



Bravalheri, A., Muqaddas, A., Uniyal, N., Casellas, R., Nejabati, R., & Simeonidou, D. (2019). VNF Chaining across Multi-PoPs in OSM using Transport API. In *2019 Optical Fiber Communications Conference and Exhibition, OFC 2019 - Proceedings: Oral Sessioning Information [W1G.4]* Optical Society of America (OSA).  
<https://doi.org/10.1364/OFC.2019.W1G.4>

Peer reviewed version

Link to published version (if available):  
[10.1364/OFC.2019.W1G.4](https://doi.org/10.1364/OFC.2019.W1G.4)

[Link to publication record in Explore Bristol Research](#)  
PDF-document

This is the author accepted manuscript (AAM). The final published version (version of record) is available online via OSA at <https://www.osapublishing.org/abstract.cfm?URI=OFC-2019-W1G.4> . Please refer to any applicable terms of use of the publisher.

## University of Bristol - Explore Bristol Research

### General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:  
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

# VNF Chaining across Multi-PoPs in OSM using Transport API

Anderson Bravalheri<sup>1</sup>, Abubakar Siddique Muqaddas<sup>1</sup>, Navdeep Uniyal<sup>1</sup>,  
Ramon Casellas<sup>2</sup>, Reza Nejabati<sup>1</sup> and Dimitra Simeonidou<sup>1</sup>

<sup>1</sup>High Performance Networks Group, Department of Electrical and Electronic Engineering, University of Bristol

<sup>2</sup>CTTC/CERCA, Av. Carl Friedrich Gauss, 7, Castelldefels (Barcelona), Spain

<sup>1</sup>{a.bravalheri, abubakar.muqaddas, navdeep.uniyal, reza.nejabati, dimitra.simeonidou}@bristol.ac.uk,

<sup>2</sup>ramon.casellas@cttc.es

**Abstract:** Management and Network Orchestration (MANO) systems permit simultaneous orchestration of compute and network resources. Here, we experimentally demonstrate the integration of Transport API based WIM with Open Source MANO (OSM), for NFV orchestration over optical networks.

**OCIS codes:** (060.4250) Networks; (060.4253) Networks, circuit-switched

## 1. Introduction

There has been recent shift in telecommunications industry towards technologies like Network Function Virtualization (NFV) and Software Defined Networking (SDN). The hardware network equipment is being replaced by Virtual Network Functions (VNFs), which can run in remote datacenters as well as the edge computing nodes. The constraints in the placement of VNFs within NFV [1] is resulting in an increase of inter-datacenter traffic, which is forecasted to reach upto 14% of the total data-center traffic by 2021 [2]. The bandwidth and latency sensitive use-cases like Augmented/Virtual Reality, 8K 360° video transmission etc. where some processing is done in remote datacenters and other on the edge nodes will be some of the contributors to this increase in traffic. The distribution of VNFs across remote and edge-computing nodes for realizing these use-cases has made it necessary to create an end-to-end network and resource control. It has also created the need to build the underlying networks to satisfy the stringent requirements of low latency and high bandwidth across the cloud and edge computing infrastructure. In order to achieve these targets, projects like Metro-Haul [3] are creating an ecosystem to provide high bandwidth, low latency, end-to-end connected and programmable networks. Such stringent network requirements can only be sustained using optical networks in the long run [4]. Furthermore, Metro-Haul utilizes ETSI NFV MANO systems to orchestrate network services over optical metro networks, to create end-to-end network services.

ETSI NFV MANO standards include the Wide-Area Networking (WAN) Infrastructure Manager (WIM) which allows to orchestrate network services over multiple data-centers in a WAN. However, due to the heterogeneous nature of the underlying networks aligned with the complexity of end-to-end orchestration, its functionality and APIs have not been clearly defined yet [5]. In this case, a Transport API (T-API) [6] based WIM can be used to integrate optical networks within an NFV MANO system, for network services with VNFs having high-bandwidth and low latency requirements. T-API has been standardized within the ONF by the Open Transport Configuration & Control (OTCC) group and is being increasingly adopted by the industry. Thus, using T-API for integration of optical networks with the existing NFV-MANO systems allows easier upgrade and operation of existing MANO systems. Furthermore, T-API simplifies the interdomain networking, encompassing both electrical domain and optical domains and abstracts the complex underlying topology.

