OPSAWG
Internet-Draft
Intended status: Informational
Expires: December 16, 2020

Q. Wu, Ed.
Huawei
M. Boucadair, Ed.
Orange
D. Lopez
Telefonica I+D
C. Xie
China Telecom
L. Geng
China Mobile
June 14, 2020

A Framework for Automating Service and Network Management with YANG
draft-ietf-opsawg-model-automation-framework-04

Abstract

Data models provide a programmatic approach to represent services and networks. Concretely, they can be used to derive configuration information for network and service components, and state information that will be monitored and tracked. Data models can be used during the service and network management life cycle, such as service instantiation, provisioning, optimization, monitoring, diagnostic, and assurance. Data models are also instrumental in the automation of network management, and they can provide closed-loop control for adaptive and deterministic service creation, delivery, and maintenance.

This document describes an architecture for service and network management automation that takes advantage of YANG modeling technologies. This architecture is drawn from a Network Operator perspective irrespective of the origin of a data module; it can thus accommodate modules that are developed outside the IETF.

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 16, 2020.

Table of Contents

1. Introduction

Service management systems usually comprise service activation/ provision and service operation. Current service delivery procedures, from the processing of customer's requirements and orders to service delivery and operation, typically assume the manipulation of data sequentially into multiple OSS/BSS applications that may be managed by different departments within the service provider's organization (e.g., billing factory, design factory, network operation center). In addition, many of these applications have been developed in-house over the years and operate in a silo mode:

- The lack of standard data input/output (i.e., data model) raises many challenges in system integration and often results in manual configuration tasks.

- Service fulfillment systems might have a limited visibility on the network state and therefore have slow response to network changes.

Software Defined Networking (SDN) becomes crucial to address these challenges. SDN techniques are meant to automate the overall service delivery procedures and typically rely upon standard data models. These models are used to not only reflect service providers' savoir-faire, but also to dynamically instantiate and enforce a set of service-inferred policies that best accommodate what has been defined and possibly negotiated with the customer. [RFC7149] provides a first tentative attempt to rationalize that service provider's view on the SDN space by identifying concrete technical domains that need to be considered and for which solutions can be provided:

- Techniques for the dynamic discovery of topology, devices, and capabilities, along with relevant information and data models that

are meant to precisely document such topology, devices, and their capabilities.

- Techniques for exposing network services [RFC8309] and their characteristics.

- Techniques used by service-derived dynamic resource allocation and policy enforcement schemes, so that networks can be programmed accordingly.

- Dynamic feedback mechanisms that are meant to assess how efficiently a given policy (or a set thereof) is enforced from a service fulfillment and assurance perspectives.

Models are key for each of the aforementioned four technical items. Service and network management automation is an important step to improve the agility of network operations. Models are also important to ease integrating multi-vendor solutions.

YANG [RFC7950] module developers have taken both top-down and bottom-up approaches to develop modules [RFC8199] and to establish a mapping between a network technology and customer requirements at the top or abstracting common constructs from various network technologies at the bottom. At the time of writing this document (2020), there are many data models including configuration and service models that have been specified or are being specified by the IETF. They cover many of the networking protocols and techniques. However, how these models work together to configure a device, manage a set of devices involved in a service, or provide a service is something that is not currently documented either within the IETF or other Standards Development Organizations (SDOs).

This document describes an architectural framework for service and network management automation (Section 3) that takes advantage of YANG modeling technologies and investigates how different layer YANG data models interact with each other (e.g., service mapping, model composing) in the context of service delivery and fulfillment (Section 4).

This framework is drawn from a Network Operator perspective irrespective of the origin of a data module; it can accommodate modules that are developed outside the IETF.

The document identifies a list of use cases to exemplify the proposed approach (Section 5), but it does not claim nor aim to be exhaustive.

2. Terminology and Acronyms

2.1. Terminology

The following terms are defined in [RFC8309][RFC8199] and are not redefined here:

- Network Operator
- Customer
- Service
- Data Model
- Service Model
- Network Element Module

In addition, the document makes use of the following terms:

Network Model: Describes a network level abstraction (or a subset of aspects of a network infrastructure), including devices and their subsystems, and relevant protocols operating at the link and network layers across multiple devices. This model corresponds to the Network Configuration Model discussed in [RFC8309].

It can be used by a Network Operator to allocate resources (e.g., tunnel resource, topology resource) for the service or schedule resources to meet the service requirements defined in a Service Model.

Device Model: Refers to the Network Element YANG data model described in [RFC8199] or the Device Configuration Model discussed in [RFC8309].

Device Models are also used to refer to model a function embedded in a device (e.g., Network Address Translation (NAT) [RFC8512], Access Control Lists (ACLs) [RFC8519]).

2.2. Acronyms

The following acronyms are used in the document:

ACL	Access Control List
CE	Customer Edge
ECA	Event Condition Action
L2VPN	Layer 2 Virtual Private Network

L3VPN	Layer 3 Virtual Private Network
NAT	Network Address Translation
OAM	Operations, Administration, and Maintenance
OWD	One-Way Delay
PE	Provider Edge
QoS	Quality of Service
RD	Route Distinguisher
RT	Route Target
SDN	Software Defined Networking
TE	Traffic Engineering
VN	Virtual Network
VPN	Virtual Private Network
VRF	Virtual Routing and Forwarding

3. Architectural Concepts and Goals

3.1. Data Models: Layering and Representation

As described in Section 2 of [RFC8199], layering of modules allows for better reusability of lower-layer modules by higher-level modules while limiting duplication of features across layers.

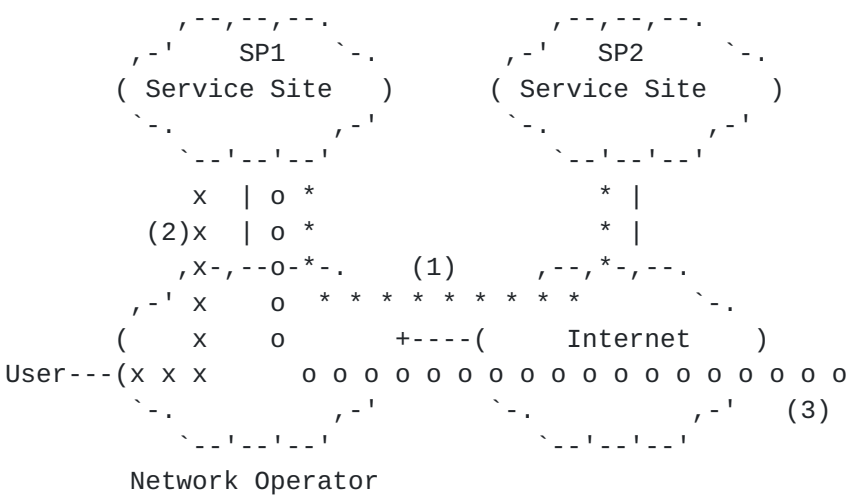
Data models can be classified into Service, Network, and Device Models. Different Service Models may rely on the same set of Network and/or Device Models.

Service Models traditionally follow top-down approach and are mostly customer-facing YANG modules providing a common model construct for higher level network services (e.g., Layer 3 Virtual Private Network (L3VPN)). Such modules can be mapped to network technology-specific modules at lower layers (e.g., tunnel, routing, Quality of Service (QoS), security). For example, the service level can be used to characterise the network service(s) to be ensured between service nodes (ingress/egress) such as:

- the communication scope (pipe, hose, funnel, ...),
- the directionality (inbound/outbound),
- the traffic performance guarantees (One-Way Delay (OWD) [RFC7679], One-Way Loss [RFC7680], ...),
- link capacity [RFC5136][I-D.ietf-ippm-capacity-metric-method],
- etc.

Figure 1 depicts the example of a VoIP service that relies upon connectivity services offered by a Network Operator. In this example, the VoIP service is offered to the Network Operator's customers by Service Provider (SP1). In order to provide global VoIP reachability, SP1 service site interconnects with other Service Providers service sites typically by interconnecting Session Border

Elements (SBEs) and Data Border Elements (DBEs) [RFC5486][RFC6406]. For other VoIP destinations, sessions are forwarded over the Internet. These connectivity services can be captured in a YANG Service Module that reflects the service attributes that are shown in Figure 2. This example follows the IP Connectivity Provisioning Profile template defined in [RFC7297].



**** (1) Inter-SP connectivity
xxxx (2) Customer to SP connectivity
oooo (3) SP to any destination connectivity

Figure 1: An Example of Service Connectivty Components

```
Connectivity: Scope and Guarantees
   (1) Inter-SP connectivity
      - Pipe scope from the local to the remote SBE/DBE
      - Full guarantees class
   (2) Customer to SP connectivity
      - Hose/Funnel scope connecting the local SBE/DBE
        to the customer access points
      - Full guarantees class
   (3) SP to any destination connectivity
      - Hose/Funnel scope from the local SBE/DBE to the
        Internet gateway
      - Delay guarantees class
Flow Identification
   * Destination IP address (SBE, DBE)
   * DSCP marking
Traffic Isolation
   * VPN
Routing & Forwarding
   * Routing rule to exclude some ASes from the inter-domain
     paths
Notifications (including feedback)
   * Statistics on aggregate traffic to adjust capacity
   * Failures
   * Planned maintenance operations
   * Triggered by thresholds
```

Figure 2: Sample Attributes Captured in a Service Model

Network Models are mainly network resource-facing modules; they describe various aspects of a network infrastructure, including devices and their subsystems, and relevant protocols operating at the link and network layers across multiple devices (e.g., network topology and traffic-engineering tunnel modules).

Device (and function) Models usually follow a bottom-up approach and are mostly technology-specific modules used to realize a service (e.g., BGP, NAT).

