that a PCC-PCE system using PCEP shall include:

- **LSP state synchronization**. A PCC takes a snapshot of the state of its LSPs and sends a state report containing the snapshot to the PCE. This reporting activity is done for each LSP. PCE would only send a notification back to the PCE if any LSP can not be synchronized. Either the PCE or the PCC may terminate the session using the PCEP.
- **LSP delegation**. After the PCE and the PCC have indicated that they support LSP Update, then the PCC may choose to grant the PCE a temporary right to update attributes from one or more LSPs. PCE can return the LSP delegation at any time if it no longer wishes to update the LSP's state and also, PCE may reject a delegation based on a local policy.
- **LSP operation**. After LSPs have been delegated to the PCE, the PCE can modify LSP parameters of delegated LSPs. To update an LSP, the PCE sends a request to the PCC with the parameters needed to build the LSP. When PCC has built the LSP, PCC sends a report message to the PCE indicating the result of the operation (whether the LSP is build and up or down). Note that PCC will send the operation state report to each stateful PCE that is connected to it.

From an overall NBI architectural point of view, the PCE module is part of the E2E IP MUST Domain controller (see Figure 26).

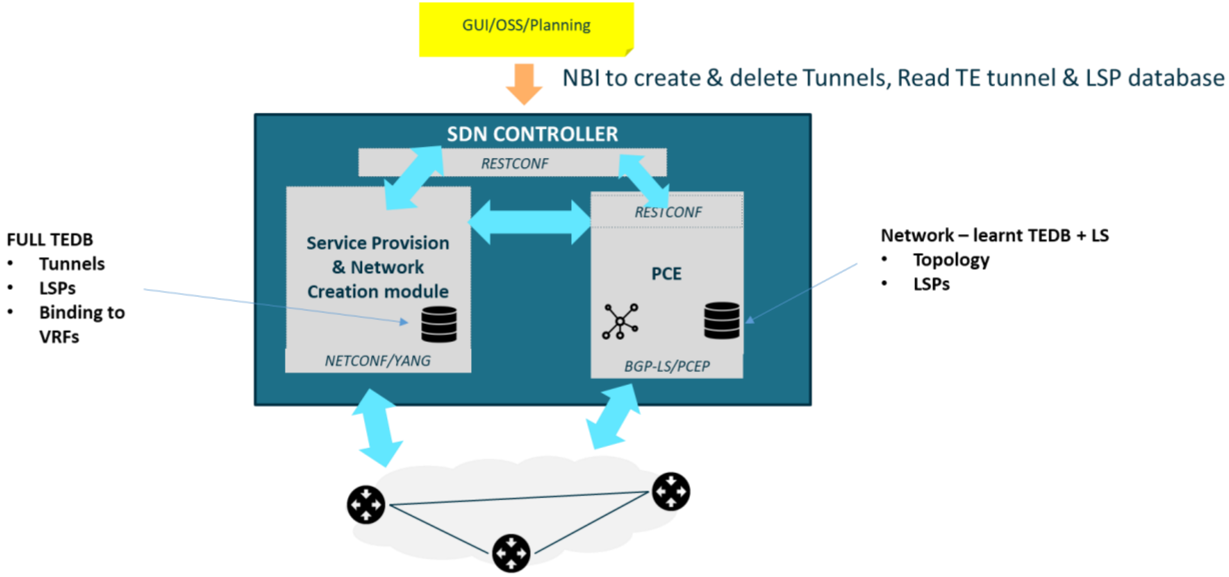

GUI/OSS/Planning
NBI to create & delete Tunnels, Read TE tunnel & LSP database
SDN CONTROLLER
RESTCONF
Service Provision & Network Creation module
NETCONF/YANG
RESTCONF
PCE
BGP-LS/PCEP
FULL TEDB
• Tunnels
• LSPs
• Binding to VRFs
Network – learnt TEDB + LS
• Topology
• LSPs

Figure 25. PCE function in the E2E IP MUST SDN Domain Controller

Depending on the entity which iniciated the tunnel two scenarios for LSP creation and management shall be considered for this specification.

- **PCC-initiated LSPs**. The PCC delegates the PCC-initiated LSPs to the main PCE for external path computation. The PCC informs the PCE about the configured parameters of a PCE-controlled LSP, such as bandwidth, ERO, and priorities. It also informs the PCE about the actual values used for these parameters to set up the LSP including the RRO, when available.

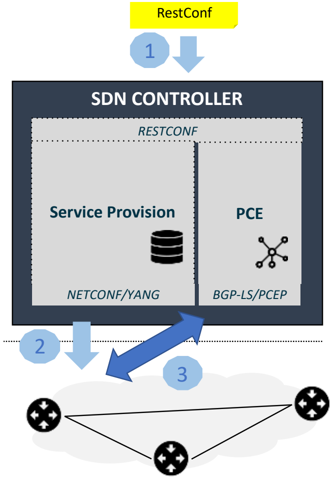


Figure 26. PCC-initiated LSP workflow

- **PCE-initiated LSPs**. A PCE-initiated LSP is dynamically created by an external PCE; as a result, there is no LSP configuration present on the PCC. The PCC creates the PCE-initiated LSP using the parameters provided by the PCE, and automatically delegates the LSP to the PCE.

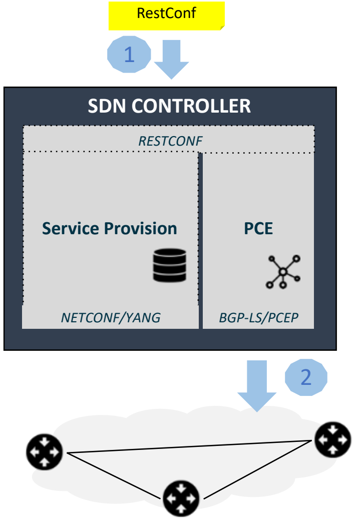


Figure 27. PCE-initiated LSP workflow

9.2 Structure and Classification

Based on previously described functionalities, the main purpose of traffic engineering use cases in the MUST Project is to reduce overall operating costs through more efficient use of network resources, including link occupation, traffic rerouting, network availability, and other components. Essentially, traffic engineering actions address the need to prevent situations where some parts of the network are overloaded, while other parts of the network remain underused and thus ensure the most appropriate path for network traffic and allow the implementation of mechanisms for protecting traffic against network failures.

The set of uses cases related to TE support considered on the SBI interface are listed in the following table.

ID	Use case Title	Section
4.1	LSP Creation, modify and delete with RSVP-TE	9.4.1
4.2	LSP create, modify and delete with constraints (delay, bandwidth and hop count) Signaling: RSVP	9.4.2
4.3	LSP create, modify and delete with constraints (delay, bandwidth and hop count) Signaling: SR	9.4.3
4.4	LSP create, modify and delete with constraints (delay, bandwidth and hop count) Signaling: Explicit Path	9.4.4
4.5	LSP create, modify and delete with constraints (delay, bandwidth and hop count) Protection: Redundancy 1+1	9.4.5

This set of use cases addresses different situations or options for establishing an LSP, from the simplest with no constraint at all, only taking into account routing optimal path or setting the path explicitly to the more comprehensive which allow to specify a number of constraints such as delay, bandwidth or hop limit for LSPs, using different types of protection approaches. These constraints will be processed by PCE that will return attributes needed to establish LSP, as explained before. All of this, using RSVP-TE or SR at the signaling plane. Finally, advanced uses cases for LSPs TE optimization and enhancement are covered.

9.3 OpenConfig Model for TE

The OpenConfig model used for LSP configuration in the SBI will be "openconfig-mpls". This module is imported into "network-instance" module at the Xpath: /oc-netinst:network-instances/oc-netinst:network-instance/mpls. It's not a standalone root in the hierarchy. Accordingly, each LSP configured will be bounded to the L2VPN or L3VPN instance in which the LSP tunnel has been created. However, it is also possible to create a generic LSP not implicitly associated with a specific network-instance. In order to do that, OpenConfig enables the use of "DEFAULT_INSTANCE"[1] option in order to handle LSPs decoupled from any network instance.