In this work, we propose and implement an architecture to integrate T-API with the leading ETSI NFV MANO system i.e. Open Source MANO (OSM) [8] for the first time, which allows orchestration of VNFs across data-centers over the underlying optical networks. The whole system including the data plane is demonstrated where the applicability of T-API as interface towards the WIM is established, using SDN controllers for multi-domain network orchestration.

## 2. T-API as WIM Northbound API

The choice of T-API as the interface between the OSM and the WIM is motivated by multiple factors. T-API, as defined by the ONF, has fulfilled its goal of having a common standardized North Bound Interface (NBI) across multiple network controllers, to be used e.g., by a network orchestrator or applications. It builds on core models that are technology agnostic (aligned with the ONF Core Information Model, which defines a common object model for all types of Software Defined Networks, including components like network resources or service constructs) while allowing technology-specific extensions for relevant layers. T-API follows a model driven development: models are defined in UML and automatically translated into YANG which, in turn, may be automatically validated and stubs generated for

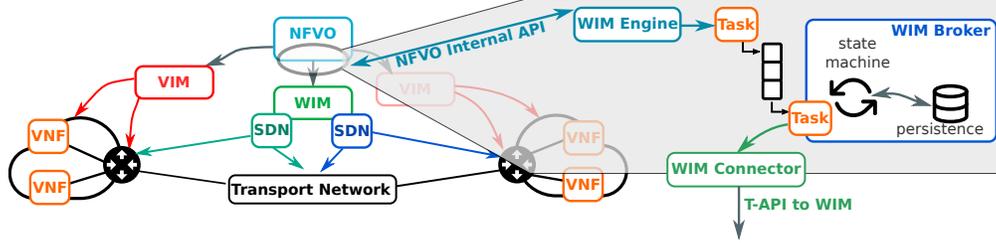


Fig. 1: Proposed WIM Architecture in the context of ETSI NFV MANO framework [11]

common programming languages. From the point of view of maturity and adoption, it has been demonstrated in multiple proof-of-concepts, interop events and is supported by many vendors [9]. It has been adopted by several initiatives and SDOs, such as MEF OpenCS Optical Transport or, for the latest v2.1 release, the ODTN Project [10] as the NBI for a controller of an optical disaggregated transport network. Release 2.0 added the ability to take into account node constraints, protection services and consistent OAM and monitoring, and the latest 2.1 release includes new models for the photonic layer, with support for flexi-grid network media channels, including model for the OCH, OTSi, OTSiA, OTSiG, OMS, OTS and Media channels as per ITU-T G.872 (2017) version 4.

### 3. Architecture for WIM Integration in NFV MANO

As shown in Fig 1, our proposed architecture conforms to the ETSI NFV MANO standards [11] focusing on the orchestration on VNFs across multiple data-centers (known as Points of Presence (PoPs)). Given a request to create a Network Service (NS) composed VNFs in a multi-PoP environment, the NFV Orchestrator (NFVO) first determines where to place each individual VNF considering a set of restrictions and optimization criteria expressed during the request. With this information, the NFVO requests each underlying Virtual Infrastructure Manager (VIM) to instantiate the corresponding VNFs and gets information about the related local virtual networks where they would be attached. In turn, the VIMs can use a local SDN controller to chain the traffic through the local VNFs to an SDN-enabled switch that serves as an endpoint for the interconnection. Meanwhile, the NFVO assigns to one or more WIMs the responsibility to create WAN links by configuring the underlying transport network via T-API.