Each level maintains a view of the supported YANG modules provided by low-levels (see for example, Appendix A).

Figure 3 illustrates the overall layering model. The reader may refer to Section 4 of [RFC8309] for an overview of "Orchestrator" and "Controller" elements.

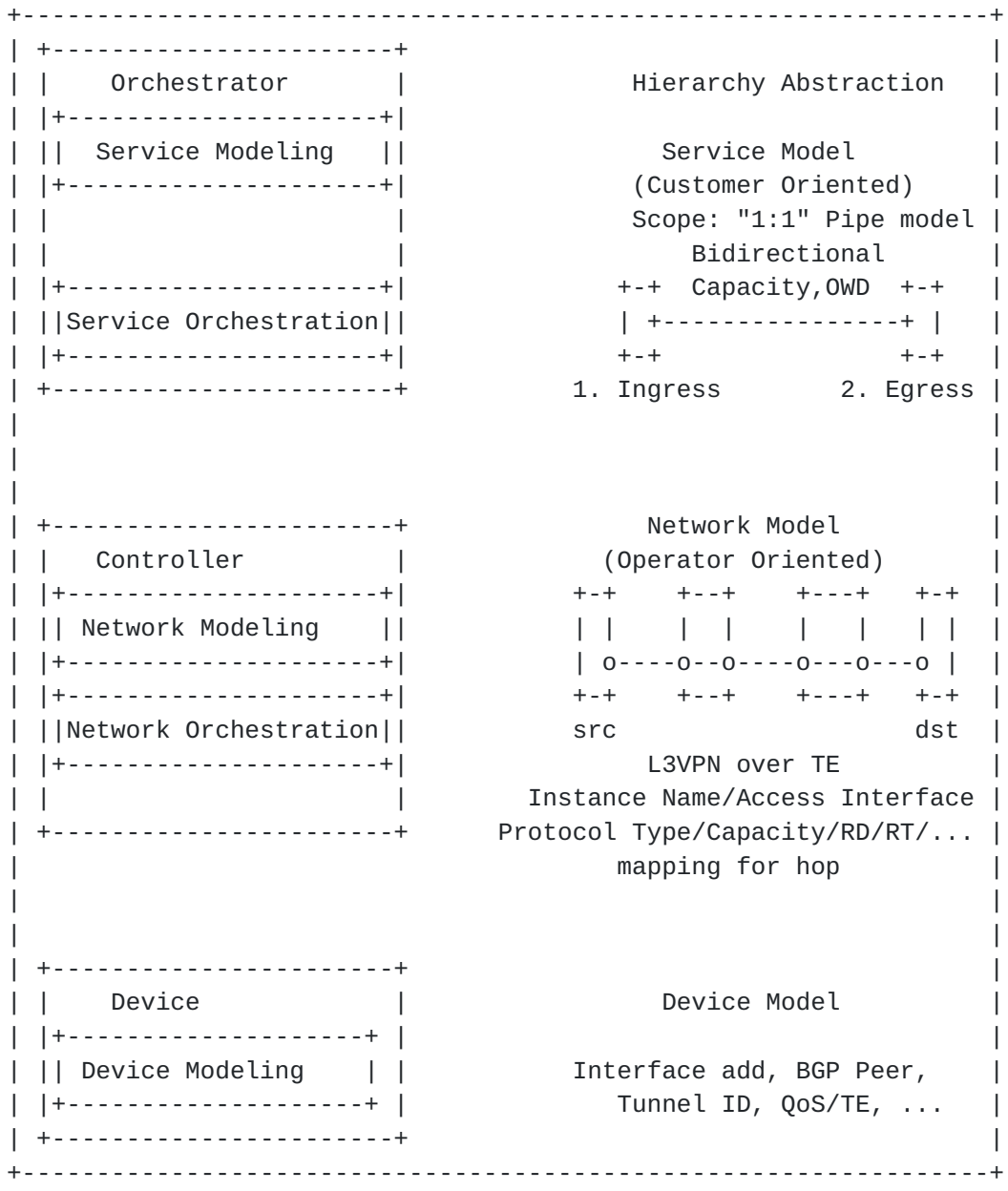


Figure 3: Layering and Representation

3.2. Automation of Service Delivery Procedures

Service Models can be used by a Network Operator to expose its services to its customers. Exposing such models allows to automate the activation of service orders and thus the service delivery. One or more monolithic Service Models can be used in the context of a composite service activation request (e.g., delivery of a caching infrastructure over a VPN). Such models are used to feed a decision-making intelligence to adequately accommodate customer's needs.

Also, such models may be used jointly with services that require dynamic invocation. An example is provided by the service modules defined by the DOTS WG to dynamically trigger requests to handle Distributed Denial-of-Service (DDoS) attacks [RFC8783].

Network Models can be derived from Service Models and used to provision, monitor, instantiate the service, and provide lifecycle management of network resources. Doing so is meant to:

- expose network resources to customers (including other Network Operators) to provide service fulfillment and assurance

- allow customers (or Network Operators) to dynamically adjust the network resources based on service requirements as described in Service Models (e.g., Figure 2) and the current network performance information described in the telemetry modules.

3.3. Service Fullfillment Automation

To operate a service, the settings of the parameters in the Device Models are derived from Service Models and/or Network Models and are used to:

- Provision each involved network function/device with the proper configuration information.

- Operate the network based on service requirements as described in the Service Model(s) and local operational guidelines.

In addition, the operational state including configuration that is in effect together with statistics should be exposed to upper layers to provide better network visibility and assess to what extent the derived low level modules are consistent with the upper level inputs.

Filters are enforced on the notifications that are communicated to Service layers. The type and frequency of notifications may be agreed in the Service Model.

Note that it is important to correlate telemetry data with configuration data to be used for closed loops at the different stages of service delivery, from resource allocation to service operation, in particular.

3.4. YANG Modules Integration

To support top-down service delivery, YANG modules at different levels or at the same level need to be integrated together for proper service delivery (including, proper network setup). For example, the

service parameters captured in Service Models need to be decomposed into a set of configuration/notification parameters that may be specific to one or more technologies; these technology-specific parameters are grouped together to define technology-specific device level models or network level models.

In addition, these technology-specific Device or Network Models can be further integrated with each other using the schema mount mechanism [RFC8528] to provision each involved network function/ device or each involved administrative domain to support newly added module or features. A collection of Device Models integrated together can be loaded and validated during the implementation time.

High-level policies can be defined at Service or Network Models (e.g., "Autonomous System Number (ASN) Exclude" in the example depicted in Figure 2). Device Models will be tweaked accordingly to provide policy-based management. Policies can also be used for telemetry automation, e.g., policies that contain conditions can trigger the generation and pushing of new telemetry data.

Performance measurement telemetry can be used to provide service assurance at Service and/or Network levels. Performance measurement telemetry model can tie with Service or Network Models to monitor network performance or Service Level Agreement.

4. Functional Bocks and Interactions

The architectural considerations described in Section 3 lead to the architecture described in this section and illustrated in Figure 4.

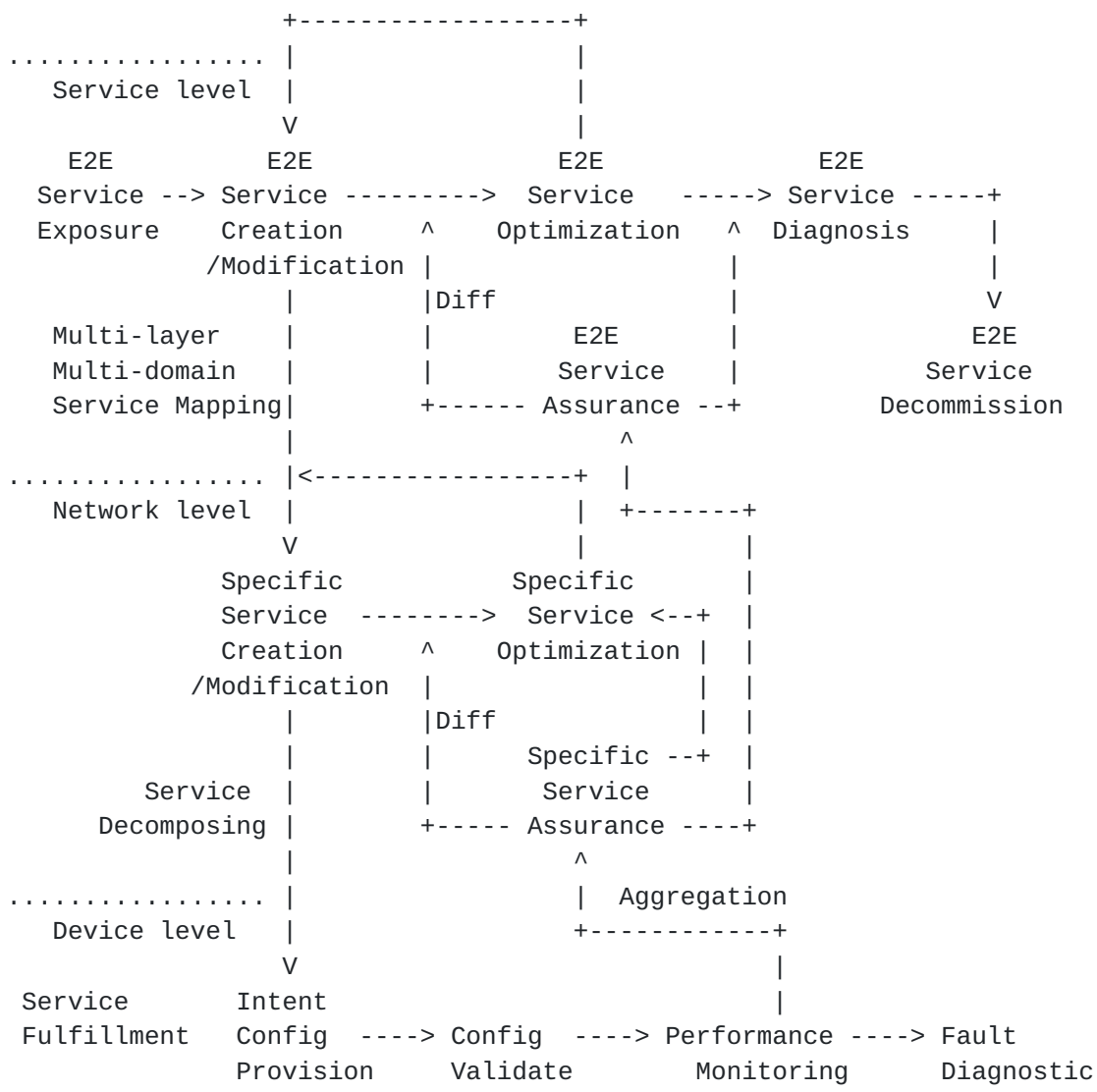


Figure 4: Service and Network Lifecycle Management

4.1. Service Lifecycle Management Procedure

Service lifecycle management includes end-to-end service lifecycle management at the service level and technology specific network lifecycle management at the network level.