This "openconfig-mpls" module provides data definitions for configuration of Multiprotocol Label Switching (MPLS) and associated protocols for signaling and traffic engineering. It consists on several modules and submodules as shown below. The top-level MPLS module describes the overall framework where the following types of LSPs are supported:

- Traffic-engineered (or constrained-path)
- IGP-congruent (LSPs that follow the IGP path)
- Static LSPs which are not signaled

The structure of each of these LSP configurations is defined in corresponding submodules as depicted below:

[1] The value "DEFAULT_INSTANCE" can be established to the parameter /network-instances/network-instance/config/type

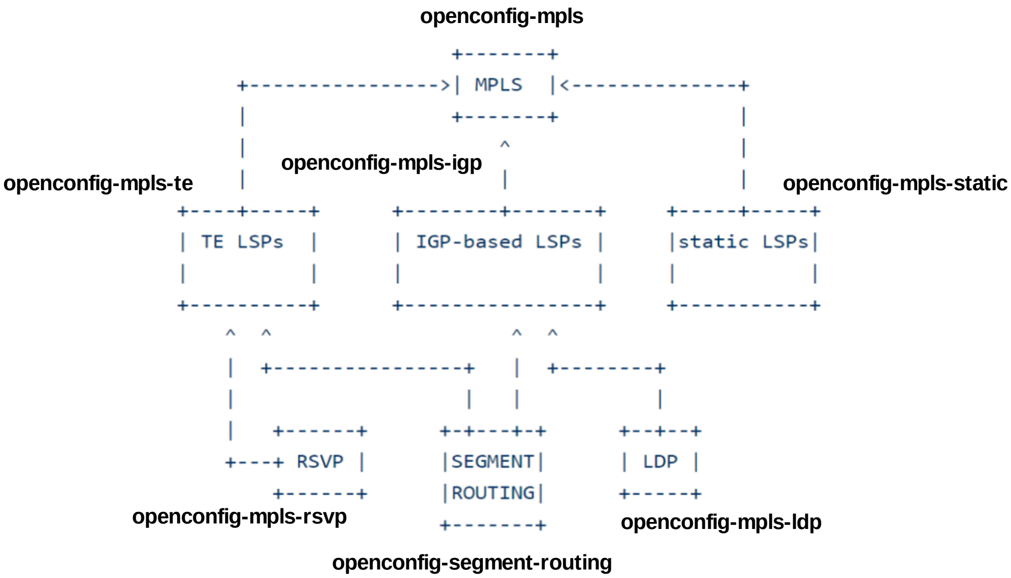


Figure 28. OC Modules for MPLS

As noted above, for TE LSP creation both RSVP or SR modules can be used, whereas for IGP-based unconstraint LSP, LDP as well as SR modules are considered. The TIP Traffic Engineering SBI specification considers TE-LSPs which use either RSVP-TE or SR as the signaling protocol for all the cases described in document (even for the use case of LSPs with no constraints at all or with an explicit path).

9.4 Use Cases

This section presents the details of how to manage LSP establishment and TE information on SBI between NE (routers) and the IP SDN domain controller, which includes a PCE module.

The set of parameters required for the creation of all types of LSP for the use cases explained in this chapter on Netconf SBI are taken from "**openconfig-network-instance**" module that import the MPLS module "**openconfig-mpls**" as a container when network-instance type = DEFAULT_INSTANCE (Global Table), as stated in previous section 9.3.

9.4.1 Use case 4.1 - LSP Creation, modify and delete with RSVP-TE

9.4.1.1 Use Case Definition

Number	**4.1**
Name	**LSP Creation, modify and delete with RSVP-TE**
Brief description	This use case expects the creation of LSPs without any type of constrain calculation or protection. The LSP generally follows the shortest path as dictated by the local routing table, usually taking the same path as destination-based, best-effort traffic. These paths automatically reroute themselves whenever a change or link occurs in a routing table or in the status of a node or link. The LSP won't have any kind of protection either.
Feature Bundles required	BASIC_TRAFFIC_ENGINEERING

Table 23 Definition of Use Case 4.1

9.4.1.2 Description

This use case would be the easiest where there isn't any kind of constraint or requirement to create the LSP, even though RSVP is selected as signaling protocol. Only basic parameters related with source and destination of the tunnel or priorities shall be taken into account. (Note: This use case don't cover any type of protection for LSP, so only a primary path should be establish).

Depending on the entity which initiated the tunnel two scenarios are considered:

- **4.1a) PCC-initiated LSP**

As of those base requirements the head-end (ingress) router will try to establish the RSVP tunnel to a tail-end (egress) router according to the optimal shortest-path based on IGP (or TE) metric. After the LSP state information is synchronized, the NE acting as PCC delegates the LSP just created to one PCE, which is the main active stateful PCE. At that point, LSP is under external control of the PCE that could ask the PCC could to re-signals the LSP based on a best path it receives from a PCE.

On the other hand, the following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case. A deeper level detail of the operations (for further study) would explain how to execute each specific function (RPC calls and XML).

ID	Operation	Description
TE -2	Create RSVP or SR signalled LSP without constraint	Creation of a RSVP or SR LSP without any type of constrain calculation or protection, only basic parameters like source and destination
TE -8	Enable LSP PCE Delegation	Delegate the path computation responsibility for a LSP to the PCE
TE -10	Delete RSVP or SR signalled LSP	Deletion of a RSVP or SR LSP without any type of constrain calculation or protection
TE-13	Disable PCE Delegation for a LSP	Un-delegate the path computation responsibility for a LSP to the PCE
TE-14	Retrieve state of an existing LSP	Show information of a LSP to know if the tunnel was successfully created by checking the “operational-state”
TE-15	Modify basic parameters of an existing LSP	Change the description or administrative status of a LSP

Table 24 Atomic Operations for Traffic Engineering (1)

- **4.1.b) PCE-initiated LSP**

This scenario would be equivalent to PCC-initiated where only a Explicit Route Object (ERO) object has to be sent from the PCE to the PCC. The PCC creates the LSP using the parameters provided by the PCE, assigns the PCE-initiated LSP a unique LSP identifier, and automatically delegates the LSP to the PCE. A

PCE-initiated LSP is dynamically created by an external PCE, as a result there is no LSP configuration present on the PCC. For this reason, there is no need of OpenConfig parametrization in NE either.

A PCC cannot revoke the delegation for the PCE-initiated LSPs for an active PCEP session. When a PCEP session terminates due to a PCE failure, the PCC shouldn't immediately delete the PCE-initiated LSPs, but wait the applicable timers to expire (Redelegation Timeout Interval and State Timeout Interval, see RFC8281) in order to avoid services disruption. Thus, only when the second timer expires and no other PCE has acquired control over the LSPs from the failed PCE, the PCC deletes all the LSPs provisioned by the failed PCE. PCE is designed to work in high availability in order to avoid this scenario to ever happen.

9.4.2 Use case 4.2 - LSP create, modify and delete with constraints (delay, bandwidth and hop count) – SIGNALLING: SR

9.4.2.1 Use Case Definition

Number	**4.2**
Name	**LSP Creation, modify and delete with SR no contraints**
Brief description	In this use case the LSP is establish based on constraint-based routing performed by the PCC (router/NE) or a external PCE. The LSP will use SEGMENT ROUTING as signalling protocol and won't have any kind of protection.
Feature Bundles required	BASIC_TRAFFIC_ENGINEERING

Table 25 Definition of Use Case 4.2

9.4.2.2 Description

For this use case would apply all the considerations mentioned for previous use case, but using segment routing as signaling-protocol in MPLS yang model node as indicated in the below table (the rest of the parameters would be equivalent to the previous use case and they are not repeated here)

Likewise, the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case are equivalent to ones described for the previous use case.

