



Network Programmability and Automation in Optical Networks

Ricard Vilalta, Ramon Casellas, Ricardo Martínez, Raul Muñoz

► To cite this version:

Ricard Vilalta, Ramon Casellas, Ricardo Martínez, Raul Muñoz. Network Programmability and Automation in Optical Networks. 23th International IFIP Conference on Optical Network Design and Modeling (ONDM), May 2019, Athens, Greece. pp.223-234, 10.1007/978-3-030-38085-4_20 . hal-03200653

HAL Id: hal-03200653

<https://inria.hal.science/hal-03200653>

Submitted on 16 Apr 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Network Programmability and Automation in Optical Networks

Ricard Vilalta¹[0000–0003–0391–9728], Ramon Casellas¹[0000–0002–2663–6571],
Ricardo Martínez¹[0000–0003–3097–5485], and Raul Muñoz¹[0000–0003–4651–4499]

Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA),
Castelldefels, Spain. ricard.vilalta@cttc.es

Abstract. During last years, novel protocols and data models are arising to control and monitor optical network equipment. These protocols enable network programmability and automation by fulfilling the vision introduced by Software Defined Networking (SDN).

This paper offers an overview and hands-on experience on programming the necessary tools to control and monitor the network equipment. These tools introduce the necessary network programmability that will enable its automation, following current Zero-Touch service control and management. Data modelling languages are firstly explored. Later, protocols for control and monitoring are presented. Moreover, common standards data models are discussed. Finally, an evaluation of the different presented protocols and data models is provided, which includes recommendations for their selection and deployment.

Keywords: data models · protocols · optical networking

1 Introduction

To understand properly the evolution of control, management, and orchestration of Optical Networks, the authors in [1] suggest that the main trigger for this network evolution is the need for an automated networks, which is able to auto-configure, auto-deploy and auto-heal.

In order to fulfil this view, novel market trends can be easily identified. Basically, three main drivers are foreseen: novel kinds of business consumer and end consumers, and a new kind of a service provider. New business consumer request private and public clouds, in order to provide them with the necessary elastic compute and storage. End consumers are currently living in the cloud, consuming cloud applications such as streaming downloads, thus, driving new network loads. Novel Service Providers are proposing novel services such as software as a service (SaaS), IaaS, PaaS, Multi-tenancy, elastic compute and storage. In view of these trends, it can be concluded that a novel programmable and automated network is required.

This network programmability brings new innovations in optical transports. It opens the the door for new service models. Moreover, it also bring in service differentiation through new applications. Network programmability also allows

simplified architectures that provide both integrated E2E/Multi-layer service creation, as well as, automatic reaction on errors or changes. Moreover, it also provides some financial benefits, such as both efficient service setup (OPEX), and fast return of investment (ROI) utilization (CAPEX).

This paper proposes novel control and monitoring mechanisms for network programmability. It is important to keep in mind that there are some metrics of success. On one hand, the industry adoption of these mechanisms will determine their success. On the other hand, multi-vendor interoperability will also be important for their success.

This paper is structured as follows. Firstly presents an overview of available data modelling languages: YANG and Protocol Buffers data modeling languages. Later, protocols for control and monitoring are exposed, including NETCONF, RESTconf and gRPC protocols. Later, several data models defined in Standards Defining Organizations (SDO) are presented. These include ONF Transport API, IETF ACTN, OpenROADM and OpenConfig. Finally, presented data modelling languages, protocols and data models are compared and discussed in order to provide guidance of deployment of network programmability.

2 Data modeling languages for network programmability and automation

It has been currently detailed the need for better management frameworks and protocols for optical network control and management. Some examples of management tasks are definition of routing policies, configuration of routing peers, which are performed by the operator and can be automated. The configuration management can be properly modelled using a unified information and data modelling language to describe network element or system capabilities, attributes, operations and notifications.

In general, an information model describes the device capabilities, in terms of operations and configurable parameters, using high level abstractions without specific details on aspects such as a particular syntax or encoding. On the other hand, a data model determines the structure, syntax and semantics of the data that is externally visible.