The end-to-end service lifecycle management is technology-independent service management and spans across multiple administrative domain or multiple layers while technology specific service lifecycle management is technology domain specific or layer specific service lifecycle management.

4.1.1. Service Exposure

A service in the context of this document (sometimes called, Network Service) is some form of connectivity between customer sites and the Internet or between customer sites across the operator's network and across the Internet.

Service exposure is used to capture services offered to customers (ordering and order handling). One typical example is that a customer can use a L3VPN Service Model (L3SM) to request L3VPN service by providing the abstract technical characterization of the intended service between customer sites.

Service Model catalogs can be created along to expose the various services and the information needed to invoke/order a given service.

4.1.2. Service Creation/Modification

A customer is usually unaware of the technology that the Network Operator has available to deliver the service, so the customer does not make requests specific to the underlying technology but is limited to making requests specific to the service that is to be delivered. This service request can be issued using a Service Model.

Upon receiving a service request, and assuming that appropriate authentication and authorization checks have been made, the service orchestrator/management system should verify whether the service requirements in the service request can be met (i.e., whether there is sufficient resources that can be allocated with the requested guarantees).

If the request is accepted, the service orchestrator/management system maps such service request to its view. This view can be described as a technology specific network model or a set of technology specific Device Models and this mapping may include a choice of which networks and technologies to use depending on which service features have been requested.

In addition, a customer may require to change the underlying network infrastructure to adapt to new customer's needs and service requirements. This service modification can be issued following the same Service Model used by the service request.

4.1.3. Service Optimization

Service optimization is a technique that gets the configuration of the network updated due to network changes, incidents mitigation, or new service requirements. One typical example is once a tunnel or a

VPN is setup, Performance monitoring information or telemetry information per tunnel (or per VPN) can be collected and fed into the management system. If the network performance doesn't meet the service requirements, the management system can create new VPN policies capturing network service requirements and populate them into the network.

Both network performance information and policies can be modelled using YANG. With Policy-based management, self-configuration and self-optimization behavior can be specified and implemented.

4.1.4. Service Diagnosis

Operations, Administration, and Maintenance (OAM) are important networking functions for service diagnosis that allow Network Operators to:

- monitor network communications (i.e., reachability verification and Continuity Check)
- troubleshoot failures (i.e., fault verification and localization)
- monitor service-level agreements and performance (i.e., performance management)

When the network is down, service diagnosis should be in place to pinpoint the problem and provide recommendations (or instructions) for the network recovery.

The service diagnosis information can be modelled as technology-independent Remote Procedure Call (RPC) operations for OAM protocols and technology-independent abstraction of key OAM constructs for OAM protocols [RFC8531][RFC8533]. These models can be used to provide consistent configuration, reporting, and presentation for the OAM mechanisms used to manage the network.

4.1.5. Service Decommission

Service decommission allows a customer to stop the service by removing the service from active status and thus releasing the network resources that were allocated to the service. Customers can also use the Service Model to withdraw the registration to a service.

4.2. Service Fullfillment Management Procedure

4.2.1. Intended Configuration Provision

Intended configuration at the device level is derived from Network Models at the network level or Service Model at the service level and represents the configuration that the system attempts to apply. Take L3SM as a Service Model example to deliver a L3VPN service, we need to map the L3VPN service view defined in the Service Model into detailed intended configuration view defined by specific configuration models for network elements, configuration information includes:

- Virtual Routing and Forwarding (VRF) definition, including VPN policy expression
- Physical Interface(s)
- IP layer (IPv4, IPv6)
- QoS features such as classification, profiles, etc.
- Routing protocols: support of configuration of all protocols listed in a service request, as well as routing policies associated with those protocols.
- Multicast support
- Address sharing (e.g., NAT)
- Security

These specific configuration models can be used to configure Provider Edge (PE) and Customer Edge (CE) devices within a site, e.g., a BGP policy model can be used to establish VPN membership between sites and VPN Service Topology.

4.2.2. Configuration Validation

Configuration validation is used to validate intended configuration and ensure the configuration take effect.

For example, a customer creates an interface "eth-0/0/0" but the interface does not physically exist at this point, then configuration data appears in the <intended> status but does not appear in <operational> datastore.

4.2.3. Performance Monitoring/Model-driven Telemetry

When configuration is in effect in the device, <operational> datastore holds the complete operational state of the device including learned, system, default configuration, and system state. However, the configurations and state of a particular device does not have the visibility to the whole network or information of the flow packets are going to take through the entire network. Therefore it becomes more difficult to operate the network without understanding the current status of the network.

The management system should subscribe to updates of a YANG datastore in all the network devices for performance monitoring purpose and build a full topological visibility of the network by aggregating (and filtering) these operational state from different sources.

4.2.4. Fault Diagnostic

When configuration is in effect in the device, some devices may be mis-configured (e.g.,device links are not consistent in both sides of the network connection), network resources be mis-allocated and services may be negatively affected without knowing what is going on in the network.

Technology-dependent nodes and RPC commands are defined in technology-specific YANG data models which can use and extend the base model described in Section 4.1.4 to deal with these issues.

These RPC commands received in the technology-dependent node can be used to trigger technology-specific OAM message exchanges for fault verification and fault isolation For example, TRILL Multicast Tree Verification (MTV) RPC command [I-D.ietf-trill-yang-oam] can be used to trigger Multi-Destination Tree Verification Message defined in [RFC7455] to verify TRILL distribution tree integrity.

4.3. Multi-Layer/Multi-Domain Service Mapping

Multi-layer/Multi-domain Service Mapping allows to map an end-to-end abstract view of the service segmented at different layers or different administrative domains into domain-specific view.

One example is to map service parameters in L3VPN service model into configuration parameters such as Route Distinguisher (RD), Route Target (RT), and VRF in L3VPN network model.

Another example is to map service parameters in L3VPN service model into Traffic Engineered (TE) tunnel parameter (e.g., Tunnel ID) in TE model and Virtual Network (VN) parameters (e.g., Access Point (AP)

list, VN members) in the YANG data model for VN operation [I-D.ietf-teas-actn-vn-yang].

4.4. Service Decomposing

Service Decomposing allows to decompose service model at the service level or network model at the network level into a set of device/ function models at the device level. These Device Models may be tied to specific device types or classified into a collection of related YANG modules based on service types and features offered, and load at the implementation time before configuration is loaded and validated.

5. YANG Data Model Integration Examples

The following subsections provides some data models integration examples.

5.1. L2VPN/L3VPN Service Delivery

In reference to Figure 5, the following steps are performed to deliver the L3VPN service within the network management automation architecture defined in this document:

1. The Customer requests to create two sites (as per service creation operation in Section 4.2.1) relying upon a L3SM Service model with each having one network access connectivity, for example:

    * Site A: Network-Access A, Link Capacity = 20 Mbps, for class "foo", guaranteed-capacity-percent = 10, average-One-Way-Delay = 70 ms.

    * Site B: Network-Access B, Link Capacity = 30 Mbps, for class "foo1", guaranteed-capacity-percent = 15, average-One-Way-Delay = 60 ms.

2. The Orchestrator extracts the service parameters from the L3SM model. Then, it uses them as input to translate ("service mapping operation" in Section 4.4) them into an orchestrated configuration of network elements (e.g., RD, RT, VRF) that are part of the L3VPN Network YANG Model specified in [I-D.ietf-opsawg-l3sm-l3nm].

3. The Controller takes orchestrated configuration parameters in the L3NM network model and translates them into orchestrated ("service decomposing operation" in) configuration of network elements that are part of, e.g., BGP, QoS, Network Instance model, IP management, and interface models.

[I-D.ogondio-opsawg-uni-topology] can be used for representing, managing, and controlling the User Network Interface (UNI) topology.

