

# Hierarchical and recursive NFV service platform for end-to-end network service orchestration across multiple NFVI domains

Raul Muñoz<sup>1</sup>, Ricard Vilalta<sup>1</sup>, Ramon Casellas<sup>1</sup>, Ricardo Martínez<sup>1</sup>, Felipe Vicens<sup>2</sup>, Josep Martrat<sup>2</sup>,  
Víctor López<sup>3</sup>, Diego López<sup>3</sup>

<sup>1</sup>*Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Castelldefels (Barcelona), Spain.*

<sup>2</sup>*ATOS, Madrid and Barcelona, Spain*

<sup>3</sup>*Telefónica I+D, Madrid, Spain.*

*raul.munoz@cttc.es, felipe.vicens@atos.net, victor.lopezalvarez@telefonica.com*

## ABSTRACT

The ETSI network function virtualisation (NFV) management and orchestration (MANO) architectural framework defines an NFV service platform composed of an NFV orchestrator and virtualised network function (VNFs) managers that are responsible to provide NFV network services. Network services can be defined as the joint instantiation of VNFs and the provisioning of required network connections between the different VNFs to jointly realise a more complex function (e.g. service function chaining). In general, it is assumed that a single service platform is governing the whole NFV infrastructure (NFVI) domain from the edge to the core of the network, composed of multiple edge computing and core DCs interconnected by heterogeneous optical access, metro and core transport networks. However, network operators will fragment their transport networks and DCs into multiple NFVI domains for scalability, administrative, or security issues. Each NFVI domain can be provided by different vendors relying on their own NFV service platforms or leveraging open source solutions such as SONATA, OSM, or ONAP. In this paper, we propose a hierarchical and recursive network service orchestration architecture to compose end-to-end network services by aggregating network services provided by per-domain NFV service platforms together with VNFs.

**Keywords:** NFV MANO, hierarchy, recursiveness, transport SDN, NFVI, datacentre

## 1. INTRODUCTION

The broad adoption of Network Function Virtualization (NFV) in the fifth generation of mobile technology (5G) requires distributed computing resources seamlessly integrated with the optical access and transport networks from the edge to the core of the network as shown in Fig. 1. In general, an NFV infrastructure (NFVI) is composed of multiple NFVI points of presence (NFVI-PoPs) at the edge and core of the network. An NFVI-PoP is a set of computing, storage and network resources to deploy virtualised network functions (VNFs) through the virtualisation layer (e.g. virtual machine –VM- or container-CT- based technologies). It is deployed in micro-DCs in the edge nodes (e.g. cell sites, street cabinets, lamppost), small-DCs (e.g., in the central offices –COs-) for low/moderate-computation capacity and low response time, and core-DCs in the core network for high-computational capacity and moderate response time. On the network side, the NFVI-PoP are interconnected with heterogeneous optical access and transport networks that go beyond a commodity providing fixed bandwidth pipes for bulk data. Latest advances in programmable and elastic optical systems and subsystems enable to provide dynamic and adaptive connectivity between distributed NFVI-PoPs.

The adoption of a reference architecture such as the ETSI NFV Management and Orchestration (MANO) framework [1] can provide an efficient network service (NS) management and resource orchestration from an end-to-end perspective. The ETSI NFV MANO architectural framework identifies three functional blocks; the Virtualized Infrastructure Manager (VIM), the VNF Managers (VNFM), and the NFV Orchestrator (NFVO). In general, the VIM is responsible for managing the NFVI compute, storage and network resources. However, the ETSI NFV MANO framework has also defined the WAN infrastructure Manager (WIM), as a particular VIM. In this scenario, the VIM is responsible for controlling and managing the NFVI-PoP's resources, whilst the WIM is used to establish connectivity between NFVI-PoP's. The VIM is commonly implemented using a cloud controller (e.g., OpenStack), and the WIM can be performed by a dedicated Transport SDN controller (e.g. OpenDaylight, ONOS, Ryu) in charge of managing the optical and packet network resources. On the other hand, the VNFM is responsible for the lifecycle management (i.e., instantiation, scaling, updating and termination) of VNF instances running on top of VMs/CTs managed by the VIM. The NFVO has two main responsibilities; the orchestration of NFV infrastructure resources across multiple VIMs and WIMs (resource orchestration), and the lifecycle management of NFV network services (NS orchestration). The NS orchestration is responsible for managing and interconnecting groups of VNF instances that jointly realise a more complex function (e.g. service function chaining) by interfacing with the VNFM and VIMs/WIMs. The NFVO and VNFM are typically implemented together in open source service platforms such as Open Network Automation Platform NFV (ONAP), Open Source MANO (OSM), and SONATA.

