TNOVA

NETWORK FUNCTIONS AS-A-SERVICE
OVER VIRTUALISED INFRASTRUCTURES

GRANT AGREEMENT NO. 619520

Deliverable D7.3

# Overall Assessment and Roadmap

**Editor** Jorge Carapinha (ALB)

**Contributors** José Bonnet, Bruno Parreira (ALB), Aurora Ramos (ATOS), Yacine Rebahi (FRAUNHOFER), Marco Di Girolamo, Samuele De Domenico, Giovanni Giuliani (HPE), Michael McGrath (INTEL), Giuseppe Monteleone (ITALTEL), David Dietrich (LUH), George Xilouris, Eleni Trouva (NCSRD), Alexander Phinikarides (PTL), Georgios Gardikis (SPH), Evangelos Markakis (TEIC), Nicolas Herbaut (VIO), Piyush Harsh (ZHAW)

**Version** 1.0

**Date** December 31st, 2016

**Distribution** PUBLIC (PU)

# Executive Summary

This report is one of the final public deliverables of the EU-FP7 Project T-NOVA, "Functions as-a-Service over Virtualised Infrastructures". It provides the results of Task 7.3, "Overall Assessment and Roadmap", which ran between months M28 and M36 as part of Work Package 7 (WP7) "Pilot Integration and Field Trials".

Since T-NOVA kicked off in January 2014, a considerable number of activities in related areas have been launched, including standardization bodies, open source projects, R&D projects and industry initiatives, which contributed to accelerate the uptake of NFV technologies and the evolution to new network paradigms, including 5G. This report aims to position T-NOVA in this highly dynamic landscape and to identify the specific contribution provided by T-NOVA in this context.

The flexibility of the T-NOVA architecture allows multiple variants or extensions of the basic model to be accommodated. Deployment guidelines are provided in this document to allow a rich variety of use cases, business models and NFV infrastructure deployment options. As is the case with most network technology evolutions, a key requirement to guarantee successful technology uptake is incremental deployment. Thus, a 3-stage deployment roadmap is proposed in this report.

Building on results from the technical WPs (3-6) and the evaluation/validation activities carried out in the scope of Task T7.2, this report provides an overall assessment of project accomplishments, putting them into perspective with ongoing activities and results from other projects and initiatives, especially taking into account the rapidly evolving technological environment in areas around virtualization of network functions.

Lessons learnt in the course of the project development are identified – this includes findings related to several stages of the project, ranging from architecture conception to practical VNFaaS implementation and deployment issues. Most of these lessons learnt are likely to provide useful guidance to other projects and activities exploring the same or similar topics.

Finally, a number of topics for future study are identified, based on the work performed and the results accomplished by T-NOVA, taking into account relevant technology evolution trends, including the exploitation of NFV in 5G systems.

# Table of Contents


**1. INTRODUCTION ....................................................................................6**

**2. TECHNOLOGY WATCH AND T-NOVA POSITIONING ..........................................8**

2.1. OPEN SOURCE NFV FRAMEWORKS .................................................... 8
*2.1.1. OPNFV.................................................... 8*
*2.1.2. OSM.................................................... 9*
*2.1.3. OpenBaton.................................................... 11*
*2.1.4. OpenStack Tacker.................................................... 14*
*2.1.5. OPEN-O.................................................... 16*
2.2. STANDARDS DEVELOPING ORGANIZATIONS.................................................... 17
*2.2.1. ETSI NFV.................................................... 17*
*2.2.2. TMForum.................................................... 27*
*2.2.3. MEF.................................................... 28*
2.3. R&D PROJECTS.................................................... 29
*2.3.1. 5G-PPP Projects.................................................... 29*
*2.3.2. Other European R&D Projects.................................................... 33*
*2.3.3. R&D projects outside EU.................................................... 35*
2.4. INDUSTRY INITIATIVES.................................................... 36
*2.4.1. AT&T ECOMP.................................................... 36*
*2.4.2. CORD.................................................... 37*