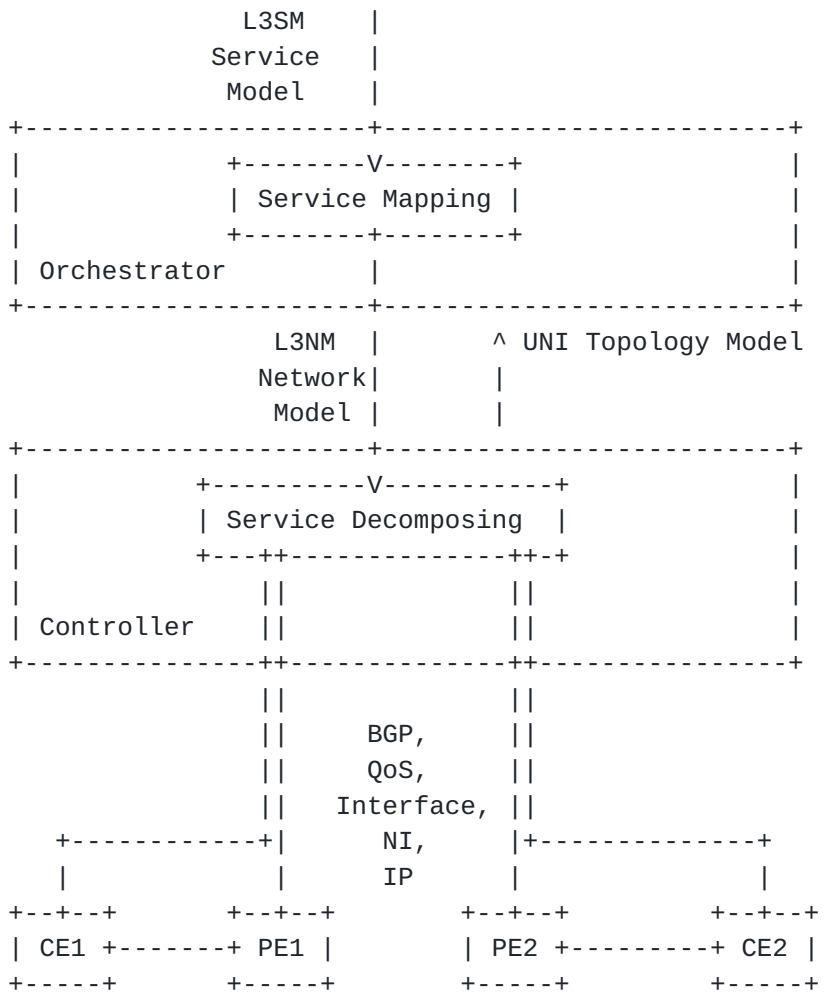


Figure 5: L3VPN Service Delivery Example (Current)

L3NM inherits some of data elements from the L3SM. Nevertheless, the L3NM does not expose some information to the above layer such as the capabilities of an underlying network (which can be used to drive service order handling) or notifications (to notify subscribers about specific events or degradations as per agreed SLAs). Some of this information can be provided using, e.g., [I-D.www-bess-yang-vpn-service-pm]. A target overall model is depicted in Figure 6.

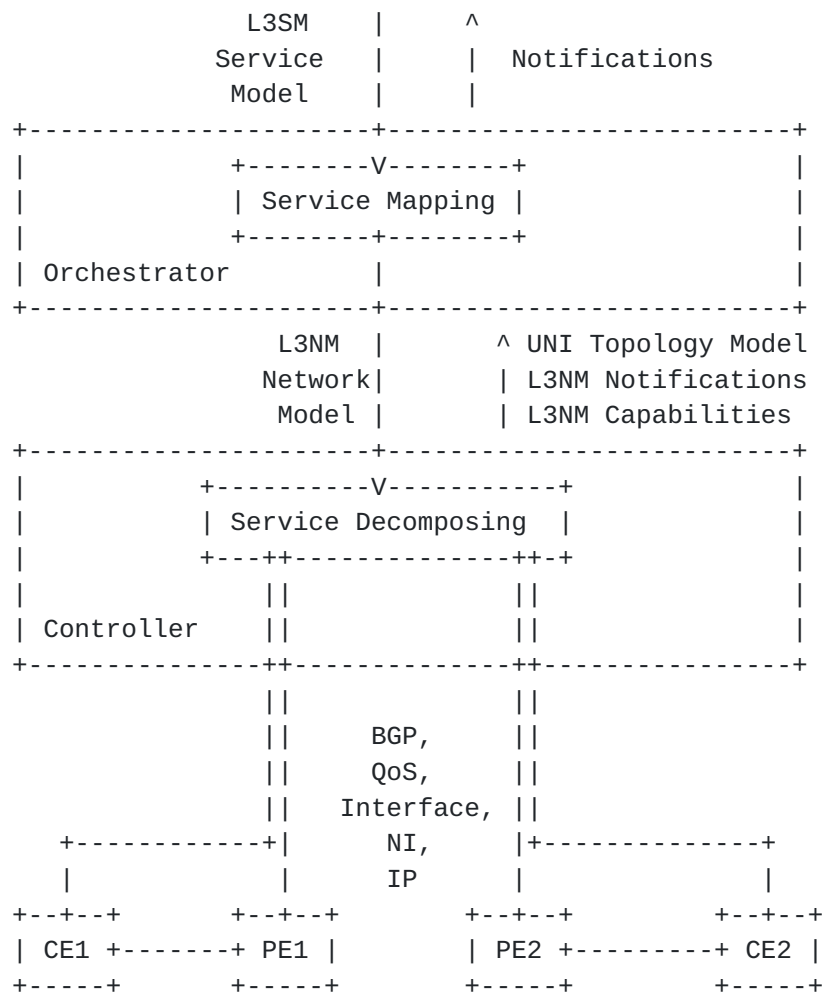


Figure 6: L3VPN Service Delivery Example (Target)

Note that a similar analysis can be performed for Layer 2 VPNs (L2VPNs). A L2VPN Service Model (L2SM) is defined in [RFC8466], while the L2VPN Network YANG Model (L2NM) is specified in [I-D.barguil-opsawg-l2sm-l2nm].

5.2. VN Lifecycle Management

In reference to Figure 7, the following steps are performed to deliver the VN service within the network management automation architecture defined in this document:

1. Customer requests (service exposure operation in Section 4.1.1) to create 'VN' based on Access point, association between VN and Access point, VN member defined in the VN YANG module.

2. The orchestrator creates the single abstract node topology based on the information captured in an VN YANG module.

3. The Customer exchanges connectivity-matrix on abstract node and explicit path using TE topology model with the orchestrator. This information can be used to instantiate VN and setup tunnels between source and destination endpoints (service creation operation in Section 4.1.2).

4. The telemetry model which augments the VN model and corresponding TE tunnel model can be used to subscribe to performance measurement data and notify all the parameter changes and network performance change related to VN topology or Tunnel [I-D.ietf-teas-actn-pm-telemetry-autonomics] and provide service assurance (service optimization operation in Section 4.1.3).

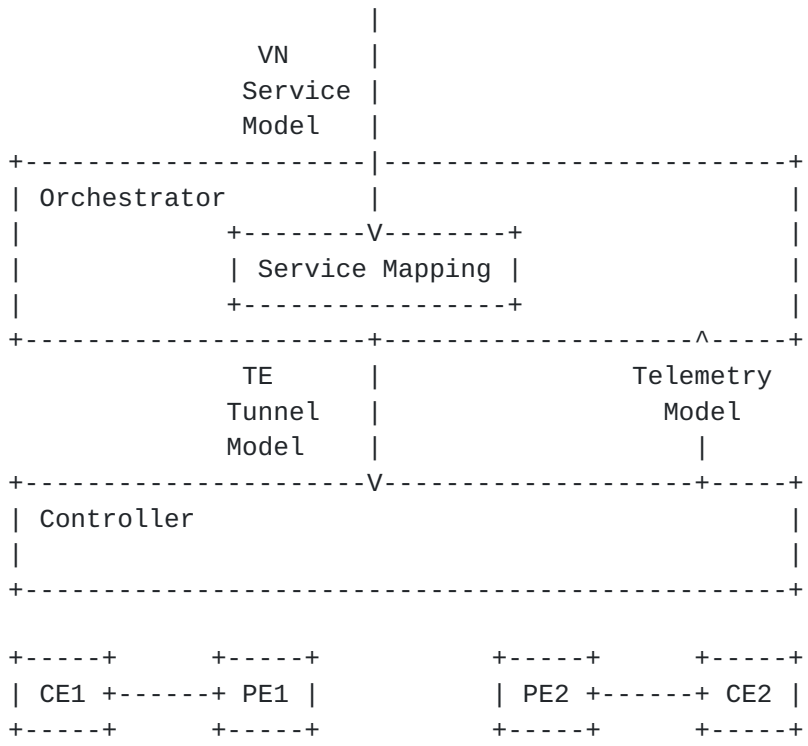


Figure 7: A VN Service Delivery Example

5.3. Event-based Telemetry in the Device Self Management

In reference to Figure 8, the following steps are performed to monitor state changes of managed objects or resources in a network device and provide device self-management within the network management automation architecture defined in this document:

1. To control which state a network device should be in or is allowed to be in at any given time, a set of conditions and actions are defined and correlated with network events (e.g., allow the NETCONF server to send updates only when the value

exceeds a certain threshold for the first time, but not again until the threshold is cleared), which constitute ECA policy or an event-driven policy control logic that can be executed on the device (e.g., [I-D.wwx-netmod-event-yang]).

2. To provide rapid autonomic response that can exhibit self-management properties, the controller pushes the ECA policy to the network device and delegates network control logic to the network device.

3. The network device uses the ECA model to subscribe to the event source, e.g., an event stream or datastore state data conveyed to the server via YANG Push subscription, monitors state parameters, and takes simple and instant actions when associated event condition on state parameters is met. ECA notifications can be generated as the result of actions based on event stream subscription or datastore subscription (model-driven telemetry operation discussed in Section 4.2.3).