2.1 YANG language

YANG is a data modeling language, initially conceived to model configuration and state data for network devices. Models define the device configurations & notifications, capture semantic details and are easy to understand [2].

During last years, adoption of YANG as a data modelling language across standard defining organizations (SDO) and open source software (OSS) projects has been notable. IETF, ETSI, ONF, MEF, 3GPP, among others have adopted this data modelling language in order to provide standard data models.

A Yang model includes a header, imports and include statements, type definitions, configurations and operational data declarations as well as actions (remote procedure calls - RPC) and notifications. Configuration and operational

data are structured into data trees (datastores), using containers and lists. Conditionals can be used to the support of optional features. An important ability of the language is the capacity to extend and constrain existing models (by inheritance/augmentation), resulting in a hierarchy of models, which might derive from abstracted models to more detailed and specific ones.

If a connection service YANG model is described, typically the module connection imports topology YANG file. The data tree consists of a list named connection that includes connection grouping elements. Each connection grouping includes several connection leaves (e.g., connectionId, source-node, target-node). Each leaf is determined by its type (e.g., string, uint32, leafref - a reference to another leaf of the data tree).

YANG data modelling language has open source tools which help the validation and the ease of programmability with the related models. The most significant is pyang [3], which is an extensible YANG validator. Pyang provides a framework for plugins that can convert YANG modules to other formats. Some examples of the possible outputs are UML diagrams, or tree diagrams, and encoded messages in XML or JSON. Pyangbind is one of the extension plugins [4] that generates a Python class hierarchy from a YANG data model. The resulting classes can be directly interacted with in Python.

2.2 Protocol Buffers

Protocol buffers (also known as protobuf) are a flexible, efficient, automated mechanism for serializing structured data in a byte-oriented protocol becoming smaller, faster, and simpler [5].

Protobufs, similarly to YANG, include an open source software tool-chain that allows special generated source code to easily write and read the structured data to and from a variety of data streams and using a variety of languages.

As you can see, the message format is simple – each message type has one or more uniquely numbered fields, and each field has a name and a value type, where value types can be numbers, booleans, strings, or other protocol buffer message types, allowing to structure data hierarchically. Optional fields can be specified, as well as required fields or repeated fields.

You specify how you want the information you're serializing to be structured by defining protocol buffer message types in .proto files. Each protocol buffer message is a small logical record of information, containing a series of name-value pairs.

3 Protocols for network programmability and automation

Once the most significant data modelling languages have been presented, it is turn to specify which protocols are suitable to use the necessary models for handling optical network elements and services. YANG data models can be properly configured using NETCONF and RESTconf protocols, while Protocol Buffers are related with gRPC protocol. Each subsection includes some examples of usage

in state of the art and from the authors detailing the usage of the proposed protocols.

3.1 NETCONF

NETCONF is a protocol based on the exchange of XML-encoded RPC messages over a secure (commonly Secure Shell, SSH) connection [6]. It offers primitives to view and manipulate data, providing a suitable encoding as defined by the data-model. As it use YANG models, data is arranged into one or multiple configuration datastores and provides the set of rules by which multiple clients may access and modify a datastore within a NETCONF server.

Figure 1 shows the NETCONF layering. Over a secure transport layer, a message layer for RPC messages or notifications is included. On top, an operations layer determines the type of operation (e.g., `get-config`, `edit-config`) and finally, configuration or notification data is included in the content layer.

