

Automated End to End Carrier Ethernet Provisioning over a Disaggregated WDM Metro Network with a Hierarchical SDN Control and Monitoring Platform

Original

Automated End to End Carrier Ethernet Provisioning over a Disaggregated WDM Metro Network with a Hierarchical SDN Control and Monitoring Platform / Morro, R., Lucrezia, F., Casellas, R., Giorgetti, A., Velasco, L., Riccardi, E., Chiado Piat, A., Percelsi, A., Pedro, J., Gifre, Ll., Sgambelluri, A., Risso, F., Marchetto, G.. - ELETTRONICO. - (2018). (44th European Conference on Optical Communication (ECOC 2018) Roma 23 - 27 settembre 2018).

Availability:

This version is available at: 11583/2712316 since: 2019-02-19T10:42:41Z

Publisher:

IEEE

Published

DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

IEEE postprint/Author's Accepted Manuscript

©2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

Automated End to End Carrier Ethernet Provisioning over a Disaggregated WDM Metro Network with a Hierarchical SDN Control and Monitoring Platform

R. Morro⁽¹⁾, F. Lucrezia⁽²⁾, P. Gomes⁽³⁾, R. Casellas⁽⁴⁾, A. Giorgetti⁽⁵⁾, L. Velasco⁽⁶⁾, E. Riccardi⁽¹⁾, A. Chiadò Piat⁽¹⁾, A. Percelsi⁽¹⁾, J. Pedro⁽³⁾, Ll. Gifre⁽⁷⁾, A. Sgambelluri⁽⁵⁾, F. Rizzo⁽²⁾, G. Marchetto⁽²⁾

⁽¹⁾ TIM, Torino, Italy, roberto.morro@telecomitalia.it; ⁽²⁾ Politecnico di Torino, Torino, Italy; ⁽³⁾ Coriant, Lisbon, Portugal; ⁽⁴⁾ CTTC/CERCA, Castelldefels, Spain; ⁽⁵⁾ CNIT, Pisa, Italy; ⁽⁶⁾ Universitat Politècnica de Catalunya, Barcelona, Spain; ⁽⁷⁾ Universidad Autónoma de Madrid, Madrid, Spain.

Abstract 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, programming the whole data path from the CPE to the core optical equipment.

Introduction

End-to-end carrier Ethernet services are a compelling use-case in the context of SDN/NFV because they require deep interaction between resources located in the Central Office (CO) and the transport network providing interconnection between the COs. CORD¹ is the ONF initiative aiming at bringing the agility, flexibility and automation of a cloud infrastructure into each Telco's CO. Enterprise CORD (E-CORD) is the CORD use-case offering enterprise connectivity services over metro networks, namely layer 2 Virtual Private Networks (L2-VPN) on Carrier Ethernet² circuits. This demo is based on the E-CORD platform^{3,4} enhanced to control an optical transport network composed by disaggregated elements and extended with a Monitoring and Data Analytics (MDA) platform⁵ able to collect performance data from the devices that can be analysed using bespoke Machine Learning (ML) algorithms to proactively detect anomalies in the network.

Demo overview

The demo objective is to show the dynamic setup of a Carrier Ethernet-Virtual-Circuit (EVC) from a global vantage point over a composite access and transport network, whose configuration is fully automated by the control-plane. The system operates on the FutureNet⁶

test-bed available in the TIM laboratories in Torino, Italy, composed of five CORD CO nodes interconnected by a disaggregated optical network. The demo will be carried out accessing remotely the FutureNet test-bed and showing: i) the global view of the logical topology made up of virtual devices representing the underlying physical domains (i.e. CORD sites and transport network), ii) the automated provisioning of EVCs that span multiple CORD sites, and iii) the collection of monitoring data and the application of data analytics procedures to detect degradations affecting the EVCs. The process of setting up a circuit involves the installation of forwarding and traffic control rules on multiple devices composing the data-path, from the terminal Customer Premises Equipment (CPE) placed at the customer building to the disaggregated ROADM switches of the metro transport network, passing through the operator COs located at the edge of the WAN (Fig. 1). The network control-plane is entirely based on ONOS; the ONOS instance controlling the optical domain has been featured with NETCONF device drivers implementing OpenROADM YANG device models⁷.

Data plane

CPE: the CPE is the Microsemi Edge Assured ea1000 that, despite the size and the energy