9.4.3 Use case 4.3 - LSP create, modify and delete with constraints (delay, bandwidth and hop count)

9.4.3.1 Use Case Definition

Number	**4.3**
Name	**LSP create, modify and delete with constraints (delay, bandwidth and hop count)**
Brief description	In this use case the LSP is establish based on constraint-based routing performed by the PCC (router/NE) or a external PCE. The result of Constrained Shortest Path First (CSPF) algorithm will be the path that meets the constraints or the set (combination) of constraints considered, in this case delay, BW and hop limit. The LSP will use RSVP or SR as signalling protocol and won't have any kind of protection.
Feature Bundles required	BASIC_TRAFFIC_ENGINEERING

Table 26 Definition of Use Case 4.3

9.4.3.2 Description

The path computation takes into account information provided by the topology information from link-state routing protocol, the current network resource utilization determined by RSVP, and the resource requirements and constraints of the LSP. The output of the CSPF (Constrained Shortest Path First) calculation is an explicit route consisting of a sequence of router addresses that provides the shortest path through the network that meets the constraints. For signaled constrained-path LSPs to work, either IS-IS or OSPF protocols and IS-IS or OSPF traffic engineering extensions must be enabled on all routers.

In this particular use case, the constraints to be taken into account for CSPF path computation process would be **delay, BW** and/or **hop limit**. (Note: This use case don't cover any type of protection for LSP, so only a primary path should be establish).

Depending on the entity which initiated the tunnel two scenarios are considered.

- **4.3a) PCC-initiated LSP**

As of defined constraints, the head-end (ingress) router will try to establish the RSVP-TE tunnel to a tail-end (egress) router according to the path derived from Constrained Shortest Path First (CSPF) computation. After the LSP state information is synchronized, the NE acting as PCC delegates the LSP just created to one PCE, which is the main active stateful PCE. At that point, LSP is under external control of the PCE that could ask the PCC to re-signal the LSP based on a best path it receives from a PCE.

The **xpath** of the parameters to be considered for a constrained LSP configuration in accordance to required values for BW, delay or hop-count taken as constraints, are additional to general for tunnel definition as described in previous section.

On the other hand, the following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case. A deeper level detail of the operations (for further study) would explain how to execute each specific function (RPC calls and XML).

ID	Operation	Description

TE -3	Create RSVP or SR signalled LSP with constraint (BW)	Creation of RSVP or SR LSP taken into account for CSPF path computation process a BW constraint
TE -4	Create RSVP or SR signalled LSP with constraint (Delay)	Creation of RSVP or SR LSP taken into account for CSPF path computation process a Delay constra ▯ NOT SUPPORTED BY OPENCONFIG MODELS YET.
TE -5	Create RSVP or SR signalled LSP with constraint (Hop-Limit)	Creation of RSVP or SR LSP taken into account for CSPF path computation process a Hop Limit constraint ▯ NOT SUPPORTED BY OPENCONFIG MODELS YET.
TE -8	Enable LSP PCE Delegation	Delegate the path computation responsibility for a LSP to the PCE
TE -10	Delete RSVP or SR signalled LSP	Deletion of a RSVP or SR LSP without any type of constrain calculation or protection
TE -13	Disable LSP PCE Delegation	Un-delegate the path computation responsibility for a LSP to the PCE
TE-14	Retrieve state of an existing LSP	Show information of a LSP to know if the tunnel was successfully created by checking the "operational-state"
TE-15	Modify basic parameters of an existing LSP	Change the description or administrative status of a LSP
TE-16	Modify constraints of an existing RSVP or SR signalled LSP	Change the value of a constraint or add a new one so that LSP re-signal with the new configuration (at the moment only apply to BW)

Table 27 Atomic Operations for Traffic Engineering (2)

- **4.3b) PCE-initiated LSP**

The same considerations as in use case 4.1 applies. As the path is PCE initiated, there is no need to send the constraints to the node, just the ERO. It is the PCE that makes all calculations using constraints received via any app (northbound or operator demand),. The path must be continuously updated, as in active stateful mode.

9.4.4 Use case 4.4 - LSP create, modify and delete with constraints (delay, bandwidth and hop count) and explicit Path (strict and loose)

9.4.4.1 Use Case Definition

Number	**4.4**
Name	**LSP create, modify and delete with constraints (delay, bandwidth and hop count) and explicit Path (strict and loose)**
Short description	An explicit-path (either SR or RVSP-TE) is triggered from an operator action. Once the operator triggers the creation of the path, it is initiated using the explicitly specified path which comes from a planning process. If the path is topologically not feasible, either because the network is partitioned or insufficient resources are available along some parts of the path, the LSP creation will fail. No alternative paths can be used. If the setup succeeds, the LSP stays on the defined path indefinitely. The initiated path in this use case does not consider protection. The explicit path can contain strict or loose hops.
Feature Bundles required	BASIC_TRAFFIC_ENGINEERING

Table 28 Definition of Use Case 4.4

9.4.4.2 Description

In the case of explicit-path LSPs, all intermediate hops of the tunnel are pre-computed (operator, planning tool…) without neither the PCE nor the PCC performs a constrained path calculation. The intermediate hops can be strict, loose, or any combination of the two. When a strict hop is configured, it identifies an exact path through which the LSP must be routed. (Note: This use case do not cover any type of protection for LSP, so only a primary path should be established).

Depending on the entity which iniciated the tunnel two scenarios are considered.

- **4.4a) PCC-initiated LSP**

An explicit-path LSP is initiated by NE only along the explicitly specified path. After the LSP state information is synchronized, the NE acting as PCC delegates the LSP just created to the main active stateful PCE.

In the table below are detailed **an example** of how it would be a LSP configuration using **Explicit Route** with 3 strict hops (Note: Using OpenConfig MPLS module "openconfig-mpls" as a container within module "network-instance" for type "DEFAULT_INSTANCE, as stated in previous section).

The **xpath** and description of the parameters to be considered for a configuration using Explicit Route with 3 strict hops are additional to general for tunnel definition as described in previous section.

On the other hand, the following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case.

ID	Operation	Description
TE -1	Create Explicit-Path LSP	Creation of a LSP where all intermediate hops are manually configured
TE -9	Delete Explicit-Path LSP	Deletion of a LSP where all intermediate hops are manually configured
TE -8	Enable LSP PCE Delegation	Delegate the path computation responsibility for a LSP to the PCE
TE-13	Disable PCE Delegation for a LSP	Un-delegate the path computation responsibility for a LSP to the PCE
TE-14	Retrieve state of an existing LSP	Show information of a LSP to know if the tunnel was successfully created by checking the "operational-state"
TE-15	Modify basic parameters of an existing LSP	Change the description or administrative status of a LSP
TE-17	Modify an existing Explicit-Path LSP hops	Change, add or erase intermediate nodes to a Explicit-Path LSP
TE-19	Retrieve hops of an existing Explicit-Path LSP	Show information of intermediate nodes used by a Explicit-Path LSP

Table 29 Atomic Operations for Traffic Engineering (3)

- **4.4b) PCE-initiated LSP**

The same considerations as in use case 4.1 applies. As the path is PCE initiated, the explicit path is sent via PCEP protocol.

9.4.5 Use case 4.5 - LSP create, modify and delete with constraints (delay, bandwidth and hop count) – PROTECTION: Redundancy 1+1

9.4.5.1 Use Case Definition

Number	**4.5**
Name	**LSP create, modify and delete with constraints (delay, bandwidth and hop count) – PROTECTION: Redundancy 1+1**
Short description	This use case refers to 1+1 protection where there is one working LSP and one protection LSP
Feature Bundles required	BASIC_TRAFFIC_ENGINEERING

Table 30 Definition of Use Case 4.5

9.4.5.2 Description

This use case refers to 1+1 protection where there is one working LSP and one protection LSP. Based on this, a pre-signaled protecting LSP over dedicated resources is deployed at the very same time with the working LSP. At the ingress node, in normal situation, the traffic is sent through working LSP and in case of a failure the two nodes R5 and R1 (egress/ingress) should be coordinated to switch the traffic from the primary to the protection LSP.