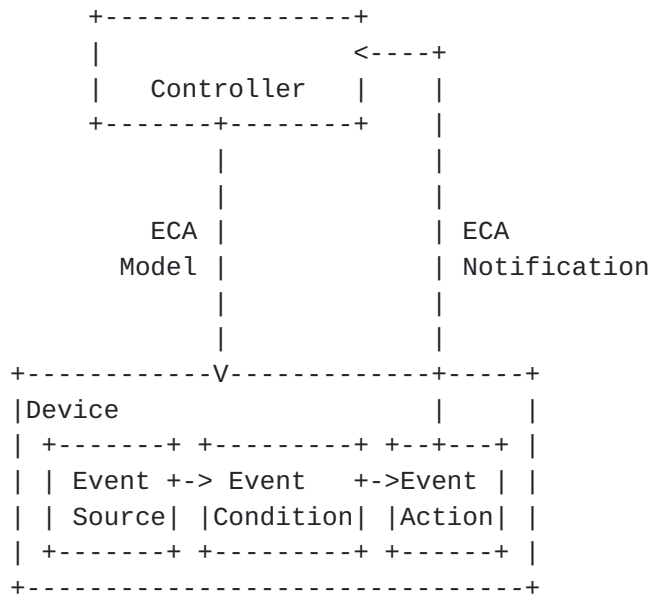


Figure 8: Event-based Telemetry

6. Security Considerations

The YANG modules cited in this document define schema for data that are designed to be accessed via network management protocols such as NETCONF [RFC6241] or RESTCONF [RFC8040]. The lowest NETCONF layer is the secure transport layer, and the mandatory-to-implement secure transport is Secure Shell (SSH) [RFC6242]. The lowest RESTCONF layer is HTTPS, and the mandatory-to-implement secure transport is TLS [RFC8446].

The NETCONF access control model [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

Security considerations specific to each of the technologies and protocols listed in the document are discussed in the specification documents of each of these protocols.

Security considerations specific to this document are listed below:

- Create forwarding loops by mis-configuring the underlying network.
- Leak sensitive information: special care should be considered when translating between the various layers in Section 4 or when aggregating data retrieved from various sources. The Network Operator must enforce means to protect privacy-related information included in cutsomer-facing models.
- Some Service Models may include a traffic isolation clause, appropriate technology-specific actions must be enforced to avoid that traffic is accessible to non-authorized parties.

7. IANA Considerations

There are no IANA requests or assignments included in this document.

8. Acknowledgements

Thanks to Joe Clark, Greg Mirsky, Shunsuke Homma, Brian Carpenter, and Adrian Farrel for the review.

9. Contributors

Christian Jacquenet
Orange
Rennes, 35000
France
Email: Christian.jacquenet@orange.com

Luis Miguel Contreras Murillo
Telifonica

Email: luismiguel.contrerasmurillo@telefonica.com

Oscar Gonzalez de Dios
Telefonica
Madrid
ES

Email: oscar.gonzalezdedios@telefonica.com

Weiqiang Cheng
China Mobile

Email: chengweiqiang@chinamobile.com

Young Lee
Sung Kyun Kwan University

Email: younglee.tx@gmail.com

10. References

10.1. Normative References

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", RFC 6242, DOI 10.17487/RFC6242, June 2011, <https://www.rfc-editor.org/info/rfc6242>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

10.2. Informative References

[I-D.barguil-opsawg-l2sm-l2nm]
Barguil, S., Dios, O., Boucadair, M., Munoz, L., Jalil, L., and J. Ma, "A Layer 2 VPN Network YANG Model", draft-barguil-opsawg-l2sm-l2nm-02 (work in progress), May 2020.

[I-D.ietf-bess-evpn-yang]
Brissette, P., Shah, H., Hussain, I., Tiruveedhula, K., and J. Rabadan, "Yang Data Model for EVPN", draft-ietf-bess-evpn-yang-07 (work in progress), March 2019.

[I-D.ietf-bess-l2vpn-yang]
Shah, H., Brissette, P., Chen, I., Hussain, I., Wen, B., and K. Tiruveedhula, "YANG Data Model for MPLS-based L2VPN", draft-ietf-bess-l2vpn-yang-10 (work in progress), July 2019.

[I-D.ietf-bess-l3vpn-yang]
Jain, D., Patel, K., Brissette, P., Li, Z., Zhuang, S., Liu, X., Haas, J., Esale, S., and B. Wen, "Yang Data Model for BGP/MPLS L3 VPNs", draft-ietf-bess-l3vpn-yang-04 (work in progress), October 2018.

[I-D.ietf-bess-mvpn-yang]
Liu, Y., Guo, F., Litkowski, S., Liu, X., Kebler, R., and M. Sivakumar, "Yang Data Model for Multicast in MPLS/BGP IP VPNs", draft-ietf-bess-mvpn-yang-02 (work in progress), December 2019.

[I-D.ietf-bfd-yang]
Rahman, R., Zheng, L., Jethanandani, M., Pallagatti, S., and G. Mirsky, "YANG Data Model for Bidirectional Forwarding Detection (BFD)", draft-ietf-bfd-yang-17 (work in progress), August 2018.

[I-D.ietf-i2rs-yang-l2-network-topology]
Dong, J., Wei, X., WU, Q., Boucadair, M., and A. Liu, "A YANG Data Model for Layer-2 Network Topologies", draft-ietf-i2rs-yang-l2-network-topology-13 (work in progress), March 2020.

[I-D.ietf-idr-bgp-model]
Jethanandani, M., Patel, K., Hares, S., and J. Haas, "BGP YANG Model for Service Provider Networks", draft-ietf-idr-bgp-model-08 (work in progress), February 2020.

[I-D.ietf-ippm-capacity-metric-method]
Morton, A., Geib, R., and L. Ciavattone, "Metrics and Methods for IP Capacity", draft-ietf-ippm-capacity-metric-method-01 (work in progress), March 2020.

[I-D.ietf-ippm-stamp-yang]
Mirsky, G., Xiao, M., and W. Luo, "Simple Two-way Active Measurement Protocol (STAMP) Data Model", draft-ietf-ippm-stamp-yang-05 (work in progress), October 2019.

[I-D.ietf-ippm-twamp-yang]
Civil, R., Morton, A., Rahman, R., Jethanandani, M., and K. Pentikousis, "Two-Way Active Measurement Protocol (TWAMP) Data Model", draft-ietf-ippm-twamp-yang-13 (work in progress), July 2018.

[I-D.ietf-mpls-base-yang]
Saad, T., Raza, K., Gandhi, R., Liu, X., and V. Beeram, "A YANG Data Model for MPLS Base", draft-ietf-mpls-base-yang-14 (work in progress), March 2020.

[I-D.ietf-opsawg-l3sm-l3nm]
Barguil, S., Dios, O., Boucadair, M., Munoz, L., and A. Aguado, "A Layer 3 VPN Network YANG Model", draft-ietf-opsawg-l3sm-l3nm-03 (work in progress), April 2020.

[I-D.ietf-pim-igmp-mld-snooping-yang]
Zhao, H., Liu, X., Liu, Y., Sivakumar, M., and A. Peter, "A Yang Data Model for IGMP and MLD Snooping", draft-ietf-pim-igmp-mld-snooping-yang-12 (work in progress), May 2020.

[I-D.ietf-pim-yang]
Liu, X., McAllister, P., Peter, A., Sivakumar, M., Liu, Y., and f. hu, "A YANG Data Model for Protocol Independent Multicast (PIM)", draft-ietf-pim-yang-17 (work in progress), May 2018.

[I-D.ietf-rtgwg-device-model]
Lindem, A., Berger, L., Bogdanovic, D., and C. Hopps, "Network Device YANG Logical Organization", draft-ietf-rtgwg-device-model-02 (work in progress), March 2017.

[I-D.ietf-rtgwg-policy-model]
Qu, Y., Tantsura, J., Lindem, A., and X. Liu, "A YANG Data Model for Routing Policy Management", draft-ietf-rtgwg-policy-model-15 (work in progress), June 2020.

[I-D.ietf-rtgwg-qos-model]
Choudhary, A., Jethanandani, M., Strahle, N., Aries, E., and I. Chen, "YANG Model for QoS", draft-ietf-rtgwg-qos-model-01 (work in progress), April 2020.

[I-D.ietf-spring-sr-yang]
Litkowski, S., Qu, Y., Lindem, A., Sarkar, P., and J. Tantsura, "YANG Data Model for Segment Routing", draft-ietf-spring-sr-yang-15 (work in progress), January 2020.

[I-D.ietf-supa-generic-policy-data-model]
Halpern, J. and J. Strassner, "Generic Policy Data Model for Simplified Use of Policy Abstractions (SUPA)", draft-ietf-supa-generic-policy-data-model-04 (work in progress), June 2017.

[I-D.ietf-teas-actn-pm-telemetry-autonomics]
Lee, Y., Dhody, D., Karunanithi, S., Vilata, R., King, D., and D. Ceccarelli, "YANG models for VN/TE Performance Monitoring Telemetry and Scaling Intent Autonomics", draft-ietf-teas-actn-pm-telemetry-autonomics-02 (work in progress), March 2020.

[I-D.ietf-teas-actn-vn-yang]
Lee, Y., Dhody, D., Ceccarelli, D., Bryskin, I., and B. Yoon, "A Yang Data Model for VN Operation", draft-ietf-teas-actn-vn-yang-08 (work in progress), March 2020.

[I-D.ietf-teas-yang-path-computation]
Busi, I., Belotti, S., Lopezalvarez, V., Sharma, A., and Y. Shi, "Yang model for requesting Path Computation", draft-ietf-teas-yang-path-computation-09 (work in progress), June 2020.

[I-D.ietf-teas-yang-rsvp-te]
Beeram, V., Saad, T., Gandhi, R., Liu, X., Bryskin, I., and H. Shah, "A YANG Data Model for RSVP-TE Protocol", draft-ietf-teas-yang-rsvp-te-08 (work in progress), March 2020.