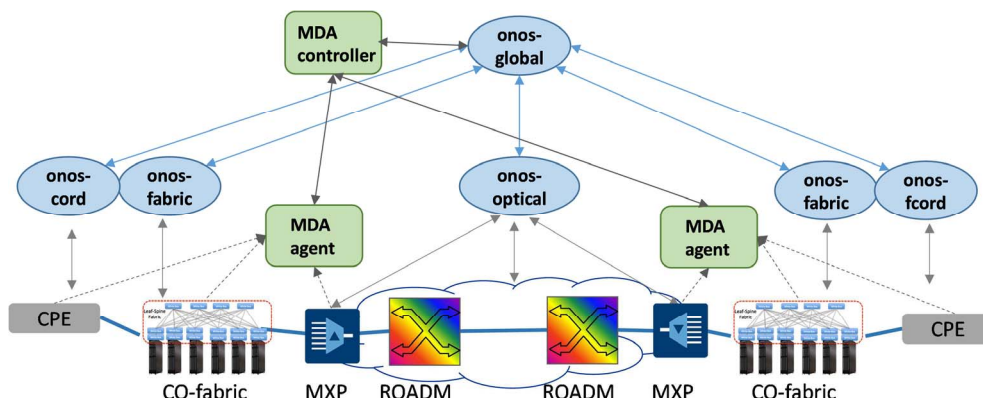


Fig. 1: Network controllers view and monitoring architecture

consumption of a normal SFP, contains an embedded Linux machine providing a NETCONF/YANG management interface. This device represents the demarcation point between the customer and the operator domain being reachable by an *in-band* NETCONF session from the operator network. It basically performs two functions: i) classify customer traffic by mapping a set of Customer VLANs (C-tags, IEEE802.1q) to a Service tag (S-tag, IEEE802.1ad QinQ), ii) check the end-to-end connectivity status through *active monitoring* of the physical layer (Connection-Fault-Management, IEEE802.1ag). In addition, an API is available for the MDA system to collect different measures such as delay, delay jitter and packet loss.

L2 network: The second device in the data path is the Openflow-based EdgeCore 5712 switch that is part of the CO fabric. This switch is programmed to cross-connect the CPE and the transport network, forwarding packets based on the S-tag.

Transport Network: the disaggregated optical domain consists of two muxponders (MXPs), the Coriant Groove G30⁸, and two low cost ROADMs prototypes developed by TIM. Muxponders and ROADMs are located at the edges of a regional field link between Torino and Chivasso made of three 76 km-long G.652 fiber spans featuring commercial amplifiers without open control interfaces. The muxponders are equipped with one sled fitting one CFP2-ACO pluggable interface which features tunable optics and supports DP-QPSK (100G), 8QAM (150G) and 16QAM (200G) modulation formats (data rates) on the line side and one QSPF+ client interface. Muxponders are modeled using YANG and equipped with a built-in NETCONF server. The ROADM prototypes are based on commercial Wavelength Selective Switch (WSS) modules to realize 3-degrees and one add-drop directionless devices using a switch-and-select architecture. The WSSs are fitted on an evaluation board that allows configuration by means of an RS232 interface. The ROADM prototypes are modeled using YANG OpenROADM device model version 2.2. A NETCONF server, exploiting the developed OpenROADM model and running on a separate computer, translates NETCONF edit-config operations into the appropriate set of WSS commands.

Control plane

The platform is composed by a two-layer hierarchy of controllers⁹ (architecture in Fig.1, service setup workflow in Fig.2). A parent

controller (named *onos-global*) coordinates three domains, two CORD COs where the subscribers are connected and one optical transport metro network interconnecting the different CORD COs. A total of five ONOS instances (plus the *onos-global* one) is involved in the process since each CORD CO uses two different instances, one for the virtual tenant and the access networks (i.e. the CPE), *onos-cord*, and the other for CO fabric control, *onos-fabric*.

Global controller: The global controller maintains an abstract topological view of the network in terms of virtual devices and links between them. The virtual devices represent the underlying domains modeled as *big-switches* exposing only the demarcation points representing specific network-to-network and user-to-network interfaces (NNIs, UNIs). From a service portal GUI the administrator can request an EVC by selecting specific UNIs (i.e. CPEs) in a geographical map, setting the list of involved customer VLANs and an associated bandwidth profile. Upon a request, the GUI contacts the global ONOS controller, step (1) in Fig.2, which contains the Carrier Ethernet application able to process the request triggering a series of APIs all the way down to the physical devices. First, an admission control routine is applied, and then, if accepted, a service tag is generated and the request is split into as many forwarding constructs as the number of virtual devices along the path between the UNIs. The forwarding constructs are then sent to the local controllers using REST-based APIs, steps (2)-(6) in Fig. 2, which are responsible to allocate the appropriate network resources.

CO controllers: the ONOS applications running in each CORD CO coincide with those of the standard E-CORD release; they take care of exposing the logical abstraction of the underlying topology to the global controller. Then the domain-specific controllers are ready to receive requests for EVC setup from the global controller. Specifically, *onos-cord* uses NETCONF protocol (i.e., edit-config operations) to configure the Microsemi ea1000, steps (2a) and (6a) in Fig. 2; whereas *onos-fabric* uses OpenFlow protocol (i.e., Flow-mod messages) to configure the leaf-spine switches, steps (3a) and (5a) in Fig. 2.