Translated to the SBI, this scenario would match with a MPLS Path Protection scheme as depicted in the **Figure 29.** LSP network diagram with 1+1 redundancy:

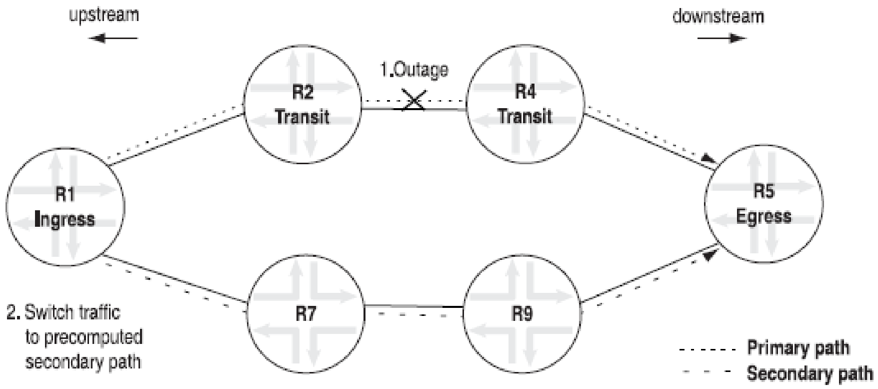


Figure 29. LSP network diagram with 1+1 redundancy

Depending on the entity which initiated the tunnel two scenarios are considered.

- **4.5a) PCC-initiated LSP**

Based on the protection required for the use case, the table below describes **an example** of what parameters or attributes should be considered for a constrained **LSP with Path Protection** with two LSP, Primary and Secondary working together in a hot standby behavior, where the secondary path remains up indefinitely to provide instant switchover if connectivity problems in Primary path occur. The protection path can be disjoint and use Node, Link or SRLG restrictions to deploy the working and protecting LSPs. (Note: Using OpenConfig mpls module “openconfig-mpls” as a container within module “network-instance” for type “DEFAULT_INSTANCE, as stated in previous section 9.3).

As can be observed, within OpenConfig MPLS model the tunnel configuration container includes a node called “protection-style-requested” that support 3 options:

- UNPROTECTED

- LINK_PROTECTION_REQUIRED
- LINK_NODE_PROTECTION_REQUESTED

where the two last ones are used for FRR (Fast Reroute) protection model. For this particular use case we can select either of them so that we could complement Path Protection end-to-end (Primary and Secondary LSPs) with FRR enabled (LINK_PROTECTION_REQUIRED or LINK_NODE_PROTECTION_REQUESTED) or not enabled (UNPROTECTED).

The **xpath** and description of the parameters to be considered for a configuration with 1+1 LSP redundancy includes specific additional p2p secondary path configuration, additional to general for tunnel and primary LSP definition as described in previous section.

On the other hand, the following table represents the set of NETCONF operations that must be supported by the SBI of the SDN controller for implementing this use case. A deeper level detail of the operations (to be included in the MUST Test plan) would explain how to execute each specific function (RPC calls and XML).

ID	Operation	Description
TE -6	Create LSP with Path Protection (Primary Path + Secondary Path)	Creation of path protection where there is one working LSP and one protection LSP.
TE -7	Create LSP with Node or Link Protection (FRR)	Creation of FRR node or link protection for a LSP
TE -8	Enable LSP PCE Delegation	Delegate the path computation responsibility for a LSP to the PCE
TE -11	Delete LSP with Path Protection	Deletion of path protection where there is one working LSP and one protection LSP
TE -12	Disable Node or Link Protection (FRR)	Deletion of FRR node or link protection for a LSP
TE -13	Disable LSP PCE Delegation	Un-delegate the path computation responsibility for a LSP to the PCE
TE-14	Retrieve state of an existing LSP	Show information of a LSP to know if the tunnel was successfully created by checking the “operational-state”
TE-15	Modify basic parameters of an existing LSP	Change the description or administrative status of a LSP

TE-18	Retrieve Secondary Path of a Primary Path for an existing LSP	Show the Secondary Path (protection) associated to a Primary Path (working) LSP

Table 31 Atomic Operations for Traffic Engineering (4)

- ### 4.5b) PCE-initiated LSP

In this case, the protection LSP can be also initiated by a stateful PCE, which retains the control of the LSP. The PCE is responsible for computing the path of the LSP and updating to the PCC with the information about the path.

10

Network Creation (Category 6)

Network creation is an essential process on the evolution of Transport's network, since it allows to provide coverage to new geographical areas, improving connectivity for millions of residential and enterprise customers. That is why the activity of expanding the network is quite critical and involves a previous careful and detailed planification to cover the customer's

10 Category 6: Network Creation

Network creation is an essential process on the evolution of Transport's network, since it allows to provide coverage to new geographical areas, improving connectivity for millions of residential and enterprise customers. That is why the activity of expanding the network is quite critical and involves a previous careful and detailed planification to cover the customer's demands.

Although network creation exists before SDN paradigm came up, this service-oriented networking concept together with cloud computing and virtualization, have the objective of reaching a fully automated network environment by leveraging Zero-touch provisioning (ZTP from now on), which promises day-zero configuration and provisioning of new infrastructure equipment without manual intervention.

However, current network devices hardware and software, and upper layer operation systems maturity is not completely ready for this ZTP way of network creation. Because of that, TIP project is proposing an approach in which new network devices have some precondition baselines (pre-commissioning) that should be enough to guarantee that the configuration left can be provided automatically by using the NBI/SBI Yang models.

10.1 Structure and Classification

Transport's network is organized into different hierarchical layers, from core/backbone to lower aggregation layers: HL1, HL2, H3, HL4, HL5. Because of that, Network Creation use cases-related have been divided to fit with configuration scope of each layered network element type.

The lists of use cases identified for network creation are:

ID	Use case Title	Section
5.1	Use Case 5.1 - Device commissioning	10.2.1

5.2	Use Case 5.2 - Network infrastructure configuration templates	10.2.2

10.2 Uses Cases

10.2.1 Use Case 5.1 - Device commissioning

Commissioning is the process of configuring the nodes in the network so they can communicate to each other for different protocols and be reachable by management servers and network functions support systems such as aaa, logging or NTP..

To make sure that the device is prepared for the commissioning, following aspects must be considered:

- Hardware specific configuration is out the scope.
- QoS configuration is out of the scope.
- Device must be powered on
- The following attributes should be preconfigured on the device prior to its commissioning:
  - Hostname
  - Management IP address
  - Default route to reach SDN controller
  - SSH and NETCONF server
  - Local user to log in
  - Hardware-related license activation

10.2.1.1 Definition

As the commissioning process is equivalent for the three types of NEs, this use case is going to be considered as a whole, applicable to HL1, HL2 and/or HL3 interchangeably. Further, HL4 and HL5 should be also considered within the scope of this use case. Nevertheless, the set of OC parameters described in the next section which apply to each NEs equipment shall be indicated.

Number	**5.1**
Name	**Use Case 5.1 - Device commissioning**

Brief description	Configuration of basic parameters related to protocols, interfaces, system attributes and other logical resources such as communities or prefix-lists on HLx devices
Feature Bundles required	CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT L3VPN_CE_PE_ROUTING BASIC_L3VPN_MANAGEMENT INTERFACE_MANAGEMENT BFD_MANAGEMENT

Table 32 Definition of Use Case 5.1

10.2.1.2 Description

The commissioning of a NE involves the configuration of a variety of protocols and functionalities included in a number of OpenConfig modules, specifically:

- ✓ **“openconfig-network-instance”** MPLS and routing protocols (RSVP, LDP, ISIS, BGP, PIM) Note: The aim of the commissioning is to load in the NE a base configuration that allows a further definition and configuration of the services. Hence, that configuration should be created in global table (network-instance type = DEFAULT) and would be seen as applying to the NE as a whole not to a specific service. Other modules also used to complete day-one commissioning configuration are the following:
- ✓ **"openconfig-bfd"** BFD
- ✓ **"openconfig-interfaces"** L2/L3 interface parametrization (importing “openconfig-if-ethernet”, “openconfig-if-ip)
- ✓ **“openconfig-acl” -** ACLs
- ✓ **“openconfig-bgp-policy”, “openconfig-routing-policy”** □ BGP Policies
- ✓ **“openconfig-bgp”** BGP neighbors (imported by “openconfig-network-instance” module)
- ✓ **“openconfig-segment-routing.yang”** Configuration and operational state parameters relating to the segment routing (imported by “openconfig-network-instance” module)

10.2.1.3 Out of the scope

The following configurations are out of the scope of the current version of this TRD.

