YANG Model and NETCONF Protocol for Control and Management of Elastic Optical Networks

M. Dallaglio[1], **N. Sambo**[1], **J. Akhtar**[1], **F. Cugini** [2], **P. Castoldi**[1]

1: Scuola Superiore Sant'Anna, Pisa, Italy; 2: CNIT, Pisa, Italy.

matteo.dallaglio@sssup.it

Abstract: This paper proposes a YANG model to describe a sliceable transponder with variable rate, code, modulation formats, and monitoring capabilities. The model is introduced into NETCONF, which is experimentally demonstrated in a testbed.

OCIS codes: 060.4250, 060.1660.

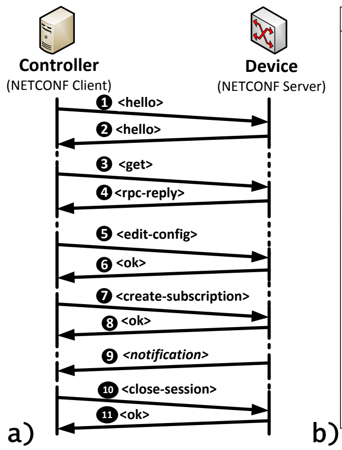
1. Introduction

In the recent years, the continuous increase of traffic has driven relevant advances in the data and control planes for optical networks. Emerging data plane transponder technologies are going to support high bit rates such as 1 Tb/s, possibly looking at downscaling power consumption and costs per bit, and offering high spectral efficiency, thus increasing network life [1, 2]. Such transponders, thanks to coherent detection, also enable monitoring of transmission parameters (e.g., pre forward error correction bit error rate —pre-FEC BER) [3]. Then, Software Define Networking (SDN) is emerging as a control plane able to remotely set network devices, thus programming transmission characteristics (such as bit rate) and switching [4, 5]. However, while data and control planes have experienced such advances, the innovations in the management plane have not followed these trends yet [6]. A key point in the management is related to the presence of network devices from different vendors. The lack of standard solutions (e.g., for operation administration and maintenance — OAM [7, 8]) complicates the management of a network. NETCONF [9] is emerging as SDN protocol providing both control (e.g., data plane device configuration) and management (e.g., access to monitoring information) functionalities. NETCONF (which is based on XML language) may exploit YANG data model to describe network elements to be controlled/managed. YANG model [10, 11] and NETCONF are of interest for operators since they provide a standard way to control and manage network elements, independently from the vendor.

This paper presents and demonstrates a YANG model describing transponders supporting slice-ability, variable rate, and monitoring functionalities. The control plane experimental demonstration is carried out as follow: first, NETCONF message exchange is shown for the configuration of transmission characteristics for transmitter and receiver; then, a signal degradation is emulated to trigger the OAM system and the alarms (or notifications).

2. NETCONF protocol and Transponder YANG

Main NETCONF messages are described in Fig. 1a with reference to sliceable transponder [1] control and management. First, the central controller (NETCONF client) sends a `Hello` message to the transponder device (NETCONF server) to create a session (1). The device replies with a `Hello` message (2) with a reference (through a namespace) to the YANG model, which lists all the parameters describing a sliceable transponder. In this phase, the controller and the device agree on the parameters to be set. Part of the proposed YANG model will be detailed later in Fig. 1b. Once the model is agreed, the controller has to be informed about the transponder capabilities (e.g., supported bit rate values). Thus, the controller sends a `Get` message to the device (3), which replies with a `Rpc-reply` message containing information about its supported transmission parameters such as bit rate (e.g., 100 and 200 Gb/s) or modulation format (e.g., polarization multiplexing quadrature phase shift keying —PM-QPSK) (4). Then, the controller, based on connection request, sets the parameters of the transponder by sending an `Edit-conf` message (5). If no error occurs, the device replies with an `Ok` message (6). NETCONF also provides a way to generate alarms in case a monitored parameter falls in a critical range. In this case, the device sends a `Notification` message to the controller when a particular event occurs (e.g., pre-FEC BER over a given threshold). Notifications supported by the device must be defined in the YANG model. The controller needs to subscribe to the device notifications in order to receive them (7-8) and may decide to subscribe to all the device notifications or only to specific notification streams. A filter can be applied to the subscription in order to receive only those notifications satisfying certain criteria (e.g., threshold exceeding). When an event satisfying the filtering criteria occurs, the device sends a `Notification` message to the controller (9). Finally, when the connection is released, the controller closes the session (10-11).



b)