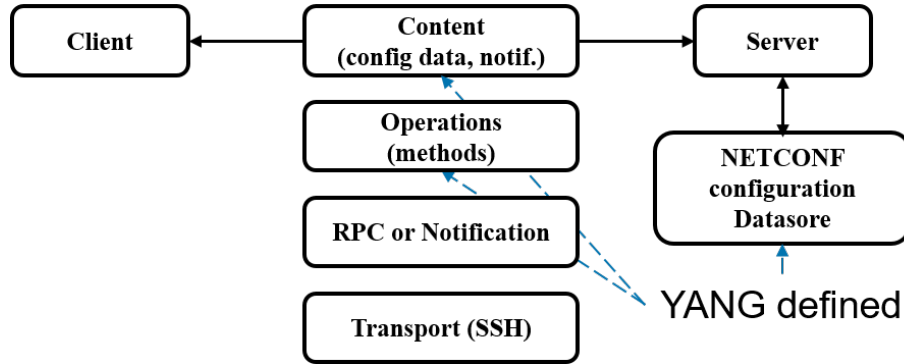


Fig. 1. Netconf protocol stack

One of the most early and significant contributions to the usage of NETCONF in optical network is [7]. This paper proposes a YANG model to describe a sliceable transponder to be deployed in an elastic optical network with variable rate, code, modulation formats, and monitoring capabilities. Two use cases are introduced by this paper: transponder configuration and notification upon BER threshold exceed. It is significant to observe the messaging of NETCONF, which includes edit-config, subscription, and notification messages.

The authors have also presented several papers with special focus on using YANG/NETCONF to enable the SDN control of Sliceable multidimensional (Spectral and Spatial) Transceivers (S-SST) [8] [9]. The presented data model encompassed a list of slices. Each slice is composed of a slice ID, a list of optical-channels and a list of optical-signal parameters (associated to the optical-channels). The paper showed the used sequence of NETCONF messages employed for provisioning one slice and requesting the BER.

3.2 RESTconf

In order to offer an ease of use protocol, based on HTTP, in order to properly modify and configure the YANG datastores. Representational State Transfer (REST) is an architectural style that defines a set of constraints to be used for creating Web services. RESTful Web services allow their clients to access and manipulate textual (JSON/XML) representations of Web resources by using a uniform and predefined set of stateless operations.

With this in mind, RESTCONF was defined as a REST-like protocol that provides a HTTP-based API to access the data, modeled by YANG [10]. The REST-like operations are used to access the hierarchical data within a datastore. The information modeled in YANG is structured in the following tree:

- /restconf/data : “Data (configuration/operational) accessible from the client”
- /restconf/modules : “Set of YANG models supported by the RESTCONF server”
- /restconf/operations : “Set of operations (YANG-defined RPCs) supported by the server”
- /restconf/streams: “Set of notifications supported by the server”

One of the first models of network services using YANG and connection establishment using RESTconf protocol was presented in as an experimental demonstration of distributed cloud and heterogeneous network orchestration with a common Transport API for E2E services with QoS [11]. The proposed Transport API would evolve in future ONF Transport API, which is later described in next section.

Other uses of RESTconf protocol have been demonstrated, with special focus on experimental validation of a converged metro architecture for transparent mobile front-/back-haul traffic delivery using SDN-enabled sliceable bitrate variable transceivers [12].

Moreover, a programmable SDN-enabled S-BVT based on hybrid electro-optical MCM [13] has been presented. On it, the programmable parameters of the SDN-enabled S-BVT were identified enabling the control plane to configure the transceiver according to the network requirements. The reconfiguration time of the S-BVT was up to 90 seconds, due to the use of offline DSP.

3.3 gRPC

gRPC Remote Procedure Calls [14] is a cloud native high performance, general purpose, RPC framework open sourced by Google. It uses HTTP/2 as a transport protocol and uses protocol buffers encodings for transported messages.

This protocol has an open source tool chain that allows the generation of efficient client libraries. As it is based on HTTP/2 transport protocol and uses byte-oriented encoding, it introduces low latency. It is highly scalable and it is a good fit for distributed systems.

It has been presented to be used for high performance in optical telemetry streaming [15]. It was demonstrated that streaming telemetry capabilities for

optical networks could be implemented as modular software service, allowing very high frequencies of all critical optical performance metrics.

4 Standards of network programmability and automation

In the previous sections, data modelling languages and protocols have been presented. During last years, the usage of them has been a significant revolution in the control and management of optical networks and devices. In order to provide adoption, operators and vendors have joint standard defining organizations or other kinds of multi-lateral agreements in order to solidify specific data models to provide inter-vendor compatibility.

ONF Transport API and IETF TEAS models have been proposed as valid control and management solutions for optical networks. Besides, OpenROADM and Openconfig are focused to the control and management of network elements, with special focus on disaggregated optical networks.