- PIM – For multicast scenarios

- System Management - "Configuration of system management functions on the device: AAA, NTP, Logging

10.2.2 Use Case 5.2 - Network infrastructure configuration templates

10.2.2.1 Definition

Number	**5.2**
Name	**Network Infrastructure Configuration (interfaces, ports, ACLS, profiles, rebooting..)**
Brief description	The main purpose of this use case is to automate the workday routine in the network infrastructure configuration that is repeated or standardized in all the network equipment or an area.
Feature Bundles required	CORE_ROUTING_SIGNALLING_PROTOCOLS_MANAGEMENT PE_PE_ROUTING_MPBGP_MANAGEMENT L3VPN_CE_PE_ROUTING BASIC_L3VPN_MANAGEMENT INTERFACE_MANAGEMENT ACL_MANAGEMENT BFD_MANAGEMENT

Table 33 Definition of Use Case 5.2

10.2.2.2 Description

Aside from commissioning use case, covered in use case Use Case 5.1: “Device commissioning” in Section 12.2.1, which was focused on the first configuration of a device at the time of its integration in the network along with other devices, there will be needed a further configuration to be performed on a daily basis. Thus, a great part of this day-to-day configuration of a device resides in the configuration of interfaces, ports, ACLs, profiles, and other type of basic information.

Hence, the aim of this use case will be to identify those network infrastructure related configuration tasks and associated parameters so that could be used in templates to be executed on demand by the SDN controller in a regular O&M task. Please note that this is only a first collection of basic configuration items, so certainly it will be extended in the future as O&M groups identify new needs).

Specifically, two main set of parameters are considered depending of the type of task to be performed (they could be further grouped in different templates):

- **Adding/modifying/deleting configuration in an interface/port, including**
  - ✓ L2: Ethernet Framing
  - ✓ L3: IP address
  - ✓ Routing Protocol: ISIS
  - ✓ MPLS: LDP, RSVP
  - ✓ Segment Routing
  - ✓ Link Failure Detection: BFD
  - ✓ Access Control: ACL
- **Adding/modifying/deleting BGP routing policies regarding items such as:**
  - ✓ IP-Prefix sets
  - ✓ BGP Community-sets
  - ✓ As-path-sets
  - ✓ BGP Policies Definition
  - ✓ BGP neighbors

As stated previously, Transport's network is organized into different hierarchical levels, from core/backbone to lower aggregation layers: HL1, HL2, H3, HL4, HL5. As for this use case, parametrization included are generic, so it affects whatever type of HLx network element are.

The set of parameters to be configured are taken from the following OpenConfig modules.

- ✓ "**network-instance.yang**" □ Configuration of MPLS and routing protocols (RSVP, LDP, ISIS, BGP, PIM). The configuration should be created in global table (network-instance type = DEFAULT). This module imports other modules for each category of protocol: "openconfig-mpls", "openconfig-bgp", "openconfig-isis", "openconfig-pim".
- ✓ **"openconfig-bfd"** BFD
- ✓ **"openconfig-system"** AAA (import "openconfig-aaa module"), NTP, DNS, SSH, …
- ✓ **"openconfig-interfaces"** L2/L3 interface parametrization (importing "openconfig-if-ethernet", "openconfig-if-ip)

- ✓ **“openconfig-acl”** ACLs
- ✓ **“openconfig-bgp-policy”, “openconfig-routing-policy”** BGP Policies
- ✓ **“openconfig-bgp”** BGP neighbors (imported by “openconfig-network-instance” module)
- ✓ **“openconfig-segment-routing.yang”** Configuration and operational state parameters relating to the segment routing (imported by “openconfig-network-instance” module)

11

Support to Performance Monitoring (Category 7)

Traditionally, the Simple Network Management Protocol SNMP has been extensively used for the Performance Management functions. Within Service Providers Business Units, SNMP has been activated in the NEs as an SNMP Agent, and some EMS/NMS subsystems as SNMP Managers with Proxy

11 Category 7: Support to Performance Monitoring

11.1 Performance Monitoring Evolution

Traditionally, the Simple Network Management Protocol SNMP has been extensively used for the Performance Management functions. Within Service Providers Business Units, SNMP has been activated in the NEs as an SNMP Agent, and some EMS/NMS subsystems as SNMP Managers with Proxy Forwarder capabilities for feeding OSS-PM systems. Thus, two traditional extraction models for collect performance information have been predominant so far:

- Directly extraction from OSS layer to the NE (no longer recommended).
- Proxy-base extraction using as a collection central point the NMS.

On the other hand, performance and event monitoring data can be classified as: element-centric, network-centric and service-centric. For element centric, SDN Controller must extract from NE, and expose in the NBI data referring to the behavior of any particular component of the equipment, either physical such as CPU consumption, Memory, Temperature, interface packets counters and errors, links occupation, etc.; or logical, such as routing function or control plane related measurements. As to network-centric information, SDN Controller, or any other system above, should orchestrate end-to-end network performance information regarding latency, jitter, and error rate, for test and diagnosis uses cases based on TWAMP or alternative standard methods. Finally, for the service-centric approach, SDN Controller should require performing Y.1564 and RFC 2544 like testing for the the validation of Ethernet service-level agreements (SLAs).

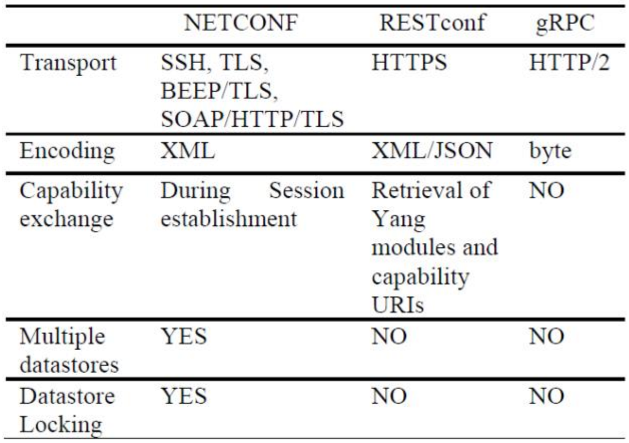
However, two new ways to obtain the performance and events information have arisen for SDN-ready network elements:

- SDN-based extraction using the SDN Controller to collect performance data encoded in XML, directly from the Network Elements, by reading (polling) the intended counter on state branch within the corresponding OpenConfig Yang data model. NETCONF protocol is used in the same way as to access device datastores for configuration.
- SDN-based extraction, using model-driven streaming telemetry engines for publishing/subscribing real time performance information, by defining sensor data delivery based on sample period or only if changes. In this case, gRPC protocol and protocol buffers are best suited for telemetry streaming

collection. gRPC Remote Procedure Calls (RPC) is a cloud native high-performance RPC framework open sourced by Google. It uses HTTP/2 as a transport protocol and uses protocol buffers encodings for transported messages.

In both cases the SDN Controller would expose collected information and data on NBI towards OSS or external 3rd Party applications through a standard messaging bus (Kafka, RabbitMQ or other).

This table provide a summary of the presented properties of the different SDN protocols used for performance and telemetry collection. As stated, gRPC offers better features and efficiency for telemetry recovery. Thus, NETCONF, which is using XML, requires more payload as gRPC due to its optimized byte encoding. Further, gRPC latency through notification stream is less than NETCONF with SSH session establishment and capability exchange (gRPC has no exchange mechanism and this implies that data models need to be known at both sides).



	NETCONF	RESTconf	gRPC
Transport	SSH, TLS, BEEP/TLS, SOAP/HTTP/TLS	HTTPS	HTTP/2
Encoding	XML	XML/JSON	byte
Capability exchange	During Session establishment	Retrieval of Yang modules and capability URIs	NO
Multiple datastores	YES	NO	NO
Datastore Locking	YES	NO	NO

In the following sections, use cases representing these two models of gathering and notifying performance information from NE in SBI, polling and telemetry, are described as well as OpenConfig parametrization associated.

11.2 Structure and Classification

Based on previously described, the lists of use cases identified for supporting performance and fault management within the TIP Project are listed in the following table.