Transport controller: similar functionalities have been added to the *onos-optical* which is featured with a NorthBound (NB) app that translates the request of the *global-onos* into an optical intent request. The optical intent resolves into a Routing and Spectrum Assignment (RSA) process within the ONOS core to allocate a lighthouse (or reuse a previously allocated one)

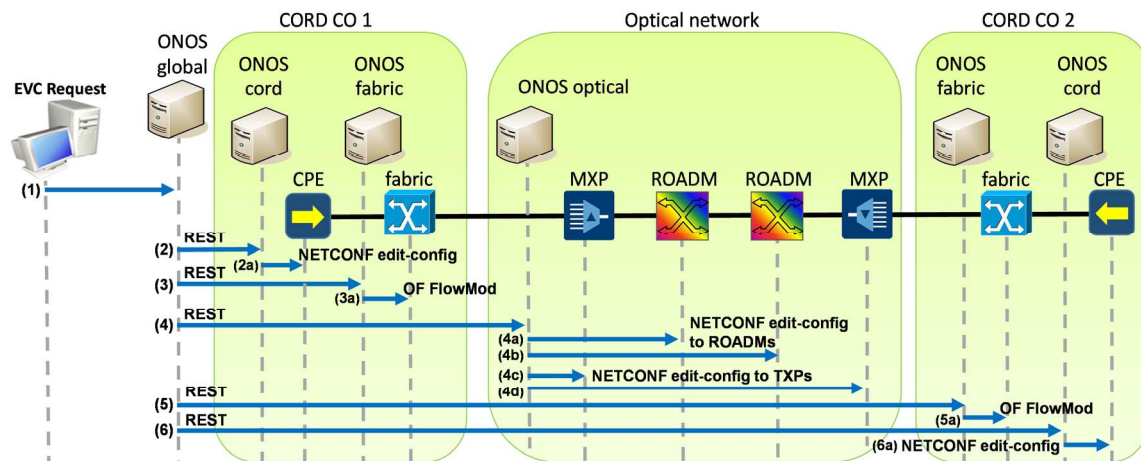


Fig. 2: Service set-up workflow

for the EVC request. The intent is then compiled in a set of FlowRules that are eventually converted in NETCONF operations (i.e., edit-config) by the ONOS driver of each optical device. Specifically, for the demonstrated setup ONOS drivers have been developed for the Coriant Groove G30 muxponder and for the TIM ROADM, implementing a set of behaviors to support setup and removal of light-paths (e.g., FlowRuleProgrammable, LambdaQuery).

Monitoring and Data Analytics (MDA)

The MDA framework is distributed; every CO includes an MDA agent that collates monitoring data from the network devices. The MDA agent includes a local module containing data analytics applications for handling and processing data records. In addition, a centralized MDA controller that contains a big data repository and data analytics capabilities is used to collect monitoring data records from the MDA agents in the COs. Data records and notifications are stored and processed in correlation with operational databases.

In the demo, the MDA agents collect monitoring data from CPEs, which aggregates them and sends to the MDA controller to be stored and visualized using a Web GUI. Measurements are analysed by ML algorithms in the MDA agents; specifically, delay measurements is used as a metric that needs to be closely followed to detect degradations that might impact the service. When a degradation is detected a notification is sent towards the MDA controller and could be used e.g., for triggering an in-depth analysis to localize the root cause of the failure.

Innovation, relevance and conclusions

The demo shows how network resources on multiple technology domains can be allocated in a unified way by means of easy-to-consume APIs, thanks to the separation of concerns provided by the two layers hierarchy of network controllers. While the E-CORD scenario has

been already demonstrated in public events, 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 NB application. To the best of the authors knowledge, this is the first attempt to show how multiple ONOS clusters can be coordinated to fully automate the provisioning of network resources from the access to the transport network. Moreover, a MDA system, able to collect performance monitoring data can be easily integrated in the platform opening the doors to applications for automatic proactive control over wide area networks.

Acknowledgements

This work was supported in part by the European Commission funded H2020-ICT-2016-2 METRO-HAUL project (G.A. 761727).

References

- [1] CORD website: <https://opencord.org>, 2018.
- [2] MEF, <https://www.mef.net/resources/technical-specifications>, 2018.
- [3] E-CORD guide: <https://guide.opencord.org/profiles/ecord>, 2018.
- [4] Codecase of E-CORD application: <https://gerrit.opencord.org/admin/projects/carrierethernet>, 2018.
- [5] Ll. Gifre et al., "Autonomic Disaggregated Multilayer Networking," JOCN, vol. 10, pp. 482-492, 2018.
- [6] "TIM FutureNet: a CORD based network demonstrator", CORD Build 2017, San Jose (CA), Nov. 7-9 2017, https://schr.ws/hosted_files/cordbuild2017/71/20171109-TIM_Futurenet-final.pdf.
- [7] OpenROADM website: <http://www.openroadm.org>, 2018.
- [8] Coriant Groove G30 datasheet: http://cdn.extranet.coriant.com/resources/Data-Sheets/DS_Groove_G30_NDP_Muxponder_74C0157.pdf, 2018.
- [9] F. Lucrezia, "Network Infrastructure for Highly Distributed Cloud-Computing", Doctoral Thesis, Politecnico di Torino, 2018.