[I-D.ietf-teas-yang-te]
Saad, T., Gandhi, R., Liu, X., Beeram, V., and I. Bryskin, "A YANG Data Model for Traffic Engineering Tunnels and Interfaces", draft-ietf-teas-yang-te-23 (work in progress), March 2020.

[I-D.ietf-teas-yang-te-topo]
Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., and O. Dios, "YANG Data Model for Traffic Engineering (TE) Topologies", draft-ietf-teas-yang-te-topo-22 (work in progress), June 2019.

[I-D.ietf-trill-yang-oam]
Kumar, D., Senevirathne, T., Finn, N., Salam, S., Xia, L., and H. Weiguo, "YANG Data Model for TRILL Operations, Administration, and Maintenance (OAM)", draft-ietf-trill-yang-oam-05 (work in progress), March 2017.

[I-D.ogondio-opsawg-uni-topology]
Dios, O., Barguil, S., WU, Q., and M. Boucadair, "A YANG Model for User-Network Interface (UNI) Topologies", draft-ogondio-opsawg-uni-topology-01 (work in progress), April 2020.

[I-D.www-bess-yang-vpn-service-pm]
WU, Q., Boucadair, M., Dios, O., Wen, B., Liu, C., and H. Xu, "A YANG Model for Network and VPN Service Performance Monitoring", draft-www-bess-yang-vpn-service-pm-06 (work in progress), April 2020.

[I-D.wwx-netmod-event-yang]
Birkholz, H., WU, Q., Bryskin, I., Liu, X., and B. Claise, "A YANG Data model for ECA Policy Management", draft-wwx-netmod-event-yang-07 (work in progress), May 2020.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC4664] Andersson, L., Ed. and E. Rosen, Ed., "Framework for Layer 2 Virtual Private Networks (L2VPNs)", RFC 4664, DOI 10.17487/RFC4664, September 2006, <https://www.rfc-editor.org/info/rfc4664>.

[RFC4761] Kompella, K., Ed. and Y. Rekhter, Ed., "Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling", RFC 4761, DOI 10.17487/RFC4761, January 2007, <https://www.rfc-editor.org/info/rfc4761>.

[RFC4762] Lasserre, M., Ed. and V. Kompella, Ed., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, DOI 10.17487/RFC4762, January 2007, <https://www.rfc-editor.org/info/rfc4762>.

[RFC5136] Chimento, P. and J. Ishac, "Defining Network Capacity", RFC 5136, DOI 10.17487/RFC5136, February 2008, <https://www.rfc-editor.org/info/rfc5136>.

[RFC5486] Malas, D., Ed. and D. Meyer, Ed., "Session Peering for Multimedia Interconnect (SPEERMINT) Terminology", RFC 5486, DOI 10.17487/RFC5486, March 2009, <https://www.rfc-editor.org/info/rfc5486>.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[RFC6406] Malas, D., Ed. and J. Livingood, Ed., "Session PEERing for Multimedia INTerconnect (SPEERMINT) Architecture", RFC 6406, DOI 10.17487/RFC6406, November 2011, <https://www.rfc-editor.org/info/rfc6406>.

[RFC7149] Boucadair, M. and C. Jacquenet, "Software-Defined Networking: A Perspective from within a Service Provider Environment", RFC 7149, DOI 10.17487/RFC7149, March 2014, <https://www.rfc-editor.org/info/rfc7149>.

[RFC7276] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", RFC 7276, DOI 10.17487/RFC7276, June 2014, <https://www.rfc-editor.org/info/rfc7276>.

[RFC7297] Boucadair, M., Jacquenet, C., and N. Wang, "IP Connectivity Provisioning Profile (CPP)", RFC 7297, DOI 10.17487/RFC7297, July 2014, <https://www.rfc-editor.org/info/rfc7297>.

[RFC7455] Senevirathne, T., Finn, N., Salam, S., Kumar, D., Eastlake 3rd, D., Aldrin, S., and Y. Li, "Transparent Interconnection of Lots of Links (TRILL): Fault Management", RFC 7455, DOI 10.17487/RFC7455, March 2015, <https://www.rfc-editor.org/info/rfc7455>.

[RFC7679] Almes, G., Kalidindi, S., Zekauskas, M., and A. Morton, Ed., "A One-Way Delay Metric for IP Performance Metrics (IPPM)", STD 81, RFC 7679, DOI 10.17487/RFC7679, January 2016, <https://www.rfc-editor.org/info/rfc7679>.

[RFC7680] Almes, G., Kalidindi, S., Zekauskas, M., and A. Morton, Ed., "A One-Way Loss Metric for IP Performance Metrics (IPPM)", STD 82, RFC 7680, DOI 10.17487/RFC7680, January 2016, <https://www.rfc-editor.org/info/rfc7680>.

[RFC8077] Martini, L., Ed. and G. Heron, Ed., "Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)", STD 84, RFC 8077, DOI 10.17487/RFC8077, February 2017, <https://www.rfc-editor.org/info/rfc8077>.

[RFC8194] Schoenwaelder, J. and V. Bajpai, "A YANG Data Model for LMAP Measurement Agents", RFC 8194, DOI 10.17487/RFC8194, August 2017, <https://www.rfc-editor.org/info/rfc8194>.

[RFC8199] Bogdanovic, D., Claise, B., and C. Moberg, "YANG Module Classification", RFC 8199, DOI 10.17487/RFC8199, July 2017, <https://www.rfc-editor.org/info/rfc8199>.

[RFC8299] Wu, Q., Ed., Litkowski, S., Tomotaki, L., and K. Ogaki, "YANG Data Model for L3VPN Service Delivery", RFC 8299, DOI 10.17487/RFC8299, January 2018, <https://www.rfc-editor.org/info/rfc8299>.

[RFC8309] Wu, Q., Liu, W., and A. Farrel, "Service Models Explained", RFC 8309, DOI 10.17487/RFC8309, January 2018, <https://www.rfc-editor.org/info/rfc8309>.

[RFC8328] Liu, W., Xie, C., Strassner, J., Karagiannis, G., Klyus, M., Bi, J., Cheng, Y., and D. Zhang, "Policy-Based Management Framework for the Simplified Use of Policy Abstractions (SUPA)", RFC 8328, DOI 10.17487/RFC8328, March 2018, <https://www.rfc-editor.org/info/rfc8328>.

[RFC8345] Clemm, A., Medved, J., Varga, R., Bahadur, N., Ananthakrishnan, H., and X. Liu, "A YANG Data Model for Network Topologies", RFC 8345, DOI 10.17487/RFC8345, March 2018, <https://www.rfc-editor.org/info/rfc8345>.

[RFC8346] Clemm, A., Medved, J., Varga, R., Liu, X., Ananthakrishnan, H., and N. Bahadur, "A YANG Data Model for Layer 3 Topologies", RFC 8346, DOI 10.17487/RFC8346, March 2018, <https://www.rfc-editor.org/info/rfc8346>.

[RFC8349] Lhotka, L., Lindem, A., and Y. Qu, "A YANG Data Model for Routing Management (NMDA Version)", RFC 8349, DOI 10.17487/RFC8349, March 2018, <https://www.rfc-editor.org/info/rfc8349>.

[RFC8466] Wen, B., Fioccola, G., Ed., Xie, C., and L. Jalil, "A YANG Data Model for Layer 2 Virtual Private Network (L2VPN) Service Delivery", RFC 8466, DOI 10.17487/RFC8466, October 2018, <https://www.rfc-editor.org/info/rfc8466>.

[RFC8512] Boucadair, M., Ed., Sivakumar, S., Jacquenet, C., Vinapamula, S., and Q. Wu, "A YANG Module for Network Address Translation (NAT) and Network Prefix Translation (NPT)", RFC 8512, DOI 10.17487/RFC8512, January 2019, <https://www.rfc-editor.org/info/rfc8512>.

[RFC8513] Boucadair, M., Jacquenet, C., and S. Sivakumar, "A YANG Data Model for Dual-Stack Lite (DS-Lite)", RFC 8513, DOI 10.17487/RFC8513, January 2019, <https://www.rfc-editor.org/info/rfc8513>.

[RFC8519] Jethanandani, M., Agarwal, S., Huang, L., and D. Blair, "YANG Data Model for Network Access Control Lists (ACLs)", RFC 8519, DOI 10.17487/RFC8519, March 2019, <https://www.rfc-editor.org/info/rfc8519>.

[RFC8528] Bjorklund, M. and L. Lhotka, "YANG Schema Mount", RFC 8528, DOI 10.17487/RFC8528, March 2019, <https://www.rfc-editor.org/info/rfc8528>.

[RFC8529] Berger, L., Hopps, C., Lindem, A., Bogdanovic, D., and X. Liu, "YANG Data Model for Network Instances", RFC 8529, DOI 10.17487/RFC8529, March 2019, <https://www.rfc-editor.org/info/rfc8529>.

[RFC8530] Berger, L., Hopps, C., Lindem, A., Bogdanovic, D., and X. Liu, "YANG Model for Logical Network Elements", RFC 8530, DOI 10.17487/RFC8530, March 2019, <https://www.rfc-editor.org/info/rfc8530>.

[RFC8531] Kumar, D., Wu, Q., and Z. Wang, "Generic YANG Data Model for Connection-Oriented Operations, Administration, and Maintenance (OAM) Protocols", RFC 8531, DOI 10.17487/RFC8531, April 2019, <https://www.rfc-editor.org/info/rfc8531>.

[RFC8532] Kumar, D., Wang, Z., Wu, Q., Ed., Rahman, R., and S. Raghavan, "Generic YANG Data Model for the Management of Operations, Administration, and Maintenance (OAM) Protocols That Use Connectionless Communications", RFC 8532, DOI 10.17487/RFC8532, April 2019, <https://www.rfc-editor.org/info/rfc8532>.