**3. DEPLOYMENT GUIDELINES ....................................................................38**

3.1. GUIDELINES FOR DEPLOYMENT OVER EXISTING NFV INFRASTRUCTURES.................................... 38
3.2. T-NOVA DEPLOYMENT USE CASES AND SCENARIOS FOR ADOPTION.................................... 43
*3.2.1. T-NOVA interfacing existing OSS/BSS.................................... 43*
*3.2.2. Multiple NFVI-PoP providers.................................... 43*
*3.2.3. Multiple T-NOVA SPs (Orchestrator Federation).................................... 44*
3.3. NFVI-POP LOCATION AND WAN CONNECTIVITY: EDGE VS. CORE.................................... 45
3.4. ROADMAP FOR INCREMENTAL DEPLOYMENT.................................... 46
*3.4.1. Phase 1 – Fully separated control.................................... 47*
*3.4.2. Phase 2 – Integration of WAN connectivity and VNFaaS.................................... 48*
*3.4.3. Phase 3 – SDN-based WAN.................................... 49*

**4. OVERALL EVALUATION OF RESULTS AND LESSONS LEARNT ...........................50**

4.1. EVALUATION OF RESULTS.................................... 50
4.2. LESSONS LEARNT.................................... 51

**5. FUTURE RESEARCH CHALLENGES ...........................................................59**

5.1. 5G POSITIONING OF T-NOVA.................................... 59
5.2. TOPICS FOR FUTURE RESEARCH.................................... 59

**6. CONCLUSIONS....................................................................................62**

**REFERENCES........................................................................................63**

**LIST OF ACRONYMS ............................................................................................ 67**

# Table of Figures

Figure 1 - OPNFV Building blocks and Upstream Projects ........ 8
Figure 2 - OSM mapping to ETSI NFV MANO ........ 9
Figure 3 - OSM Release Zero VNF package ........ 10
Figure 4 - Open Baton High Level Architecture ........ 11
Figure 5 - Open Baton Marketplace ........ 13
Figure 6 - Tacker architecture ........ 15
Figure 7 - Tacker SFC architecture ........ 16
Figure 8 - Open-O Architecture Overview ........ 17
Figure 9 - Mapping of ETSI specs into T-NOVA architecture ........ 19
Figure 10 - ETSI NFV architecture ........ 24
Figure 11 - Relationship of MEF LSO with ONF SDN and ETSI NFV MANO ........ 29
Figure 12 - SESAME overall architecture ........ 32
Figure 13 - VITAL overall architecture ........ 34
Figure 14 - HPE pilot hardware testbed ........ 39
Figure 15 - T-NOVA deployment on HPE pilot ........ 40
Figure 16 - OSS/BSS deployment scenario ........ 43
Figure 17 - Multiple PoP Providers ........ 44
Figure 18 - T-NOVA enabled SPs Orchestration ........ 45
Figure 19 - VNFaaS incremental deployment: starting point ........ 47
Figure 20 - VNFaaS deployment phase 1: WAN / VNFaaS full separation ........ 48
Figure 21 - VNFaaS deployment phase 2: WAN / VNFaaS limited integration ........ 49
Figure 22 - VNFaaS deployment phase 3: SDN-based WAN ........ 49

# Table of Tables

Table 1 - Comparing architectures of Open Baton and T-NOVA ........ 12
Table 2 - Generic Networking service deployment flavours ........ 26
Table 3 - T-NOVA lessons learnt ........ 52
Table 4 - Future research topics ........ 60

# 1. INTRODUCTION

Since T-NOVA started in January 2014, the ICT industry at large, and particularly the technological sector around SDN, NFV and Cloud Networking, have evolved quite significantly. It is fair to say that the original vision of T-NOVA anticipated some of the industry evolutions that took place in the last couple of years. In fact, the virtualization of network functions and the ability the deploy network functions "as-a-Service" is no longer just a promising research topic, but rather one of the main innovation catalysts of the ICT industry today.

Now that T-NOVA activities are nearly finalized, the project accomplishments should be assessed not only for their intrinsic value and the adherence to the initial plan, as described in the DoW, but also taking into account the rapid evolution of relevant technologies and the dynamic external environment, with a myriad of research, industry and standardization initiatives taking place in parallel. To name some of the most relevant, outcomes of open source initiatives such as OpenDaylight, ONOS, OPNFV, OSM, and specifications from SDOs such as ETSI NFV, MEF, TMF.