4.1 ONF Transport API

In Transport Networks, where network-control function and behavior are well-understood and established, standardizing application programmer's interfaces (APIs) to the network control functions becomes important. ONF has proposed the set of functional requirements and information model for the Transport API (T-API) [16].

The T-API abstracts the main services for an SDN controller: (1) Network Topology, (2) Connectivity Requests, (3) Path Computation, (4) Network Virtualization, and (5) Notification.

Network Topology functionality requires, at a minimum, that the interface exports the context, which is the scope of control, interaction and naming that a particular T-API provider (SDN controller) or client application has with respect to the information exchanged over the interface. The context describes the Service Interface Points, which refer to the customer-facing aspects of the Node Edge Points functions, and the Network Topology. The SDN controller provides information about the links and nodes in the domain (complete or abstracted, depending on the configured shared context). It is clear that the more information is shared, the less abstracted the network appears.

The Connectivity Service represents an "intent-like" request for connectivity between two or more Service Interface Points. Instead, a connection is the provisioned potential for forwarding (circuit/packet) between two or more Node Edge Points of a Node. However, there are other Connectivity Constraints that can be requested, such as: excluding or including nodes/links for traffic engineering, defining the protection level, defining its bandwidth or defining its disjointness from another connection.

ONF Transport API was first demonstrated in [17], as a solution for SDN in carriers networks. A first implementation of topology and connectivity services

was provided, alongside a discuss of the clear benefits of providing an standard interface for multi-vendor inter-operability.

The authors have proposed several extensions to ONF T-API to support multiple novel technologies, such as Space Division Multiplexing. [18] demonstrated an experimental NFV MANO Framework for the Control of SDM/WDM-Enabled Fronthaul and Packet-Based Transport Networks by Extending the TAPI.

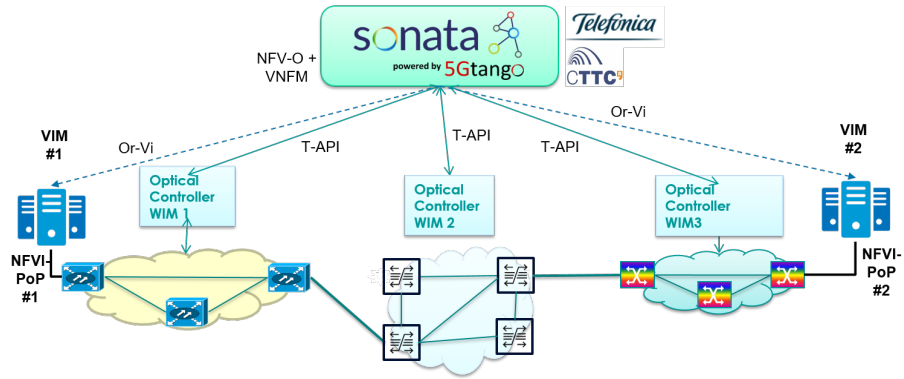


Fig. 2. Interconnection of Multiple NFV Infrastructure Points of Presence using ONF T-API

Figure 2 shows the interoperability demonstration and the usage of Transport API extensions for the Interconnection of Multiple NFV Infrastructure Points of Presence [19].

Finally, Transport API has also been demonstrated as a valid solution of hierarchical control and management of optical networks [20].

4.2 IETF ACTN

IETF has proposed the Abstraction and Control of TE Networks (ACTN) as a framework with multiple data models that has been demonstrated as a feasible and scalable solution for multi-domain, multi-technology transport network scenarios to provide E2E network services. In the proposed architecture, the Multi-Domain Service Coordinator (MDSC) offers a Transport NBI in order to retrieve network topology [21] and establish tunnel connectivity [22].

The authors extended this framework in order to allow the control and management of Flexible Ethernet (FlexE) over OTN networks using RESTconf protocol [23]. In this work, extensions were proposed to enable that specific FlexE capabilities are supported within the YANG data models currently being defined within the context of IETF TEAS working group: topology5 and tunnel6, which define a Transport NBI.

The demonstration in [24] presented Cross Stratum Orchestration (CSO) as a feasible solution for NFV points of presence interconnection. The demo architecture and the interfaces/APIs were aligned with IETF ACTN.