Top level transponder YANG	Sub-carrier module state	Supported parameters example
container transponder {	grouping subcarrier-module-state {	<supported-bit-rates>
list subcarrier-module {	container supported-bit-rates{	<bit-rate>112</bit-rate>
key "subcarrier-id";	leaf-list bit-rate {	<bit-rate>124</bit-rate>
leaf subcarrier-id {	type bit-rate-type;	<bit-rate>224</bit-rate>
type uint32;	}	<bit-rate>248</bit-rate>
}	}	</supported-bit-rates>
uses subcarrier-module;	container supported-baud-rates {	<supported-baud-rates>
}	leaf-list baud-rate {	<baud-rate>28</baud-rate>
......	type baud-rate-type;	<baud-rate>31</baud-rate>
container connections {	}	</supported-baud-rates>
uses connections;	}	<supported-modulations>
}	container supported- modulations {	<modulation>mf:bpsk</modulation>
}	leaf-list modulation {	<modulation>mf:qpsk</modulation>
	type modulation-type;	</supported-modulations>
	}	<supported-fec>
	}	<fec>fec:ldpc</fec>
	uses fec-state;}	</supported-fec>

Fig. 1. (a) NETCONF message exchange; (b) YANG model.

A portion of the proposed YANG model describing the transponder is shown in Fig. 1b. The sliceable transponder –supporting several sub-carriers– has been modelled as a list of sub-carrier modules and a list of connections (Fig. 1b, *Top level transponder YANG*). Sub-carrier modules and connections have configuration and state data (Fig. 1b, *Sub-carrier module state*). The configuration data are set by the controller through `Edit-conf` messages. The state data are updated by the device and monitored by the controller through polling or notifications. In the model, a sub-carrier module can be used either in transmission or in reception. The configuration data of the sub-carrier module are the following: bit rate, baud rate, modulation, FEC, central frequency, and bandwidth. Some additional fields may appear depending whether the sub-carrier is used in reception (e.g., local oscillator, sampling rate, and analog bandwidth [1]) or in transmission (e.g., the launch power). The state data of the sub-carrier module are the following (Fig. 1b, *Supported parameters example*): supported bit rates, baud rates, modulation formats, and FEC. Then, an additional field is proposed for monitoring capabilities, i.e. listing the parameters that can be monitored by the transponder itself: e.g., input power at the receiver, pre-FEC BER, polarization mode dispersion, chromatic dispersion, and Q-factor. Regarding the connections, the configuration data are the following: a list of sub-carriers module ids referring to those sub-carriers involved in the connection, and the frequency slot composed by the nominal central frequency (n) and the slot width (m), according to ITU-T [5].

3. Experimental demonstration

The controller runs a python implementation of a NETCONF client. The device runs ConfD, a NETCONF server implementation made by Tail-f (CISCO), and a C program we made to emulate the monitoring capabilities of the device. Our program communicate with ConfD through specific APIs provided by the latter. The transponder YANG model, as in the previous section, has been included in the capabilities of both client and server. In the experiment, a 100 Gb/s connection request has been considered. We derived supported transmission parameters from [12]: e.g., a baud rate of 28 Gbaud PM-QPSK supports 100 Gb/s net rate and 7% FEC, while 31 Gbaud around 20% FEC. Fig. 2a shows the capture of the NETCONF message exchange. The `Edit-config` message is generated containing the information to configure the involved sub-carrier and the parameters of the connection. The sub-carrier is configured by setting the values of all the fields, such as bit-rate (112 Gb/s), modulation format (PM-QPSK), FEC (7%), baud rate (28 Gbaud) and so on (Fig. 2b). The connection is then inserted in the connections list, specifying the id of the involved sub-carrier (id=2 in the experiment), the nominal central frequency (193.1 THz, i.e. n=0) and the frequency slot width (37.5 GHz, i.e. m=3). Moreover, the controller subscribes to the transponder `Notification` stream specifying, through a filter, that it is interested on pre-FEC BER exceeding 9×10^{-4} threshold (Fig. 2c). We emulated pre-FEC BER threshold exceed so that the device sent a `Notification` message to inform the controller (Fig. 2d).

4. Conclusions

This paper proposed and demonstrated a NETCONF protocol including YANG model describing a transponder with monitoring capabilities. Two use cases have been considered: transponder configuration and notification upon BER threshold exceed. NETCONF message capture is provided for `Edit-config`, `Subscription`, and `Notification` messages.