ID	Use case Title	Section

6.1	Reporting performance information [Node: CPU, Temp, Memory]	11.3.1
6.2	Reporting performance information [Interface: Counters, Packets, Errors]	11.3.2

11.3 Use Cases

11.3.1 Use Case 6.1 Reporting performance information [Node: CPU, Temp, Memory]

11.3.1.1 Use Case Definition

Number	**Use Case 6.1**
Name	Reporting performance information [Node: CPU, Temp, Memory]
Brief description	In this use case, the main objective is to obtain the performance information state is about CPU, Temperature and Memory of Routers and Switches equipment. This information should be exposed throughout Platform OpenConfig models that will be described in this document.
Feature Bundles required	DEVICE_INTERNAL_COMPONENTS_MANAGEMENT HARDWARE_PERFORMANCE_STATISTICS

Table 34 Definition of Use Case 6.1

11.3.1.2 Description

This use case describes the scenario of a SDN Controller collecting performance information about CPU, Temperature and Memory, directly through SBI interface. SDN Controller will use NETCONF and OpenConfig Yang data models to poll all network devices for related counters and statistics parameters, which will be exposed on NBI Interface. (Note: SNMP might keep using during a transition period until SDN Controller and network elements with Netconf/OpenConfig support were widely deployed in the Service Providers networks).

The list of parameters is included in the feature bundles DEVICE_INTERNAL_COMPONENTS_MANAGEMENT and HARDWARE_PERFORMANCE_STATISTICS. These two bundles are used to get information status about the CPU, Temperature and Memory of Routers and Switches equipment's, are mandatory to implement according with the Openconfig-Platform model **"openconfig-platform"** and **"openconfig-platform-types".**

11.3.2 Use Case 6.2 Reporting performance information [Interface: Counters, Packets, Errors]

11.3.2.1 Use Case Definition

Number	**Use Case 6.2**
Name	**Reporting performance information [Interface: Counters, Packets, Errors]**
Brief description	With this case it should be possible to get all the performance information of a system including, delay, packet error rate, dropped packets, throughput, and any other counter. It should use Yang based collectors instead of data gathering SNMP techniques.
Feature Bundles required	INTERFACE_PERFORMANCE_STATISTICS

Table 35 Definition of Use Case 6.2

11.3.2.2 Description

As in the use case 6.1 the present use case is aimed at reporting element-centric information, namely, interface performance data for L2 ethernet KPIs, including counters for packets processed or errors found and throughput. Again, SDN Controller will use NETCONF and OC Yang data models parameters to read counters (polling from SDN Controller) and expose on NBI (Note: SNMP might keep using during a transition period until SDN Controller and network elements with Netconf/OpenConfig support were widely deployed in the Service Providers networks).

The **xpath** of the parameters used belongs to these Yang modules: **“openconfig-interfaces”, “openconfig-if-ethernet”, “openconfig-if-ip”, “openconfig-platform”, “openconfig-platform-transceiver”**.

Note that these are KPI performance counters already supported by referred OC models, but there is some others which TIP has missed for being used now.

12

Category 8: Telemetry

While the previous uses cases 6.1 and 6.2 focuses on how to get performance information through a classic polling method, recovering it from the NE, but using NETCONF and Yang data model schema, this use case 7.1 is intended to show a model-driven Telemetry scenario based on a subscription approach to get real time performance information..

12 Category 8: Telemetry

12.1.1 Use Case 7.1 Reporting Telemetry information [Links: Occupation]

12.1.2 Definition

Number	**7.1**
Name	**Reporting telemetry information [Links: Occupation]**
Brief description	With this case it should be possible to get all the performance information of a link including, delay, packet error rate, dropped packets, throughput, and any other counter by means of Telemetry. Telemetry should use Yang based collectors instead of data gathering SNMP techniques.
Feature Bundles required	TELEMETRY_MANAGEMENT

Table 36 Definition of Use Case 7.1

12.1.3 Description

While the previous uses cases 6.1 and 6.2 focuses on how to get performance information through a classic polling method, recovering it from the NE, but using NETCONF and Yang data model schema, this use case 7.1 is intended to show a model-driven Telemetry scenario based on a subscription approach to get real time performance information. Specifically, the information to be reported by telemetry refers to the of occupation (%) reached in an interface over time.

According with the protocols described at the beginning of this document, telemetry collection can be accomplished throughout gRPC/gNMI and Netconf, so they must be mandatory supported in Router or Switches equipment's (Note: SNMP might keep using during a transition period until SDN Controller and network elements with Netconf/OpenConfig support were widely deployed in the Service Providers networks).

As already mentioned, streaming telemetry is a new paradigm in monitoring the health of a network. It provides a mechanism to efficiently stream configuration and operational data of interest from NE. This streamed data is transmitted in a structured format to remote management systems for monitoring and troubleshooting purposes. It works on a subscription model where you subscribe to the data of interest in the form of sensor paths. The sensor paths describe OpenConfig data models or vendor native data models. You choose who initiates the subscription by establishing a telemetry session between the router and the receiver. The session is established using either a dial-out mode or a dial-in mode.

Although the modes to establish a telemetry session are different, both modes use the same data model and stream the same data.

- In a **dial-out** mode, the router dials out to the receiver to establish a subscription-based telemetry session. Because the router initiates the connection, there is no need to manage the ports for inbound traffic. If the connection between the router and the destination is lost, the router re-establishes the connection with the destination and continues to push data again. However, data transmitted during the time of reconnection is lost. The protocols used to establish a session gives you the flexibility to choose between simplicity of TCP or security capabilities to authenticate and encrypt the session using gRPC.

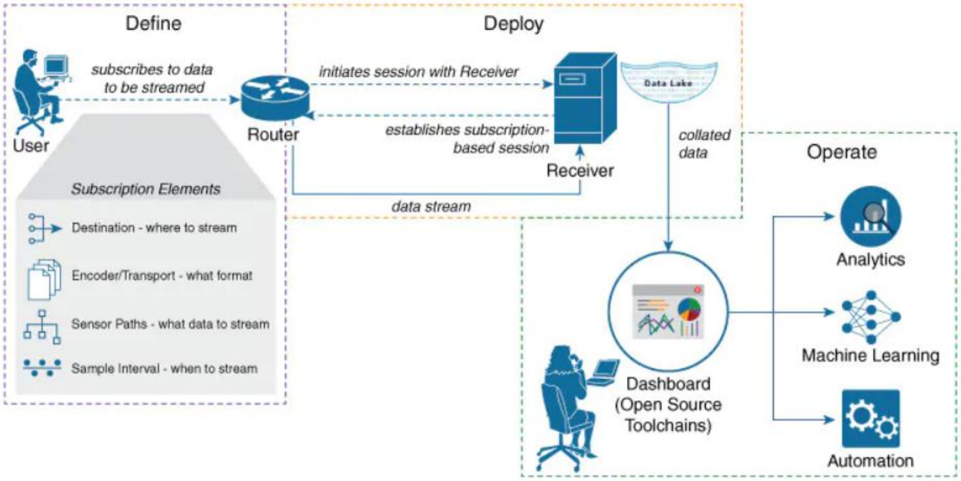

Define
Deploy
subscribes to data to be streamed
User
Router
initiates session with Receiver
establishes subscription-based session
Receiver
Data Lake
collated data
data stream
Subscription Elements
Destination - where to stream
Encoder/Transport - what format
Sensor Paths - what data to stream
Sample Interval - when to stream
Operate
Analytics
Machine Learning
Automation
Dashboard (Open Source Toolchains)

Figure 30. Telemetry session dial-out mode scenario

- In a **dial-in** mode, a collector dials in to the router to establish a telemetry session and subscribes dynamically to one or more sensor paths specified in a subscription. The router streams telemetry data through the same session that is established by the receiver. Because the collector establishes the session, there is no need to create destinations in the configuration. If the connection between the router and the collector is lost, the session is cancelled. The collector must reconnect to the router to restart streaming data. This dynamic subscription terminates when the receiver cancels the subscription or when the session terminates. Only gRPC supports dial-in session, thus the router must have enabled gRPC for connections from the collector.