4.3 OpenROADM

The Open ROADM Multi-Source Agreement (MSA) defines interoperability vendor-neutral specifications for Reconfigurable Optical Add/Drop Multiplexers (ROADM) [25]. Included are the ROADM switch as well as transponders and pluggable optics. Specifications consist of both Optical interoperability as well as YANG data models. First version focuses on Metro fixed-grid Networks. Second version includes Flex-grid and Long distance networks. Current third version includes support of Layer 0 (ROADM components) and Layer 1 (OTN: Lambda, ODU).

One of the first OpenROADM demonstrations was presented by [26]. This demo shows how a hierarchical control plane of ONOS SDN controllers orchestrates the dynamic provisioning of end-to-end Carrier Ethernet circuits on a composite network. This demo integrates the control of a disaggregated optical metro network based on NETCONF and on YANG models defined by the OpenROADM project. For this, ONOS has been featured with new device drivers and with a suitable northbound application.

4.4 Openconfig and gNMI

OpenConfig is led by network operators who share the goal of more dynamic, programmable infrastructure by adopting software-defined networking principles [27].

In the next sections, the proposed data models are discussed, as well as the gRPC Network Management Interface(gNMI).

Data models OpenConfig focuses on compiling a consistent set of vendor-neutral YANG data models. These data models cover a variety of network elements, from routers to optical switches. The set of models provided for optical-transport provide a configuration and state model for terminal optical devices within a DWDM system, including both client- and line-side parameters.

gNMI gNMI is part of Openconfig and it is a standardized protocol defined in protocol buffers to deal with YANG-based data models and to interact using a specific Network Management Interface [28]. The main operations are detailed:

- SetRequest: is sent from a client to the server to update values in the data tree.
- SetResponse: is the response to a SetRequest, sent from the server to the client.

- **GetRequest**: is sent when a client initiates a Get RPC. It is used to specify the set of data elements for which the server should return a snapshot of data.
- **GetResponse**: is used by the server to respond to a GetRequest from a client.
- **Subscription**: Subscription is a single request to subscribe to a certain path that is specified and interpreted as the elements of the data tree that the client is subscribing to. The mode determines how the server should trigger updates to be sent.

Although the specification claims that the data can be either encoded in JSON or in Protobuf, open source implementations currently only use JSON, which does not benefit from a byte-oriented protocol.

5 Evaluation

This section provides a basic summary of the presented data models, protocols and standards that have been detailed during this invited paper. The objective of this section is to provide a quick guide in order to select the most suitable solution for the inclusion of control and management mechanism in optical use cases and scenarios. Each of the proposed solutions has been demonstrated its validity and necessity, but it is true that each solution provides some specificities that make it more adequate for certain scenarios and use cases.

Detailed examples for each of the proposed solutions can be found in the repository [29].

5.1 Data modelling languages

In Table 1, YANG and Protocol Buffers are compared. The key fundamental differences are the protocols that can be combined, its usage focus and the encoding types.

Table 1. Data modelling languages

Data Models	YANG	Protocol Buffers
Protocol	NETCONF, RESTconf, gNMI	gRPC
Focus	Model of config and state data	Structured serialization
Encoding	JSON, XML	Byte-oriented

5.2 Protocols

NETCONF, RESTconf and gRPC are compared in Table 2. These protocols can be compared using the transport layer behind, the different data encoding formats, their own auto-discovery mechanisms, their efficiency and protocol delay.

If we consider transport mechanisms, NETCONF uses SSH, while RESTconf introduces HTTP (adding slight protocol delay and reducing efficiency), and gRPC introduces HTTP/2 to overcome the HTTP limitations in terms of efficiency and protocol delay. Auto-discovery is a significant feature in NETCONF and RESTconf that is not introduced in gRPC, being necessary the previous knowledge of common protocol buffers between the client and the server.

Table 2. protocols

Protocols	NETCONF	RESTconf	gRPC
Transport	SSH	HTTP	HTTP/2
Encoding	XML	XML/JSON	byte
Autodiscovery	yes	yes	no
Efficiency	low	low	high
Protocol delay	slow	slow	fast

5.3 Standards

Table 3 summarizes the presented standards for control and management of optical networks. They can be compared depending on the data modelling language of use (basically YANG), the complexity of data model (based on the length of the proposed data trees), and the standardization defining organization that has validated the data models.