A single and monolithic NFV service platform covering the network operator's NFVI from end-to-end is not a feasible solution. Network operators will fragment their transport networks and DCs into multiple NFVI

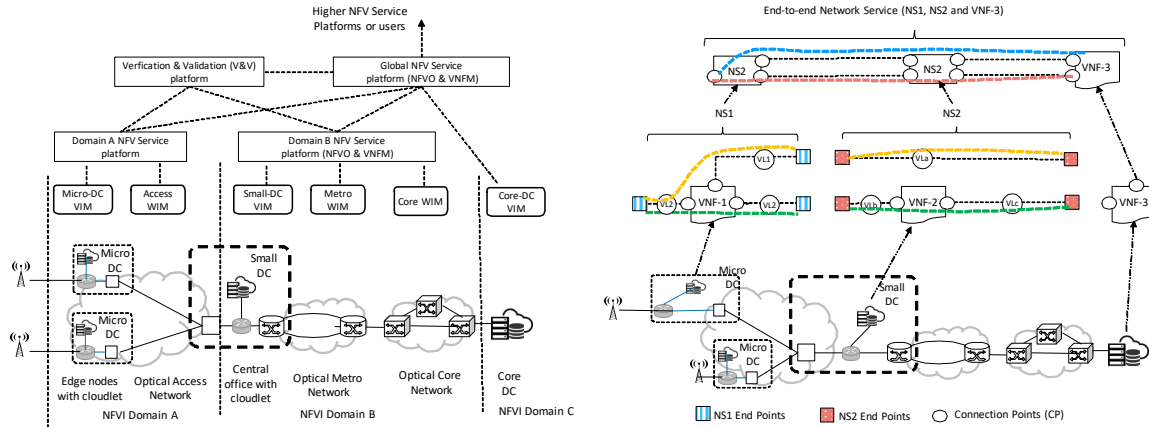


Figure 1. a) Proposed hierarchical and recursive NFV service platform architecture. b) Example of aggregated end-to-end NS composed of per-domain NS

domains to cope with administrative and regional organisations. Each NFVI domain will be provided by different vendors relying on proprietary NFV service platforms or leveraging open source solutions. In this paper, we explore the hierarchy and recursiveness of the NFV service platform in order to enable end-to-end network services across multiple NFVI domains. We propose a hierarchical and recursive network service orchestration architecture to compose end-to-end network services by aggregating network services provided by per-domain NFV service platforms together with network functions (VNFs or PNFs). We also propose to make use of the verification and validation (V&V) platform developed in 5GTANGO project to properly test per-domain network services on each NFVI domain with different NFV service platform implementations.

## 2. HIERARCHICAL AND RECURSIVE NFV ORCHESTRATION ARCHITECTURE WITH PER-DOMAIN NETWORK SERVICE VERIFICATION AND VALIDATION

Cooperative NFV service platforms can be organised in hierarchical or peer architectures. In the hierarchical model, one NFV service platform acts as global orchestrator of multiple NFV service platforms with a parent/child hierarchy. The hierarchical architecture requires to use a common API between the parent NFV service platform (southbound interface) and the child platforms (northbound interface) to deliver the end-to-end network services. Moreover, if a parent NFV service platform also deploys the common API as northbound interface, then it can also become a child of another parent platform, enabling the recursiveness of the NFV service platforms. In the recursive hierarchical architecture, each higher level has the potential for greater abstraction and broader scope of the network services from the lower levels. In the peer model, the NFV service platforms are interconnected each other in an arbitrary mesh, which cooperates to provide end-to-end services using east/ west interfaces. The peer model is preferable in a multi-operator scenario where there is no hierarchy, no cross-domain control, and no cross-domain visibility. In this paper, we focus on the recursive hierarchical architecture for single-operator scenarios with multiple NFVI domains for scalability, or organisational purposes.