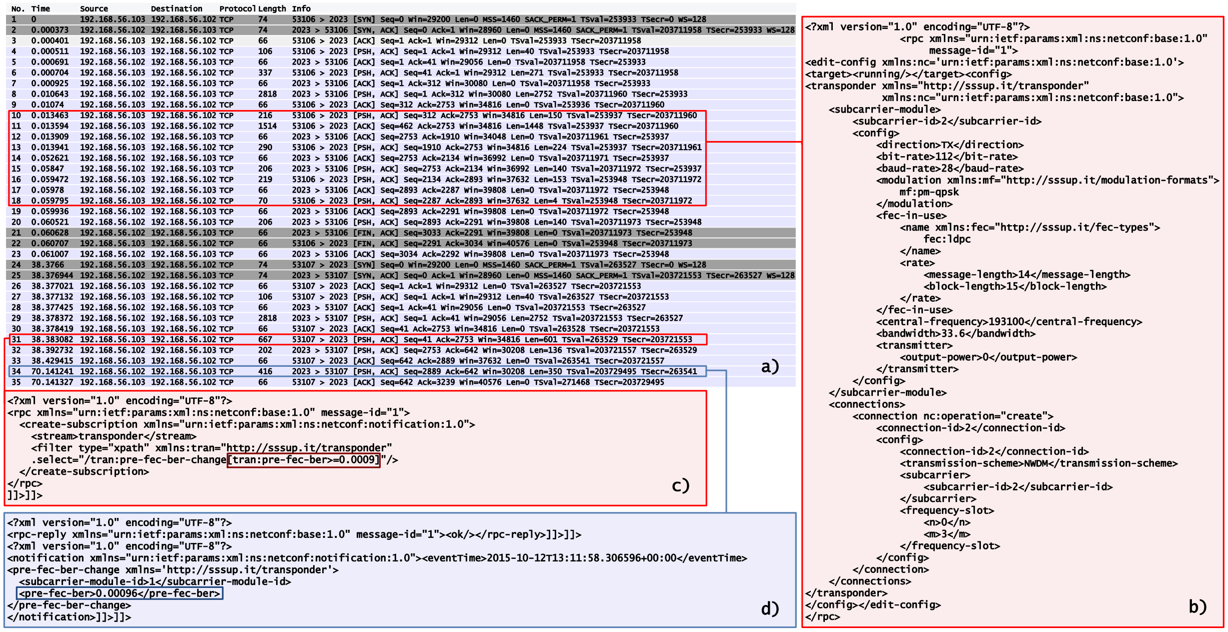


Fig. 2. Capture of NETCONF messages.

Acknowledgment: This work was supported by the EC through the Horizon 2020 ORCHESTRA project (grant agreement 645360).

References

1. N. Sambo and et al., “Next generation sliceable bandwidth variable transponders,” *Communications Magazine, IEEE*, vol. 53, no. 2, pp. 163–171, Feb 2015.
2. J. Cai, H. Batshon, M. Mazurczyk, H. Zhang, Y. Sun, O. Sinkin, D. Foursa, and A. Pilipetskii, “64QAM based coded modulation transmission over transoceanic distance with ¿ 60 Tb/s capacity,” in *Proc. of OFC*, 2015.
3. A. Napoli and et al., “Next generation elastic optical networks: The vision of the european research project IDEALIST,” *Communications Magazine, IEEE*, vol. 53, no. 2, pp. 152–162, Feb 2015.
4. L. Liu, R. Munoz, R. Casellas, T. Tsuritani, R. Martinez, and I. Morita, “Openslice: An OpenFlow-based control plane for spectrum sliced elastic optical path networks,” in *Proc. of ECOC*, Sept 2012, pp. 1–3.
5. N. Sambo, G. Meloni, F. Paolucci, F. Cugini, M. Secondini, F. Fresi, L. Poti, and P. Castoldi, “Programmable transponder, code and differentiated filter configuration in elastic optical networks,” *JLT*, 2014.
6. A. Martinez, M. Yannuzzi, V. Lopez, D. Lopez, W. Ramirez, R. Serral-Gracia, X. Masip-Bruin, M. Maciejewski, and J. Altmann, “Network management challenges and trends in multi-layer and multi-vendor settings for carrier-grade networks,” *Communications Surveys Tutorials, IEEE*, vol. 16, no. 4, 2014.
7. D. King and A. Farrel, “A PCE-based architecture for application-based network operations,” IETF RFC 7491.
8. N. Sambo, F. Cugini, A. Sgambelluri, and P. Castoldi, “Hierarchical monitoring architecture and OAM handler,” in *Proc. of ECOC*, 2015.
9. R. Enns, M. Bjorklund, J. Schoenwaelder, and A. Bierman, “Network configuration protocol (NETCONF),” IETF RFC 6241, June 2011.
10. M. Bjorklund, “YANG - a data modeling language for the network configuration protocol (NETCONF),” IETF RFC 6020.
11. J. Vergara and et al., IETF draft-vergara-flexigrid-yang-00, Oct. 2014.
12. in *http://www.cisco.com/c/en/us/products/collateral/optical-networking/ons-15454-series-multiservice-provisioning-platforms/data_sheet_c78-713298.pdf.*