Essentially, the current deliverable aims to:

- provide a final overall evaluation of the project results, particularly taking into account the pilot site evaluation and validation conducted by T7.2 and reported in D7.2;
- position the project results in relation to other relevant ongoing projects, activities and industry initiatives, in the same or in closely related areas;
- provide guidelines for T-NOVA deployment, especially taking into account the lessons learnt in the experimental activities of the project;
- propose a roadmap for deployment of T-NOVA technology in service provider networks;
- identify challenges and propose topics for future research.

With a view to reaching these essential objectives, the document is organized as follows:

- Section 2 provides a technology watch of the most relevant industry initiatives in this area and positions the results of T-NOVA accordingly. A special attention is paid to ETSI NFV and how T-NOVA results compare with the latest ETSI NFV phase 2 specifications, taking into account that the conception of T-NOVA architecture was mainly based on ETSI phase 1 documents;
- Section 3 provides guidelines for T-NOVA deployment; in particular, several scenarios, deployment variations and extensions of the basic model are analysed. In particular, this section provides guidelines for deployment over existing NFV infrastructures, namely HPE's commercial OpenStack release. A roadmap is proposed for incremental deployment of T-NOVA technology in service provider environments;
- Section 4 provides an overall assessment of the project results, based on the outcome of task T7.2. In addition, a list of lessons learnt in the course of the project activities is included;

- Section 5 positions T-NOVA in the context of the telco industry evolution towards 5G, and suggests topics for future research, especially to further explore areas in which the project has been active;
- Section 6 closes the document with overall conclusions.

# 2. TECHNOLOGY WATCH AND T-NOVA POSITIONING

In this section, a technology watch is provided, essentially to position T-NOVA results vis-à-vis relevant open source platforms, standardization bodies, research projects and industry initiatives.

## 2.1. Open Source NFV Frameworks

### 2.1.1. OPNFV

The Open Platform for NFV Project (OPNFV) [1] [2] [3] aims to be a carrier-grade, integrated platform that introduces new products and services to the industry more quickly. OPNFV works closely with the European Telecommunications Standards Institute (ETSI) NFV [4] [5] and others to press for consistent implementation of open standards.

The project focuses on building NFV Infrastructure (NFVI) and Virtualised Infrastructure

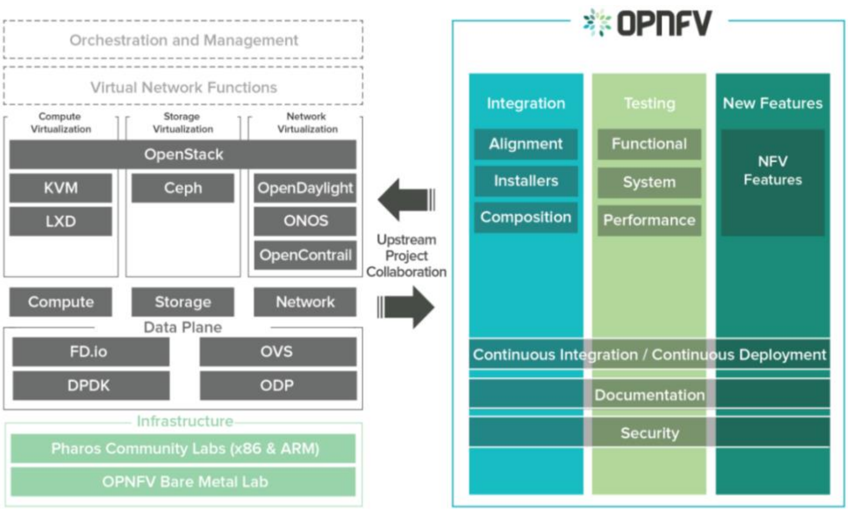



**Figure 1 - OPNFV Building blocks and Upstream Projects**

Manager (VIM); other objectives are [2]:

- To create an integrated and verified open source platform that can investigate and showcase foundational NFV functionality;
- To provide proactive cooperation of end users to validate OPNFV's strides to address community needs;
- To form an open environment for NFV products founded on open standards and open source software;
- To contribute and engage in open source projects that will be influenced in the OPNFV reference platform.

Current release of OPNFV platform is named Colorado. Figure 1 summarises the OPNFV building blocks and the collaboration of upstream projects operating within OPNFV.