In order to realize the aforementioned steps, we propose the introduction of sub-components inside the NFVO, as indicated in Fig. 1. This architecture is based in the execution of asynchronous tasks to avoid blocking behaviour and improve the overall network service provisioning time. The **WIM Engine** has three responsibilities: firstly, to provide information to the NFVO about the available WIMs and the characteristics of the connectivity they are able to sustain; secondly, to check the feasibility of the presented placement solution; and finally, to specify and schedule a series of tasks that encode the instructions for establishing WAN links. While each task works as a standalone processing unit, implementing a state machine upon activation, the **WIM Broker** coordinates their threaded execution in a task queue. By tapping into a shared communication mechanism internal to the NFVO (implemented via either a common database or event stream), tasks can verify preconditions and fire actions to the external WIMs accordingly. As an example, special information, such as VNF IP and VLAN segmentation id, might need to be published by a VIM to the NFVO after its workload is processed. To wait for this information, the tasks can be suspended, and rescheduling them is the responsibility of the WIM Broker. Conversely, when all the preconditions are met, a task invokes commands on a **WIM Connector**. To ensure fault tolerance, tasks are made idempotent and synchronized with a persistent data store before any transition. Although the WIM Connector is designed to abstract the WIM API permitting protocol-agnostic systems, in this work we propose the usage of the T-API due to its support to a wide variety of transport technologies and SDN topologies.

### 4. Implementation and Experimental Demonstration

We have developed a preliminary implementation of the proposed architecture using OSM as baseline system to demonstrate and evaluate the use of T-API with MANO for NFV orchestration over a Multi-PoP infrastructure. This implementation introduces a new WIM subsystem in OSM Resource Orchestrator (RO), containing the sub-components described in Section 3. For demonstrating its applicability, a network service composed by 3 VNFs (Fig. 2a) was deployed over an optical infrastructure. Fig. 2b shows the experimental setup, which consists of two OpenStack datacenters interconnected by a combination of packet switches and optical cross connects (OXC) which are SDN-enabled and managed by a combination of SDN controllers. On top of the SDN controller, we employ a T-API proxy based on a T-API 2.0 reference implementation [12]. The T-API proxy exposes two service endpoints (*svc\_ep\_1* and *svc\_ep\_2*), at both edges of the interconnection network between the datacenters, to OSM. The information about these service endpoints is added to OSM before network service deployment as part of a port mapping

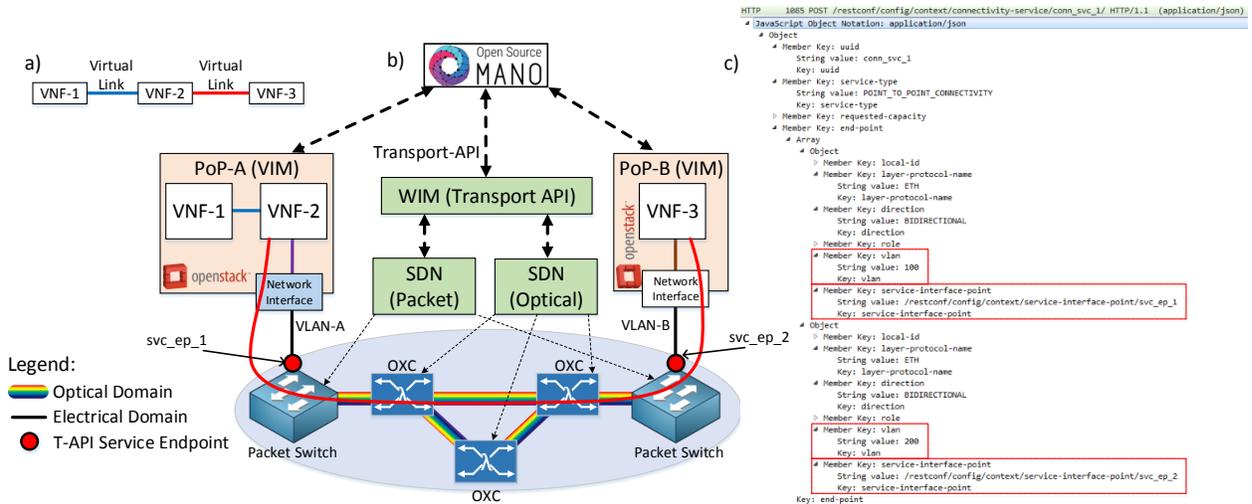


Fig. 2: a) Network Service, b) Testbed for T-API with OSM, c) Wireshark Capture along with Create Connectivity JSON

process, making them available for usage.