[RFC8533] Kumar, D., Wang, M., Wu, Q., Ed., Rahman, R., and S. Raghavan, "A YANG Data Model for Retrieval Methods for the Management of Operations, Administration, and Maintenance (OAM) Protocols That Use Connectionless Communications", RFC 8533, DOI 10.17487/RFC8533, April 2019, <https://www.rfc-editor.org/info/rfc8533>.

[RFC8632] Vallin, S. and M. Bjorklund, "A YANG Data Model for Alarm Management", RFC 8632, DOI 10.17487/RFC8632, September 2019, <https://www.rfc-editor.org/info/rfc8632>.

[RFC8652] Liu, X., Guo, F., Sivakumar, M., McAllister, P., and A. Peter, "A YANG Data Model for the Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD)", RFC 8652, DOI 10.17487/RFC8652, November 2019, <https://www.rfc-editor.org/info/rfc8652>.

[RFC8675] Boucadair, M., Farrer, I., and R. Asati, "A YANG Data Model for Tunnel Interface Types", RFC 8675, DOI 10.17487/RFC8675, November 2019, <https://www.rfc-editor.org/info/rfc8675>.

[RFC8676] Farrer, I., Ed. and M. Boucadair, Ed., "YANG Modules for IPv4-in-IPv6 Address plus Port (A+P) Softwires", RFC 8676, DOI 10.17487/RFC8676, November 2019, <https://www.rfc-editor.org/info/rfc8676>.

[RFC8783] Boucadair, M., Ed. and T. Reddy.K, Ed., "Distributed Denial-of-Service Open Threat Signaling (DOTS) Data Channel Specification", RFC 8783, DOI 10.17487/RFC8783, May 2020, <https://www.rfc-editor.org/info/rfc8783>.

Appendix A. Layered YANG Modules Examples Overview

This appendix lists a set of data models that can be used for the delivery of connectivity services. These models can be classified as Service, Network, or Device Models.

It is not the intent of this appendix to provide an inventory of tools and mechanisms used in specific network and service management domains; such inventory can be found in documents such as [RFC7276].

A.1. Service Models: Definition and Samples

As described in [RFC8309], the service is "some form of connectivity between customer sites and the Internet and/or between customer sites across the Network Operator's network and across the Internet". More concretely, an IP connectivity service can be defined as the IP transfer capability characterized by a (Source Nets, Destination Nets, Guarantees, Scope) tuple where "Source Nets" is a group of unicast IP addresses, "Destination Nets" is a group of IP unicast and/or multicast addresses, and "Guarantees" reflects the guarantees (expressed in terms of QoS, performance, and availability, for example) to properly forward traffic to the said "Destination" [RFC7297].

For example:

- The L3SM model [RFC8299] defines the L3VPN service ordered by a customer from a Network Operator.

- The L2SM model [RFC8466] defines the L2VPN service ordered by a customer from a Network Operator.

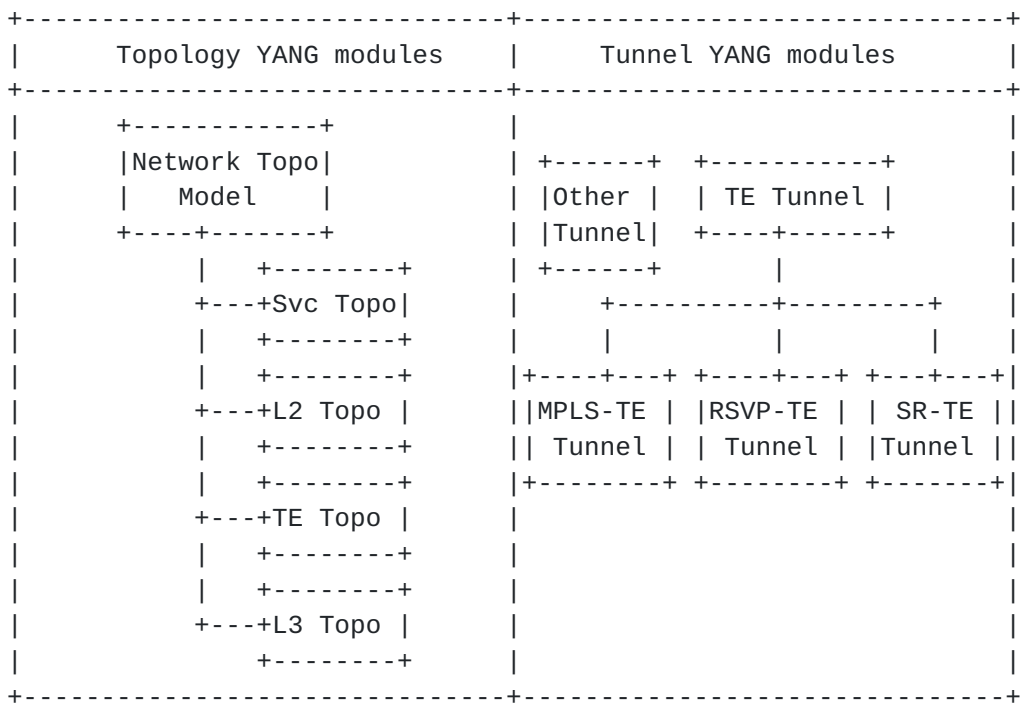
- The Virtual Network (VN) model [I-D.ietf-teas-actn-vn-yang] provides a YANG data model applicable to any mode of VN operation.

L2SM and L3SM are customer service models as per [RFC8309].

A.2. Network Models: Samples

L2NM [I-D.barguil-opsawg-l2sm-l2nm] and L3NM [I-D.ietf-opsawg-l3sm-l3nm] are examples of YANG Network Models.

Figure 9 depicts a set of additional Network Models such as topology and tunnel models:



Legend:
Topo: Topology
Svc: Service

Figure 9: Sample Resource Facing Network Models

Examples of topology YANG modules are listed below:

- Network Topology Models: [RFC8345] defines a base model for network topology and inventories. Network topology data include link resource, node resource, and terminate-point resources.

- TE Topology Models: [I-D.ietf-teas-yang-te-topo] defines a data model for representing and manipulating TE topologies.

  This module is extended from network topology model defined in [RFC8345] with TE topologies specifics. This model contains technology-agnostic TE Topology building blocks that can be augmented and used by other technology-specific TE topology models.

- Layer 3 Topology Models:

  [RFC8346] defines a data model for representing and manipulating Layer 3 topologies. This model is extended from the network topology model defined in [RFC8345] with Layer 3 topologies specifics.

- Layer 2 Topology Models:

  [I-D.ietf-i2rs-yang-l2-network-topology] defines a data model for representing and manipulating Layer 2 topologies. This model is extended from the network topology model defined in [RFC8345] with Layer 2 topologies specifics.

Examples of tunnel YANG modules are provided below:

- Tunnel identities to ease manipulating extensions to specific tunnels [RFC8675].

- TE Tunnel Model:

  [I-D.ietf-teas-yang-te] defines a YANG module for the configuration and management of TE interfaces, tunnels, and LSPs.

- Segment Routing (SR) Traffic Engineering (TE) Tunnel Model:

  [I-D.ietf-teas-yang-te] augments the TE generic and MPLS-TE model(s) and defines a YANG module for SR-TE specific data.

- MPLS-TE Model:

  [I-D.ietf-teas-yang-te] augments the TE generic and MPLS-TE model(s) and defines a YANG module for MPLS-TE configurations, state, RPC and notifications.

- RSVP-TE MPLS Model:

  [I-D.ietf-teas-yang-rsvp-te] augments the RSVP-TE generic module with parameters to configure and manage signaling of MPLS RSVP-TE LSPs.

Other sample Network Models are listed hereafter:

- Path Computation API Model:

  [I-D.ietf-teas-yang-path-computation] YANG module for a stateless RPC which complements the stateful solution defined in [I-D.ietf-teas-yang-te].

- OAM Models (including Fault Management (FM) and Performance Monitoring):

  [RFC8532] defines a base YANG module for the management of OAM protocols that use Connectionless Communications. [RFC8533] defines a retrieval method YANG module for connectionless OAM

protocols. [RFC8531] defines a base YANG module for connection oriented OAM protocols. These three models are intended to provide consistent reporting, configuration, and representation for connection-less OAM and Connection oriented OAM separately.

Alarm monitoring is a fundamental part of monitoring the network. Raw alarms from devices do not always tell the status of the network services or necessarily point to the root cause. [RFC8632] defines a YANG module for alarm management.

- Generic Policy Model:

  The Simplified Use of Policy Abstractions (SUPA) policy-based management framework [RFC8328] defines base YANG modules [I-D.ietf-supa-generic-policy-data-model] to encode policy. These models point to other device-, technology-, and service-specific YANG modules. Policy rules within an operator's environment can be used to express high-level, possibly network-wide, policies to a network management function (within a controller, an orchestrator, or a network element). The network management function can then control the configuration and/or monitoring of network elements and services. This document describes the SUPA basic framework, its elements, and interfaces.

A.3. Device Models: Samples

Network Element models (Figure 10) are used to describe how a service can be implemented by activating and tweaking a set of functions (enabled in one or multiple devices, or hosted in cloud infrastructures) that are involved in the service delivery. Figure 10 uses IETF-defined models as an example.