#### 2.1.1.1. T-NOVA positioning vs. OPNFV

The work done in T-NOVA related to NFVI and VIM is mostly covered by activities in WP4. The main components used for the implementation of the NFVI/VIM environment are Openstack for the virtualization and also for the VIM implementation and OpenDaylight to support SDN based NFVI-PoPs integrated with Openstack Neutron using NetFloc T-NOVA plugin. In this context, the OPNFV distribution - which builds on-top of the same components - is expected to integrate seamlessly with TeNOR (T-NOVA Orchestrator). Some adaptations might be required considering the SFC (Service Function Chaining) mechanisms that are offered, which in the case of T-NOVA are more basic than the ones supported by the OPNFV release.

### 2.1.2. OSM

"Open Source Mano is an ETSI-hosted project to develop an open source NFV Management and Orchestration (MANO) software stack aligned with ETSI NFV" [6]. In this project several service providers and vendors collaborate and provide their products to develop and test the complete MANO stack, which is similar to T-NOVA's goal. The core of the OSM project, the MANO stack, is implemented using RIFT.ware [7], OpenMANO [8] and Juju [9] which were provided by RIFT.io, Telefonica and Canonical respectively. This project's first release was announced in May 2016 and their first demo was presented in the Mobile World Congress (MWC) in 2016. This release gathered a lot of attention due to their demo in MWC in which they provisioned an NFV-based service mixing multi-tenant and single-tenant VNFs. The complete service is comprised of various VNFs deployed in a distributed PoP scenario and a pre-established inter-datacenter connection. The second release of OSM was announced in October 2016 with some incremental additions to the first release.

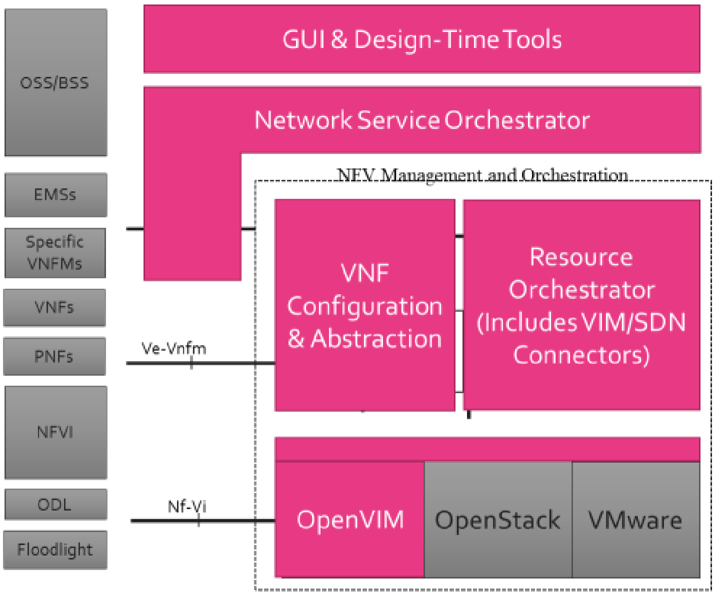



**Figure 2 - OSM mapping to ETSI NFV MANO**

Figure 2 [6] shows the main components for OSM (only the elements in pink are part of OSM code):

- GUI and Design-Time Tools (UI) – graphical interface which allows users to manage the NS (Network Service) and VNF catalogues and lifecycle of NSs. Also, through this interface users can perform NS and VNF level configurations and compose new NSs. This component is realized with Launchpad which is part of RIFT.ware.
- Network Service Orchestrator (SO) – this component provides the End-to-End service orchestration by managing resources through the RO and using configuration plugins such as Juju to manage VNFs. RIFT.ware implements this component.
- Resource Orchestrator (RO) – the RO, which is realized with OpenMANO, is responsible for the creation and management of compute and network resources necessary for the NS instantiation.
- VNF Configuration & Abstraction (CM or VCA) – this component uses Juju to realize the interface with the applications deployed on top of the virtual resources. In essence, Juju provides VNF configuration management capabilities to the SO.
- OpenVIM – OpenVIM is the reference VIM implementation in OSM for all-in-one installations.

The second release of OSM is actually called "Release One" because in the first release the main objective was to provide a first integration of the various components and data models to provide a single entry point to end-users. This first integration still reflects the fact that each of these components was developed independently leading to a fragmented data model and loss of modularity. One example is the VNF package, which is a mixture of Juju and OpenMANO data models, see Figure 3 [6].