At the time of deployment of the network service by OSM, the location of VNFs is specified, such that the network service spans across multiple datacenters as in Fig. 2b. This is followed by OSM deploying the VNFs at the specified datacenters through the VIMs. In addition to the VNFs, the VIMs also create the network connected to VNF-2 and VNF-3, which spans outside the datacenter to the transport network with VLAN A and VLAN B respectively. Once the VNFs and the networks are created, the VIM sends the VLAN information to the OSM. OSM detects that the virtual link connecting VNFs 2 and 3 spans multiple datacenters. Once OSM receives the VLAN information, it uses the pre-defined port mapping to obtain the T-API service endpoints connected to each datacenter. The T-API service endpoint (*svc\_ep*) and the VLAN for each datacenter is sent to the T-API proxy to provision a network between the datacenters. The T-API proxy translates this information into low-level device specific flow installation using the SDN controller to interconnect the two VLANs at the either datacenter, as shown by the red line in Fig. 2b. Fig. 2c shows the Wireshark capture of the T-API *Create Connectivity Service* message that OSM sends to the T-API proxy during network deployment. It includes the two T-API service endpoints and the VLANs used at either datacenter, which is used to provision the end-to-end network between the datacenters.

## 5. Conclusion

In this work, we presented the integration of standard T-API with OSM, a popular ETSI based open source MANO system, focusing on the orchestration of VNFs over multiple PoPs connected over optical networks. The proposed architecture was validated and demonstrated using a multi-datacenter and multi-domain network experiment.

Our presented implementation has been submitted as an open source contribution planned for the next ETSI OSM Release. In future, the upcoming T-API 2.1 photonic extensions will be exploited to improve optical network orchestration capabilities.

## Acknowledgments

This work is funded by EPSRC grant EP/L020009/1: TOUCAN and EU H2020 Metro-Haul (grant agreement 761727).

## References

1. J. G. Herrera et. al., "Resource Allocation in NFV: A Comprehensive Survey," IEEE TNSM, 2016
2. Cisco, "Global Cloud Index: Forecast and Methodology, 2016-2021 White Paper," 2018
3. Metro-Haul, <https://metro-haul.eu>
4. D. King et. al., "The role of SDN and NFV for flexible optical networks: Current status, challenges and opportunities," ICTON 2015.
5. R. Casellas et al., "Control, Management, and Orchestration of Optical Networks: Evolution, Trends, and Challenges," IEEE JLT 2018.
6. V. Lopez et al., "Transport API: A Solution for SDN in Carriers Networks," ECOC 2016
7. ETSI NFV ISG PoC Report [https://nfvwiki.etsi.org/images/PoC\\_42\\_Final\\_Report\\_Mapping\\_ETSI-NFV\\_onto\\_Multi-Vendor\\_....pdf](https://nfvwiki.etsi.org/images/PoC_42_Final_Report_Mapping_ETSI-NFV_onto_Multi-Vendor_....pdf)
8. Open Source MANO, <https://osm.etsi.org>
9. OIF-ONF Whitepaper, "SDN Transport API Interoperability Demonstration," 2017
10. ONF *Open Disaggregated Transport Networks ODTN project*, <https://wiki.onosproject.org/display/ODTN/ODTN>.
11. ETSI, "NFV; Ecosystem; Report on SDN Usage in NFV Architectural Framework," ETSI GS NFV-EVE 005 (V1.1.1) 2015.
12. ONF, "TAPI Reference Implementation," <https://github.com/OpenNetworkingFoundation/TAPI>