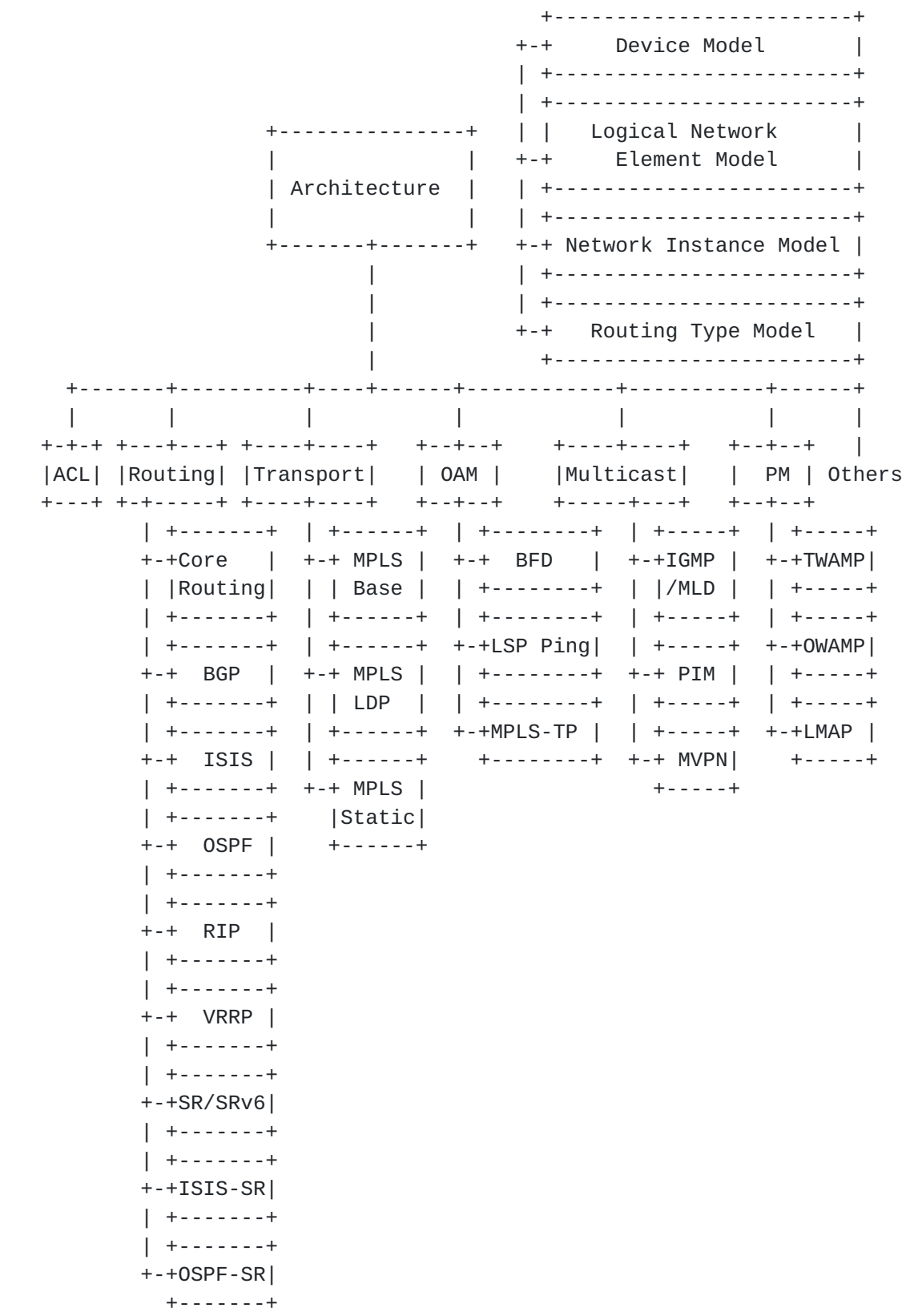


Figure 10: Network Element Modules Overview

A.3.1. Model Composition

- Device Model

  [I-D.ietf-rtgwg-device-model] presents an approach for organizing YANG modules in a comprehensive logical structure that may be used to configure and operate network devices. The structure is itself represented as an example YANG module, with all of the related component models logically organized in a way that is operationally intuitive, but this model is not expected to be implemented.

- Logical Network Element Model

  [RFC8530] defines a logical network element module which can be used to manage the logical resource partitioning that may be present on a network device. Examples of common industry terms for logical resource partitioning are Logical Systems or Logical Routers.

- Network Instance Model

  [RFC8529] defines a network instance module. This module can be used to manage the virtual resource partitioning that may be present on a network device. Examples of common industry terms for virtual resource partitioning are VRF instances and Virtual Switch Instances (VSIs).

A.3.1.1. Schema Mount

Modularity and extensibility were among the leading design principles of the YANG data modeling language. As a result, the same YANG module can be combined with various sets of other modules and thus form a data model that is tailored to meet the requirements of a specific use case. [RFC8528] defines a mechanism, denoted schema mount, that allows for mounting one data model consisting of any number of YANG modules at a specified location of another (parent) schema.

That capability does not cover design time.

A.3.2. Device Models: Samples

The following provides an overview of some Device Models that can be used within a network. This list is not comprehensive.

BGP: [I-D.ietf-idr-bgp-model] defines a YANG module for configuring and managing BGP, including protocol, policy,

and operational aspects based on data center, carrier, and content provider operational requirements.

MPLS: [I-D.ietf-mpls-base-yang] defines a base model for MPLS which serves as a base framework for configuring and managing an MPLS switching subsystem. It is expected that other MPLS technology YANG modules (e.g., MPLS LSP Static, LDP, or RSVP-TE models) will augment the MPLS base YANG module.

QoS: [I-D.ietf-rtgwg-qos-model] describes a YANG module of Differentiated Services for configuration and operations.

ACL: Access Control List (ACL) is one of the basic elements used to configure device forwarding behavior. It is used in many networking technologies such as Policy Based Routing, Firewalls, etc. [RFC8519] describes a data model of ACL basic building blocks.

NAT: For the sake of network automation and the need for programming Network Address Translation (NAT) function in particular, a data model for configuring and managing the NAT is essential.

[RFC8512] defines a YANG module for the NAT function covering a variety of NAT flavors such as Network Address Translation from IPv4 to IPv4 (NAT44), Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers (NAT64), customer-side translator (CLAT), Stateless IP/ICMP Translation (SIIT), Explicit Address Mappings (EAM) for SIIT, IPv6-to-IPv6 Network Prefix Translation (NPTv6), and Destination NAT.

[RFC8513] specifies a DS-Lite YANG module.

Stateless Address Sharing: [RFC8676] specifies a YANG module for A+P address sharing, including Lightweight 4over6, Mapping of Address and Port with Encapsulation (MAP-E), and Mapping of Address and Port using Translation (MAP-T) softwire mechanisms.

Multicast: [I-D.ietf-pim-yang] defines a YANG module that can be used to configure and manage Protocol Independent Multicast (PIM) devices.

[RFC8652] defines a YANG module that can be used to configure and manage Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) devices.

[I-D.ietf-pim-igmp-mld-snooping-yang] defines a YANG module that can be used to configure and manage Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping devices.

[I-D.ietf-bess-mvpn-yang] defines a YANG data model to configure and manage Multicast in MPLS/BGP IP VPNs (MVPNs).

EVPN: [I-D.ietf-bess-evpn-yang] defines a YANG module for Ethernet VPN services. The model is agnostic of the underlay. It applies to MPLS as well as to VxLAN encapsulation. The module is also agnostic to the services, including E-LAN, E-LINE, and E-TREE services.

L3VPN: [I-D.ietf-bess-l3vpn-yang] defines a YANG module that can be used to configure and manage BGP L3VPNs [RFC4364]. It contains VRF specific parameters as well as BGP specific parameters applicable for L3VPNs.

L2VPN: [I-D.ietf-bess-l2vpn-yang] defines a YANG module for MPLS based Layer 2 VPN services (L2VPN) [RFC4664] and includes switching between the local attachment circuits. The L2VPN model covers point-to-point VPWS and Multipoint VPLS services. These services use signaling of Pseudowires across MPLS networks using LDP [RFC8077][RFC4762] or BGP [RFC4761].

Routing Policy: [I-D.ietf-rtgwg-policy-model] defines a YANG module for configuring and managing routing policies based on operational practice. The module provides a generic policy framework which can be augmented with protocol-specific policy configuration.

BFD: Bidirectional Forwarding Detection (BFD) [RFC5880] is a network protocol which is used for liveness detection of arbitrary paths between systems. [I-D.ietf-bfd-yang] defines a YANG module that can be used to configure and manage BFD.

SR/SRv6: [I-D.ietf-spring-sr-yang] a YANG module for segment routing configuration and operation.

Core Routing: [RFC8349] defines the core routing data model, which is intended as a basis for future data model development covering more-sophisticated routing systems. It is expected that other Routing technology YANG modules (e.g.,

VRRP, RIP, ISIS, OSPF models) will augment the Core Routing base YANG module.

PM: [I-D.ietf-ippm-twamp-yang] defines a data model for client and server implementations of the Two-Way Active Measurement Protocol (TWAMP).

[I-D.ietf-ippm-stamp-yang] defines the data model for implementations of Session-Sender and Session-Reflector for Simple Two-way Active Measurement Protocol (STAMP) mode using YANG.

[RFC8194] defines a data model for Large-Scale Measurement Platforms (LMAPs).

Authors' Addresses

Qin Wu (editor)
Huawei
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012
China

Email: bill.wu@huawei.com

Mohamed Boucadair (editor)
Orange
Rennes 35000
France

Email: mohamed.boucadair@orange.com

Diego R. Lopez
Telefonica I+D
Spain

Email: diego.r.lopez@telefonica.com

Chongfeng Xie
China Telecom
Beijing
China

Email: xiechf@chinatelecom.cn

Liang Geng
China Mobile

Email: gengliang@chinamobile.com