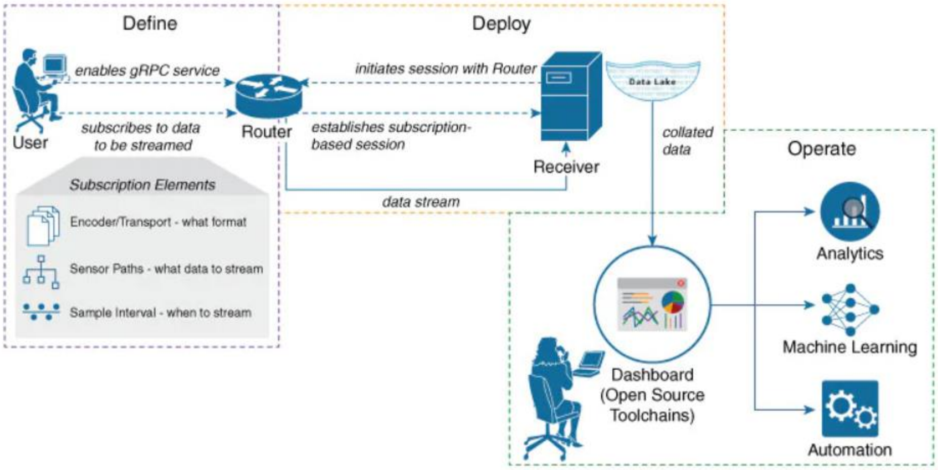

Define
Deploy
enables gRPC service
User
subscribes to data to be streamed
Router
initiates session with Router
establishes subscription-based session
Receiver
Data Lake
collated data
data stream
Subscription Elements
Encoder/Transport - what format
Sensor Paths - what data to stream
Sample Interval - when to stream
Operate
Analytics
Machine Learning
Automation
Dashboard (Open Source Toolchains)

Figure 31. Telemetry session dial-in mode scenario

In both scenarios, when creating a subscription to define the data of interest to be streamed from the router to the destination, a set of elements or parameters have to be provided:

- ✓ **Destination-groups**: Contain the details about the destinations. Include the destination address (ipv4 or ipv6), port, transport, and encoding format (only apply to DIAL-OUT)
- ✓ **Sensor-groups**: Contain the sensor paths. Sensor path represents the path in the hierarchy of a telemetry YANG data model, specifying the subset of the data that you want to stream from the router
- ✓ **Subscriptions:** Subscription binds the destination-group (DIAL-OUT) with the sensor-group and sets the streaming method. Separating the sensor-paths into different subscriptions enhances the efficiency of the router to retrieve operational data at scale. The streaming method can be cadence-driven or event-driven telemetry.
  - o Cadence-driven telemetry continually streams data (operational statistics and state transitions) at a configured cadence. The higher frequency of the data that is continuously streamed helps you closely identify emerging patterns in the network
  - o Event-driven telemetry optimizes data that is collected at the receiver and streams data only when a state transition occurs and

thus optimizes data that is collected at the receiver. For example, EDT streams data about interface state transitions, IP route updates, and so on.

The configuration for event-driven telemetry is similar to cadence-driven telemetry, with only the sample interval as the differentiator. Configuring the sample interval value to 0, zero, sets the subscription for event-driven telemetry, while configuring the interval to any non-zero value sets the subscription for cadence-driven telemetry.

The OC model to configure model-driven telemetry parametrization as defined above is "**openconfig-telemetry.yang**". In case of using DIAL-IN mode, previously is going to be necessary to enable gRPC server on the router to accept incoming connections from the receiver. That configuration is made by means of **"openconfig-system.yang"** module which has configuration data for the gRPC server.

- **Subscriptions**

A telemetry subscription consists of a set of collection destinations, stream attributes, and associated paths to state information in the model (sensor data). There are two types, persistent and dynamic mapping to "dial-out" and "dial-in" modes, respectively. Although the modes to establish a telemetry session are different, both modes use the same data model and stream the same data:

- ✓ A persistent telemetry subscription is configured locally on the device through configuration, and is persistent across device restarts or other redundancy changes.
- ✓ A dynamic subscription is typically configured through an RPC channel, and does not persist across device restarts, or if the RPC channel is reset or otherwise torn down.

Accordingly, OC split subscription in two containers, one including "read-write access" parameters to configure permanent subscription and a second for dynamic subscription, where parameters are "read-only" in that case (a system generated identifier of the telemetry subscription is used to refer to each on-demand dynamic telemetry session created).

PROTOCOL	DESCRIPTION

STREAM_SSH	Telemetry stream is carried over a SSH connection
STREAM_GRPC	Telemetry stream is carried over via the gRPC framework
STREAM_JSON_RPC	Telemetry stream is carried via the JSON-RPC framework
STREAM_THRIFT_RPC	Telemetry stream is carried via the Apache Thrift framework
STREAM_WEBSOCKET_RPC	Telemetry stream is carried by the WebSocket framework

*/telemetry-system/subscriptions/persistent-subscriptions/persistent-subscription/config/**encoding*** ▯ Selection of the specific encoding or RPC framework for telemetry messages to and from the network element for configuration and operational state data. The following options are admitted, where at least PROTO BUFFERS should be supported.

ENCONDING	DESCRIPTION
ENC_XML	XML encoding
ENC_JSON_IETF	JSON encoded based on IETF draft standard
ENC_PROTO3	Protocol buffers v3 (Compact and Self-Describing GPB)

12

Category 9: Support to Fault Management

Nowadays, network management means understanding the details of a network or service failure. It means also resolve several post-mortem questions like What it comprises? Why does it fail? Who is impacted? Can it happen again? For that reason, the more information the devices can provide to the systems...

13 Category 9: Support to Fault Management

Nowadays, network management means understanding the details of a network or service failure. It means also resolve several postmortem questions like What it comprises? Why does it fail? Who is impacted? Can it happen again? For that reason, the more information the devices can provide to the systems the better the understanding of the network, and probably more precautionary actions can be taken.

13.1 Use Cases

13.1.1 Use Case 8.1 Sending Network events/alarms to Fault Management OSS [Node Alarms.]

13.1.1.1 Definition

Number	**8.1**
Name	**Sending network events/alarms to Fault Management OSS [Node Alarms]**
Brief description	In this use case, the main objective is to identify what state parameters are necessary to Fault Management and activate alarms information about state changes of principal components in an equipment. Node components like a (CPU, Memory, PSU, FAN, Transceiver and Temperature). This information should be exposed throughout Platform OpenConfig models that will be described in this document.
Feature Bundles	ALARM_MANAGEMENT HARDWARE_PERFORMANCE_STATISTICS DEVICE_INTERNAL_COMPONENTS

Table 37 Definition of Use Case 8.1

13.1.1.2 Description

The scope of this use case from SBI Interface is to identify the read parameters that help us build network events or alarms about a network equipment, these changes of specific components like a (CPU, Temperature, Memory and Power Supply) are classify in events with severity according to initial configuration or hardware specifications.

The SDN Controller subscribe to specific network events in a Network Element and perform periodic/on-change queries to receive the state information about any alarm risen by a hardware component. If any failure occurs, the network equipment sends the information to SDN Controller subscribed to the alarm's events.

The figure below depicts how the network equipment replies with the state information whenever necessary according to establish policies from NBI Interface or if is necessary send a specific alarm occurred to subscriber.

The streaming of information state collection can be throughout gRPC or Netconf, these must be mandatory supported in the network equipment's. The notification alarms about any hardware component should be get in SBI interface according with the Platform OpenConfig Yang data models, this information is gathered by SDN Controller and correlate all network events then evaluate the failure and take decisions about this. (Note: SNMP might keep using during a transition period until SDN Controller and network elements with Netconf/OpenConfig support were widely deployed in the Service Providers networks).