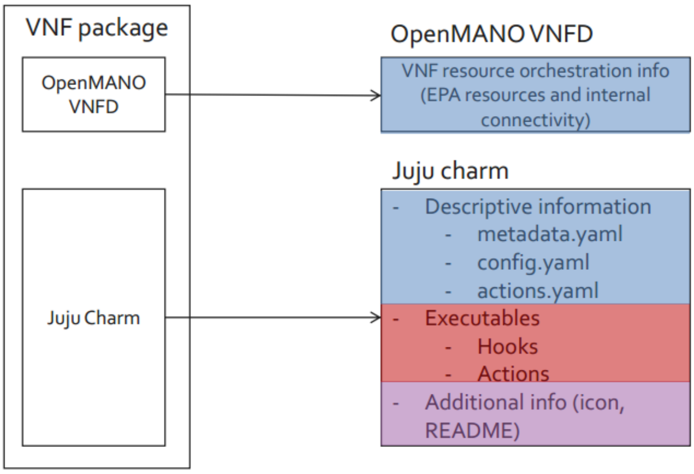



**Figure 3 - OSM Release Zero VNF package**

The second release however tackled three important issues, among others:

- Agnostic Data Model – create a unified data model that better reflects ETSI's specification while abstracting specific technology used for implementing OSM.
- Plugin Framework – specify and implement more generic interfaces between components which will lead to a plugin framework that eases the integration with third party components (e.g. SDN controllers, VIMs).
- Multi-VIM – support for instantiating network services across multiple VIMs.

Addressing the first two issues increased the modularity and abstraction between components, which are two important goals for OSM.

### 2.1.2.1. T-NOVA positioning vs. OSM

OSM and T-NOVA are platforms to provision End-to-End services but with some differences from an architecture perspective – for instance, OSM has an explicit architectural split between Resource Orchestrator and Service Orchestrator. Another aspect that sets them apart is their origin; T-NOVA implemented their components from scratch as micro-services using generic interfaces. This gives the ability to develop or replace any of TeNOR's components freely without loss of functionality, which is not possible in OSM at the moment. Moreover, TeNOR's has a specific module, Gatekeeper, which provides a common security mechanism to the interfaces between components while OSM is dependent on each component security features.

Regarding VIM, both support Enhanced Platform Awareness (EPA) and can be extended to use other VIMs, but OSM already supports VMWare, OpenVIM and OpenStack while TeNOR only supports OpenStack. Although both support a plugin framework to manage and configure VNFs, the latter also features a fully customizable monitoring system, which is the basis for a complete VNF lifecycle management.

Finally, the greatest distinction between these two platforms is that TeNOR also addresses the business aspects of NFV in the VNF and NS data models, while OSM is only focused on the operationalization of NSs. With that in mind, OSM GUI and T-NOVA Marketplace also share some functionality in a graphical interface, e.g. they use a similar approach to on-board VNFs and to compose NSDs.

## 2.1.3. OpenBaton

Open Baton is an ETSI NFV compliant MANO framework, very similar in its functionalities to T-NOVA TeNOR orchestration engine. Although many functions are similar, T-NOVA framework differs in some critical technical capabilities. Figure 4 shows the high-level architecture of Open Baton. This is similar to the design adopted by T-NOVA consortium.

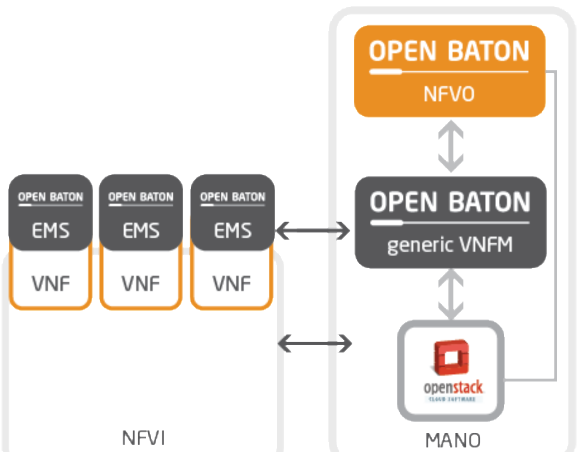
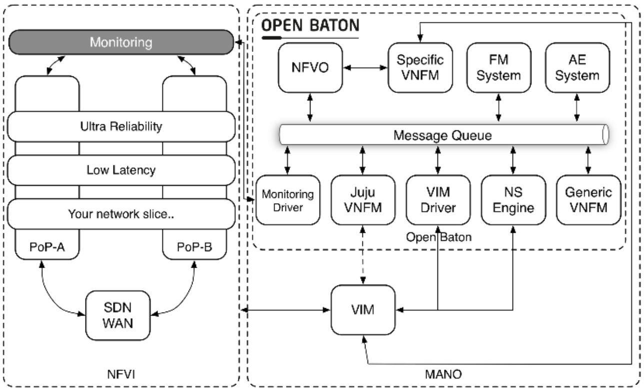