Fig. 1.a shows an example to illustrate the proposed hierarchical and recursive NFV service platform architecture. It is composed of three NFVI domains (Domain A, B and C). The NFVI domain A is covering the optical access segment (e.g. NG-PON, BVTs with SDM [2]) and the micro-DCs at the edge nodes. It deploys one VIM for the micro-DCs, an SDN controller acting as a WIM for the optical access network, and an NFV network service platform managing per-domain network services and VNFs. Domain B is pretty much similar to domain A, but targeting small-DCs at the central office, and the optical metro and core network segments (e.g., Flexi-grid DWDM). It deploys one VIM for the small-DCs, two WIMs, one for the metro network and another for the core network, and an NFV service platform. Finally, domain C is just composed by a core-DC and the corresponding VIM. Domain C does not deploy any NFV service orchestrator. It is worth to mention that currently the interface between the NFVO and the WIM is not widely implemented by most of the reference NFV service platform implementations and it still lacks maturity. 5GTANGO project is extending the SONATA platform with the Transport API (TAPI) as southbound interface for the WIM. Transport API is defined in [3] and experimentally validated by the authors in [4]. The Transport API enables to abstract a set of common SDN control plane functions (e.g., path computation, topology and connection provisioning) and define a common data model and protocol based on YANG/RESTconf. It allows to uniformly interact with heterogeneous SDN controllers, regardless of the specific open-source or proprietary SDN controller implementation. This abstraction enables to virtualise the network, that is, to partition the physical infrastructure into virtual networks that can be exposed to the NFV service platform.

Many different NFV service platforms implementations have emerged in the last few years using different network service programming models. A network service programming model defines the concepts and abstractions that developers build and use in order to define network services descriptors (NSD) and VNF descriptors (VNFD). For example, ETSI NFV ISG has proposed the TOSCA framework extended to support NFV services, the OpenStack HEAT has also been extended for such purpose, and many research projects and the OSM project have proposed proprietary network service descriptors based on YAML, JSON, XML, or even

dedicated programming languages. This implies that a conversion and transformation process is needed in order to cope with the current diversity in network service description formats. In the same way, the VNF packages (VNFD and VNF image) required to allow the exchange of such services between different entities and platforms use their package specifications making the onboarding process to different catalogues or platforms very complicated. To address this, ETSI recently started to define and specify a common VNF package format, based on the TOSCA CSAR standard [5].

Thus, the verification and validation of developed the per-domain network services and VNFs in a multi-NFVI domain environment with different NFV service platforms implementations are extremely challenging due to the large diversity in descriptor formats and packages. 5GTANGO project is designing a Verification and Validation (V&V) platform [6]. The V&V provides a verification and validation service that tests submitted VNFs or network services to ensure they pass a range of tests. The V&V can run the network service or VNF in a per-domain NFV service platform, stressing the network service and VNF and collecting the results. The V&V can test network services or VNFs which have been prepared for any NFV service platform implementation (both open source or proprietary). Thus, this ensures that when a developer, network operator, or other user obtains a VNF or network service from the V&V they can be sure that its V&V status is valid and up to date, and that the network service or VNF has passed the V&V process.

On top of the three NFVI domains that are composed by two Domain Service Platform and one NFVI, we deploy a top NFV service orchestrator that manages end-to-end network services composed of VNFs and the per-domain network services. It enables faster and easier reuse of existing network services by just reusing and referencing the corresponding per-domain NSDs. 5GTANGO has already started to introduce the notion of recursive NSDs to describe recursive network services that reference other network services [7]. Similarly to VNFs, per-domain network services can also be represented by connection points that are mapped to the network service termination points. Then, virtual links can connect connection points of VNFs and network services. The VNF forwarding graph (VNFFG) of an end-to-end network service can be defined by using the constituting per-domain network services, VNFs, and virtual links to construct the VNF forwarding paths (VNFFP). Fig. 1.b shows an example of two per-domain network services and an end-to-end aggregated network service composed by the two per-domain network services and one VNF. More specifically, NFVI domain A has a network service (NS1) that is composed of VNF-1 and three virtual links (VL1, VL2 and VL3) connecting the three VNF-1's connection points with the NS1 endpoints. NS1 also defines a VNFFG composed of two NFP as shown in Fig.1.b. The second per-domain network service (NS2) is offered by NFVI domain B. It is composed of VNF-2 that is connected with two virtual links (VLb and VLc) to the two NS2 end-points. Additionally, there is a virtual link (VLa) crossing the whole domain connecting other two NS2 end-points. NS2 also has defined a VNFFG composed of two paths, one going through VNF-2 and another through the VLa that crosses the NFVI domain B. Finally, the top NFV service platform offers an aggregated end-to-end network service that is composed of NS1, NS2 and VNF-3. The connection points of the NSs and the VNF are connected with virtual links, and a VNFFG is defined with two VNFFPs as shown in Fig.1.b.