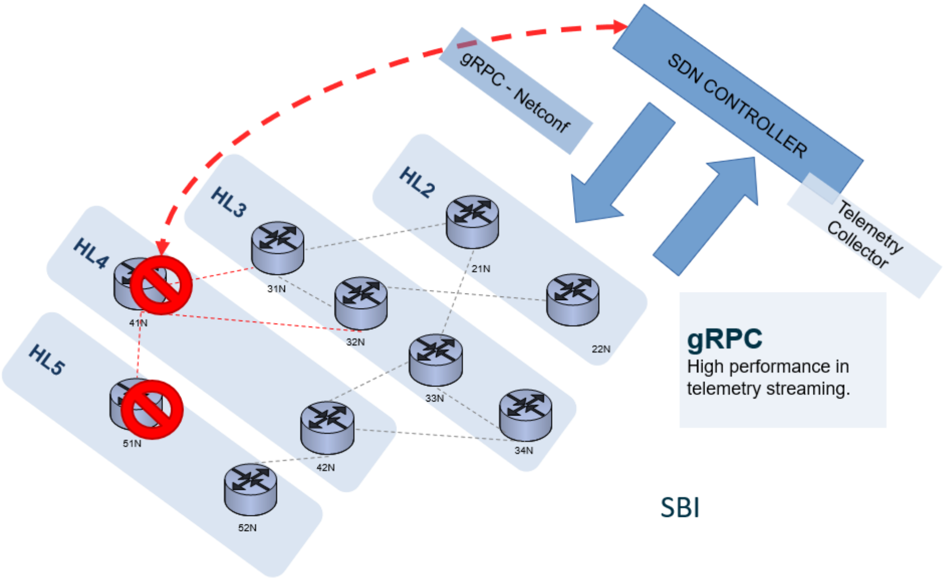


Figure 32. Scenario Fault Management streaming data.

The openconfig-alarm module has the alarm structure to notify the following information:

*/system/alarms/alarm/state/**id***	Unique ID for the alarm -- this will not be a configurable parameter on many implementations.
*/system/alarms/alarm/state/**resource***	The item that is under alarm within the device. The resource may be a reference to an item which is defined elsewhere in the model. For example, it may be a platform/component, interfaces/interface, terminal-device/logical-channels/channel, etc. In this case the system should match the name of the referenced item exactly. The referenced item could alternatively be the path of the item within the model.
*/system/alarms/alarm/state/**text***	The string used to inform operators about the alarm. This MUST contain enough information for an operator to be able to understand the problem. If this string contains structure, this format should be clearly documented for programs to be able to parse that information
*/system/alarms/alarm/state/time-**created***	The time at which the alarm was raised by the system. This value is expressed relative to the Unix Epoch.
*/system/alarms/alarm/state/**severity***	The severity level indicating the criticality and impact of the alarm
*/system/alarms/alarm/state/**type-id***	The abbreviated name of the alarm, for example LOS, EQPT, or OTS. Also referred to in different systems as condition type, alarm identifier, or alarm mnemonic. It is recommended to use the OPENCONFIG_ALARM_TYPE_ID identities where possible and only use the string type when the desired identityref is not yet defined

The lists of parameters (***resources***) below are used to report information of state failure/changes about the CPU, Temperature, Memory and Power Supply of network equipment's. Those attributes are mandatory to implement according with the openconfig-platform model with their platform sub-models.

Note that these are **PATHS** already supported by referred OC models, but the list should be extended as these Openconfig models or other included new ones.

13.1.2 Use Case 8.2 Sending Network events/alarms to Fault Management OSS [Interface Alarms.]

13.1.2.1 Definition

Number	**8.2**
Name	**Sending network events/alarms to Fault Management OSS [Interface Alarms]**

Brief description	In this use case, the main objective is identify what state parameters are necessary to Fault Management and activate alarms information about state changes of interface components in an equipment. This information should be expoused throughout Platform Openconfig models that will be described in this document.
Feature Bundles	INTERFACE_PERFORMANCE_STATISTICS

Table 38 Definition of Use Case 8.2

13.1.2.2 Description

The scope of this use case from SBI Interface is to identify the read parameters that help us build network events or alarms about an interface, these state changes like a (Counters, Packets (Inbound/Outbound, errors) are classify in events with severity according to initial configuration or hardware specifications.

The streaming of information state collection can be throughout gRPC and Netconf, these must be mandatory supported in the network equipment's. The notification alarms about any state parameters of interface attributes should be get in SBI interface according with the Platform OpenConfig Yang data models, this information is gathered by SDN Controller and correlate all network events then evaluate the failure and take decisions about this. (Note: SNMP might keep using during a transition period until SDN Controller and network elements with Netconf/OpenConfig support were widely deployed in the Service Providers networks).

The lists of parameters needed are used to get information of changes state about the interface attributes (Counters, Packets, errors or properties) are mandatory to implement according with the openconfig-platform model with their platform sub-models.

14 References

[1] Telecom Infra Project, “Open Transport SDN Architecture Whitepaper”, available online at https://cdn.brandfolder.io/D8DI15S7/at/jh6nnbb6bjvn7w7t5jbgm5n/OpenTransportArchitecture-Whitepaper_TIP_Final.pdf last seen on 01-February-2021.

[2] "Network Configuration Protocol (NETCONF)". RFC 6241

[3] "Path Computation Element (PCE) Communication Protocol (PCEP)". RFC 5440

[4] "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP". RFC 7752

[5] draft-openconfig-rtgwg-gnmi-spec-01 – “gRPC Network Management Interface (gNMI)”, https://datatracker.ietf.org/doc/html/draft-openconfig-rtgwg-gnmi-spec-01

[6] BGP - Link State (BGP-LS) Advertisement of IGP Traffic Engineering Performance Metric Extensions". RFC 8571

[7] OpenConfig, Vendor-neutral, model-driven network management designed by users, https://openconfig.net/

15Glossary

ACL	Access List
BGP	Border Gateway Protocol
BGP-LS	Border Gateway Protocol Link State
BSS	Business Support System
BW	Bandwidth
CE	Customer Edge
CIR	Committed Information Rate
CoS	Class of Service
CPE	Customer Premises Equipment
CSPF	Constrained Shortest Path First
DSCP	Differentiated Service CodePoint
ECN	Explicit Congestion Notification
ERO	Explicit Route Object
GRE	Generic Routing Encapsulation
gRPC	Remote Procedure Call developed by Google.
IANA	Internet Assigned Numbers Authority
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISIS	Intermediate System - Intermediate System
LACP	Link Aggregation Control Protocol
LAG	Link AGgregation
LLDP	Link Layer Discovery Protocol
LSP	Link State Path
MDI	Media Dependent Interface
MED	Multi Exit Discriminator
MIB	Management Information Base
MPLS	Multiprotocol Label Switching
MTU	Maximum Transmission Unit
MUST OOPT	Mandatory Use Cases for SDN for Transport / Open Optical & Packet Transport
NBI	Northbound Interface

NE	Network Element
NETCONF	Network Configuration
NLRI	Network Layer Reachability Information
OC	OpenConfig
OSPF	Open Shortest Path First
OSS	Operations Support System
PCC	Path Computational Client
PCE	Path Computational Element
PCEP	Path Computation Element Protocol
PDU	Protocol Data Unit
PE	Provider Edge
PIR	Peak Information Rate
QoS	Quality of Service
RD/RT	Route Distinguisher / Route Target
RED	Random Early Detection
RFC	Request For Comments
RPC	Remote Procedure Call
RRO	Record Route Object
RSVP	Resource Reservation Protocol
SBI	Southbound Interface
SDN	Software Defined Networks
SLA	Service Level Agreement
SR	Segment Routing
TE	Traffic Engineering
TED	Traffic Engineering Database
TLV	Type Length Value
TTL	Time To Live
UNI	User-to-Network Interface
VLL	Virtual Leased Line
VPN	Virtual Private Network
VSI	Virtual Switch Interface
VXLAN	Virtual Extensible Local Area Network
WFQ	Weighted Fair Queuing
WRED	Weighted Random Early Detection
XML	Extensible Markup Language
YANG	Yet Another Next Generation

TIP Document License

Disclaimers

THIS DOCUMENT IS PROVIDED "AS IS," AND TIP MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

TIP WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name or trademarks of TIP may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with TIP and its Contributors.
This TIP Document License is based, with permission from the W3C, on the W3C Document License which may be found
at https://www.w3.org/Consortium/Legal/2015/doc-license.html.