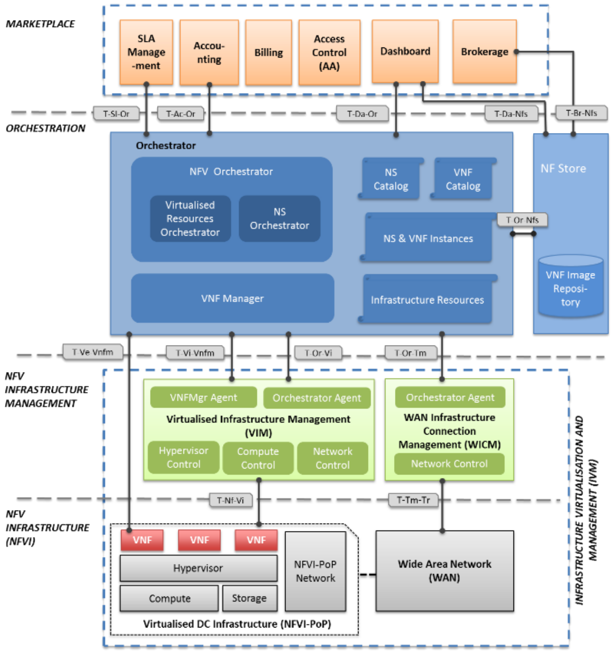

**Figure 4 - Open Baton High Level Architecture**

In T-NOVA the architectural pieces are referred as VNF Manager, NFV Orchestrator (TeNOR), Virtualized Infrastructure Management (VIM) that enables advanced functionalities on top of NFV infrastructures (NFVI). Open Baton has support for more plugins such as ability to work with Canonical Juju VNFM, and additional pub-sub system for dispatching lifecycle events for aiding the execution of various orchestration phases. One stark difference is a better integration in T-NOVA of the SDN-WAN management and orchestration via the TeNOR and NetFloc plugin within T-NOVA VIM. It is worthwhile comparing the detailed architectures side by side, as depicted in Table 1, which clearly shows where the strengths of each solution lie.

**Table 1 - Comparing architectures of Open Baton and T-NOVA**




**Open Baton Detailed Architecture**




**T-NOVA Detailed Architecture**

Here is a list of features which are common to both frameworks:

- Automated scaling.
- Multi PoP deployment of a virtual NS.
- VNF Catalog and service store / registry is also part of both frameworks, although it is more prominent in T-NOVA TeNOR than Open Baton.
- Functional dashboard is included in both frameworks although the T-NOVA orchestrator dashboard (TeNOR) as well as overall marketplace dashboard is more feature rich. Figure 5 shows marketplace section of the Open Baton dashboard.