Table 3. standards

Standards	T-API	IETF TEAS	OpenROADM	OpenConfig	gNMI
Focus	Transport SDN Controller NBI	Transport SDN Controller NBI	Disaggregated ROADM	Router and line card config.	Operations and notifications of network elements
Data Model	YANG	YANG	YANG	YANG	Protobuf
Complexity	low	high	high	high	low
SDO	ONF, OIF	IETF	MSA	-	-

6 Conclusions

This paper has justified the need for data models and protocols in order to control and monitor optical networks and elements. The authors have reviewed the current state of the art with significant contributions to each of the proposed solutions. Finally, a comparison of the proposed solutions is provided in order to better adapt them to the necessary optical use cases and scenarios.

Acknowledgments

Research partly funded by EC H2020 5GPPP 5GTANGO (H2020-ICT-2016-2 761493) and Spanish MINECO project AURORAS (RTI2018-099178-B-I00).

References

1. Casellas, R., Martínez, R., Vilalta, R., Muñoz, R.: Control, management, and orchestration of optical networks: Evolution, trends, and challenges. *Journal of Lightwave Technology* **36**(7), 1390–1402 (2018)
2. Bjorklund, M.: Yang-a data modeling language for the network configuration protocol (netconf). Tech. rep. (2010)
3. Pyang. <https://github.com/mbj4668/pyang/>, accessed: 2019-02-14
4. Pyangbind. <https://github.com/robshakir/pyangbind>, accessed: 2019-02-14
5. Buffers, P.: Google’s data interchange format (2011)
6. Enns, R., Bjorklund, M., Schoenwaelder, J., Bierman, A.: Network configuration protocol (netconf). Tech. rep. (2011)
7. Dallaglio, M., Sambo, N., Akhtar, J., Cugini, F., Castoldi, P.: Yang model and netconf protocol for control and management of elastic optical networks. In: *Optical Fiber Communications Conference and Exhibition (OFC)*, 2016. pp. 1–3. IEEE (2016)
8. Muñoz, R., Yoshikane, N., Vilalta, R., Fàbrega, J., Rodríguez, L., Casellas, R., Moreolo, M.S., Martínez, R., Nadal, L., Soma, D., et al.: Sdn control of sliceable multidimensional (spectral and spatial) transceivers with yang/netconf. *Journal of Optical Communications and Networking* **11**(2), A123–A133 (2019)
9. Muñoz, R., Yoshikane, N., Fàbrega, J., Vilalta, R., Rodríguez, L., Moreolo, M.S., Casellas, R., Martínez, R., Beppu, S., Soma, D., et al.: Sdn control and monitoring system for soft-failure detection and optical restoration of spectral/spatial superchannels. In: *2018 European Conference on Optical Communication (ECOC)*. pp. 1–3. IEEE (2018)
10. Bierman, A., Bjorklund, M., Watsen, K.: Restconf protocol. Tech. rep. (2017)
11. de Lerma, A.M.L., Vilalta, R., Muñoz, R., Casellas, R., Martínez, R., Moreolo, M.S., Fàbrega, J.M., Yan, S.Y., Aguado, A., Salas, E.H., et al.: First experimental demonstration of distributed cloud and heterogeneous network orchestration with a common transport api for e2e services with qos. In: *Optical Fiber Communication Conference*. pp. Th1A–2. Optical Society of America (2016)
12. Fabrega, J.M., Moreolo, M.S., Reixats, L.N., Vílchez, F.J., Casellas, R., Vilalta, R., Martínez, R., Muñoz, R., Fernández-Palacios, J.P., Contreras, L.M.: Experimental validation of a converged metro architecture for transparent mobile front-/back-haul traffic delivery using sdn-enabled sliceable bitrate variable transceivers. *Journal of Lightwave Technology* **36**(7), 1429–1434 (2018)
13. Nadal, L., Moreolo, M.S., Fàbrega, J., Casellas, R., Vílchez, F., Martínez, R., Vilalta, R., Muñoz, R.: Programmable sdn-enabled s-bvt based on hybrid electro-optical mcm. *IEEE/OSA Journal of Optical Communications and Networking* **10**(6), 593–602 (2018)
14. grpc. <https://grpc.io/>, accessed: 2019-02-14
15. Sadasivarao, A., Jain, S., Syed, S., Pithewan, K., Kantak, P., Lu, B., Paraschis, L.: High performance streaming telemetry in optical transport networks. In: *Optical Fiber Communication Conference*. pp. Tu3D–3. Optical Society of America (2018)