Much like the TAPI for the transport SDN controllers, the definition of a common API as northbound and southbound interface for the NFV service platform is key to enable hierarchy and recursiveness in the orchestration of end-to-end network services. ETSI NFV ISG is defining the interface and information model specification for the Os-Ma-Nfvo reference point used for exchanges between the NFVO and the OSS [8]. This interface may be used as the common API for recursive and hierarchical orchestration. The Os-Ma-nfvo reference point supports the following interfaces:

- Network Service Descriptor (NSD) Management. This interface allows the management of NSDs and associated PNFs, as well as the VL descriptors and VNFFG descriptors. Some operations already defined are the on-boarding, disabling, enabling, updating, querying, and deleting of NSDs and PNFs.
- Network Service (NS) Lifecycle Management. Some operations already defined are the instantiation, termination, query, scaling, updating of NSs. The following operations are defined; create and delete NS Identifier; instantiate, scale, update, query, terminate, and heal NS; get operation status.
- NS Performance Management. This interface allows the OSS/BSS to invoke NS lifecycle management operations towards the NFVO.
- NS Fault Management. It allows the NFVO to provide alarms related to the NSs visible to the consumer.
- VNF Package Management. It allows for the management of VNF Packages. The following operations are defined for this interface: create, upload, update, and query VNF Package, delete and fetch VNF package Info, fetch VNF Package artefacts, subscribe to new notifications, notify of on-boarding of new VNF Package or changes of VNF Packages, terminate Subscription, query Subscription Info.

### 3. LIFECYCLE MANAGEMENT OF PER-DOMAN AND AGGREGATED NETWORK SERVICES

First, the NS/Test developer develops the required per-domain network services (NSD, VNF Package) using the own SDKs offered by the per-domain NFV service platforms, as well as deploy the tests to verify and validate the developed per-domain network service. Then, the NS/Test developer uploads all the per-domain network services and the related test packages in the V&V and invokes the tests to schedule them, as shown in