16. Sethuraman, K., Chen, Q., Segev, E., Varma, E., Zhang, G., Ding, H., Busi, I., He, J., Ong, L., Davis, N., et al.: Functional requirements for transport api. v. 0.16, ONF (2016)
17. Lopez, V., Vilalta, R., Uceda, V., Mayoral, A., Casellas, R., Martínez, R., Muñoz, R., Palacios, J.P.F.: Transport api: A solution for sdn in carriers networks. In: ECOC 2016; 42nd European Conference on Optical Communication; Proceedings of. pp. 1–3. VDE (2016)
18. Vilalta, R., Muñoz, R., Landi, G., Rodriguez, L., Capitani, M., Casellas, R., Martínez, R.: Experimental demonstration of the bluespace’s nfv mano framework for the control of sdm/wdm-enabled fronthaul and packet-based transport networks by extending the tapi. In: 2018 European Conference on Optical Communication (ECOC). pp. 1–3. IEEE (2018)
19. Vilalta, R., de Lerma, A.M.L., López, V., et al.: Transport api extensions for the interconnection of multiple nfv infrastructure points of presence. In: Optical Fiber Communication Conference. Optical Society of America (2019)
20. Mayoral, A., Vilalta, R., Muñoz, R., Casellas, R., Martínez, R., López, V.: Cascading of tenant sdn and cloud controllers for 5g network slicing using transport api and openstack api. In: Optical Fiber Communications Conference and Exhibition (OFC), 2017. pp. 1–3. IEEE (2017)
21. Liu, X., Bryskin, I., Beeram, V., Saad, T., Shah, H., De Dios, O.G.: Yang data model for te topologies. draft-ietf-teas-yang-te-topo, work in progress (2016)
22. Xhang, X., et al.: A yang data model for traffic engineering tunnels and interfaces. draft-ietf-teas-yangte, work in progress
23. Vilalta, R., Martínez, R., Casellas, R., Muñoz, R., Lee, Y., Fei, L., Tang, P., López, V.: Network slicing using dynamic flex ethernet over transport networks. In: Optical Communication (ECOC), 2017 European Conference on. pp. 1–3. IEEE (2017)
24. Vilalta, R., López, V., Lee, Y., Zheng, H., Lin, Y., Casellas, R., González-de Dios, O., Martínez, R., Muñoz, R.: Towards ip & transport network transformation using standardized transport northbound interfaces. In: 2018 Optical Fiber Communications Conference and Exposition (OFC). pp. 1–3. IEEE (2018)
25. MSA, O.R.: Open roadm overview (2016)
26. Morro, R., Lucrezia, F., Gomes, P., Casellas, R., Giorgetti, A., Velasco, L., Riccardi, E., Piat, A.C., Percelsi, A., Pedro, J., et al.: Automated end to end carrier ethernet provisioning over a disaggregated wdm metro network with a hierarchical sdn control and monitoring platform. In: 2018 European Conference on Optical Communication (ECOC). pp. 1–3. IEEE (2018)
27. Shaikh, A., Hofmeister, T., Dangui, V., Vusirikala, V.: Vendor-neutral network representations for transport sdn. In: Optical Fiber Communication Conference. pp. Th4G–3. Optical Society of America (2016)
28. Shakir, R., et al.: grpc network management interface (gnmi). draft-openconfig-rtgwg-gnmi-spec-01, work in progress
29. Ofc2019 short course 472 - hands-on: Controlling and monitoring optical network equipment with netconf/yang, accessed: 2019-02-14