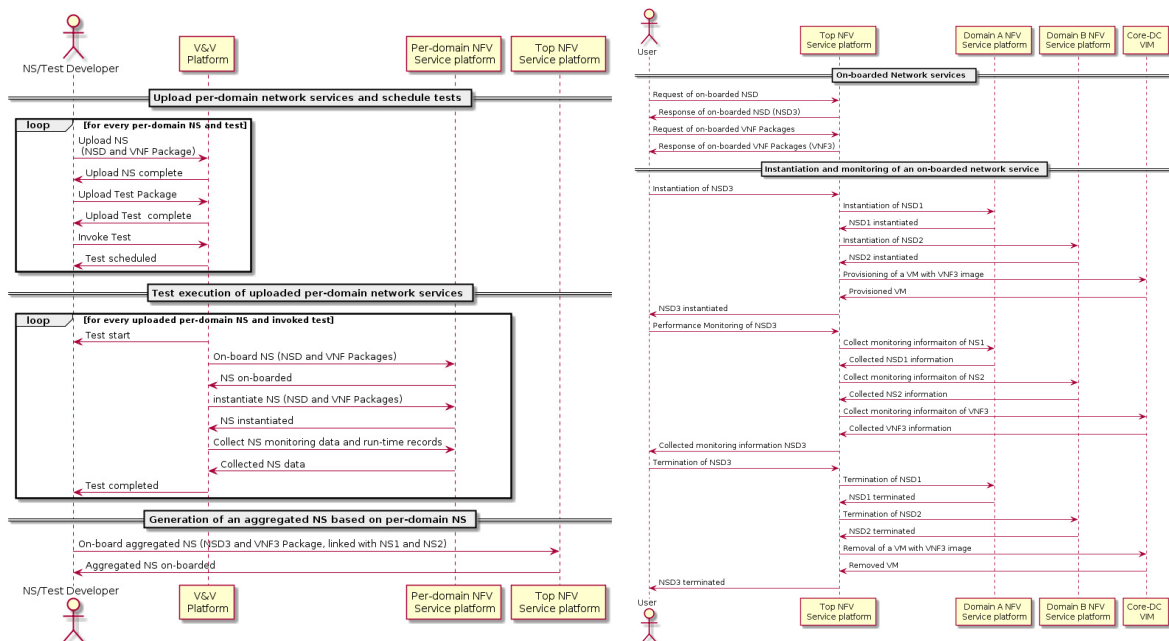


Figure 2. Workflow a) Per-domain NS test and end-to-end aggregated NS onboarding . b) Instantiation, monitoring and termination of the aggregated NS

Fig. 2.a. After some time, the V&V starts the tests and notifies the NS/Test developer. First, the V&V onboards and instantiate the network service, and finally collects the required network service monitoring data and run-time records. If the test results are good, the V&V sends a test completed message to the NS/Test developer. Otherwise, the V&V sends a test failed message, and the NS/Test developer is required to modify the network service and restart the whole process again. If the tests for all per-domain network services are successful, the NS/Test developer can develop a new aggregated network service by composing the developed network services together with VNFs. Then, the aggregated network service is onboarded in the top NFV service platform.

After that, the user (e.g., network operator or another NFV service platform) can request the onboarded network services to the top NFV service platform and select one, as shown in Fig. 2.b, and requests the instantiation. If the network service is composed of per-domain network services as the example in Fig. 1.b, the top NFV platform service requests to the per-domain NFV platforms the instantiation of the corresponding services (i.e., NS1 to domain A and NS2 to domain B), and then the provisioning of VNF-3 to the VIM of the core-DC. If everything is correct, the top NFV service platform sends a notification to the user. The same procedure can be applied for collecting performance monitoring information of the aggregated network service, as well as terminating the service.

#### 4. CONCLUSIONS

This paper has presented a hierarchical and recursive network service orchestration architecture to compose end-to-end network services by aggregating per-domain network services and VNFs. It is combined with the V&V platform developed in the 5GTANGO project to test the per-domain network services on each NFVI domain with vendor-specific or open-source NFV service platform implementations such as SONATA, OSM, or ONAP.

#### ACKNOWLEDGEMENTS

Work supported by EC H2020 5GTANGO (761493) and Spanish DESTELLO (TEC2015-69256-R) projects.

#### REFERENCES

- [1] ETSI GS NFV-MAN 001 (v.1.1.1) "Network Functions Virtualisation (NFV); Management and Orchestration, December 2014.
- [2] R. Muñoz et al., SDN control and monitoring of SDM/WDM and packet transport networks for 5G fronthaul/backhaul, IEEE Summer Topicals meetings, 9-11 July 2016, Waikoloa (USA).
- [3] ONF technical recommendation, Functional Requirements for Transport API, ONF TR-527, June 2016.
- [4] A. Mayoral, et al. "First experimental demonstration of distributed cloud and heterogeneous network orchestration with a common Transport API for E2E services with QoS." OFC, March 2016, USA.
- [5] ETSI GS NFV-SOL 004 (V2.3.1) "Network Functions Virtualisation (NFV) Release 2; Protocols and Data Models; VNF Package specification", July 2017.
- [6] 5GTANGO, Deliverable "D3.1 Verification and Validation strategy and Automated metadata management", October 2017. Available online at [https://5gtango.eu/documents/D31\\_v1.pdf](https://5gtango.eu/documents/D31_v1.pdf).
- [7] 5GTANGO, Deliverable "D4.1 First open-source release of the SDK toolset", April 2018, Available online at [https://5gtango.eu/documents/D41\\_v1.pdf](https://5gtango.eu/documents/D41_v1.pdf).
- [8] ETSI GS NFV-IFA 013 (v2.4.1) "Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification" Feb 2018.