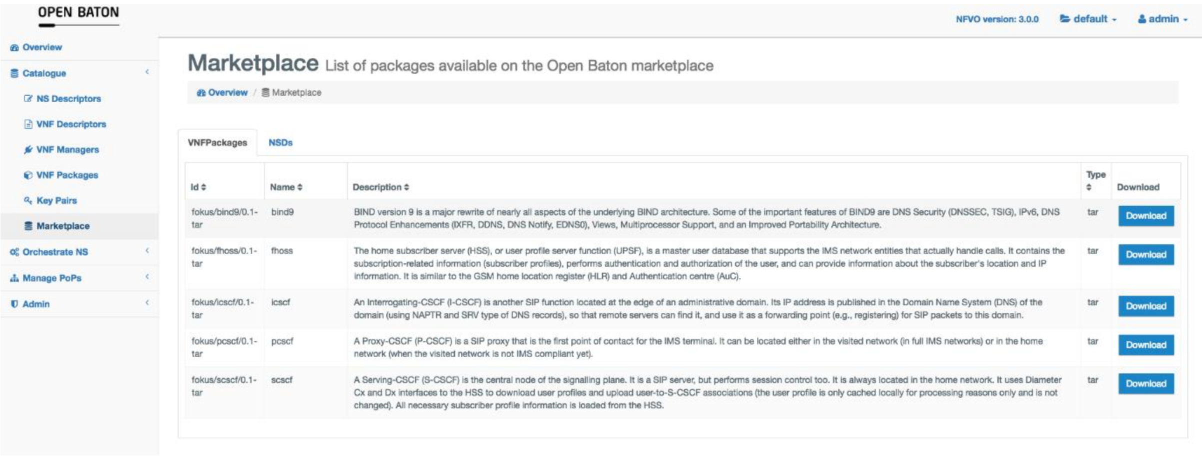



**Figure 5 - Open Baton Marketplace**

### 2.1.3.1. T-NOVA positioning vs Open Baton

T-NOVA offers a richer experience to all actors in the VNF ecosystem including function developers, service operators and marketplace brokers. There are several features available in T-NOVA which are not (yet) present in Open Baton offering. Some of the prominent ones are:

- Support for a rich marketplace including accounting, billing as well as brokerage (although one can argue that these are not necessarily features to be supported by an NFV Orchestrator)
- T-NOVA VIM includes EPA (enhanced platform awareness) that enables fine-grained and online placement optimization engine which is part of TeNOR but missing in Open Baton.
- T-NOVA orchestrator supports Transport Network management between POPs and supports advanced service function chaining capabilities (through NetFloc). Open Baton acquiesce SDN management to external modules not influenced by the orchestrator itself. Open Baton's external module Network-Slicing-Engine[1]♣ performs network slicing and ensuring QoS.
- T-NOVA orchestrator supports advanced SLAs and QoS enforcements. Support for SLAs is rudimentary in Open Baton.
- Gatekeeper supports access logs and auditability which is a critical requirement of many operators. Open Baton documents does not (yet) mention auditability. Although it is foreseeable that extensive logging as supported by both frameworks definitely aids in traceability of events, it is definitely more prominently entrenched in T-NOVA offering.
- T-NOVA has a more extensive Service life-cycle management support built into the framework, whereas till the point of writing, Open Baton has extensive support for VNF life-cycle management only.

---

[1] ♣ Open Baton Network Slicing Engine: https://github.com/openbaton/network-slicing-engine [accessed 22/11/2016]

### 2.1.4. OpenStack Tacker

Tacker is an official OpenStack project started during 2015, to help filling the gap between the mostly IT-oriented features of OpenStack releases and the specific requirements of a suitable VIM component to employ in NFV systems. Specifically, Tacker aims at implementing a generic VNF Manager (VNFM) component and an NFV Orchestrator (NFVO) component, both following the ETSI MANO specifications. Hence, it can be considered as a recently arisen alternative to the homologous T-NOVA components (T-NOVA VNFM and TeNOR).

The technical architecture of Tacker is based on three major modules [10]:

- NFV Catalog
- VNFM
- NFVO

The NFV Catalog specifies and implements the key data models for a NFV system, namely:

- VNF Descriptors (VNFD)
- Network Services Descriptors (NSD)
- VNF Forwarding Graph Descriptors (VNFFGD)

The VNFM takes over the main VNF lifecycle functions, namely:

- Basic life-cycle of a VNF (create/update/delete)
- EPA-based placement of high-performance NFV workloads
- Monitoring of running VNFs
- Policy-based Auto Healing / Auto Scaling of VNFs
- Initial VNF configuration

Finally, the NFVO implements the typical orchestration functions prescribed by the ETSI MANO, namely:

- Template-based end-to-end Network Service deployment
- Policy-based VNF placement
- VNFs interconnection via SFC (described by a VNFFGD)
- VIM Resource Check and Resource Allocation (service mapping)
- Multi-VIM and multi-PoP VNFs orchestration

Figure 6 shows the architecture of Tacker.

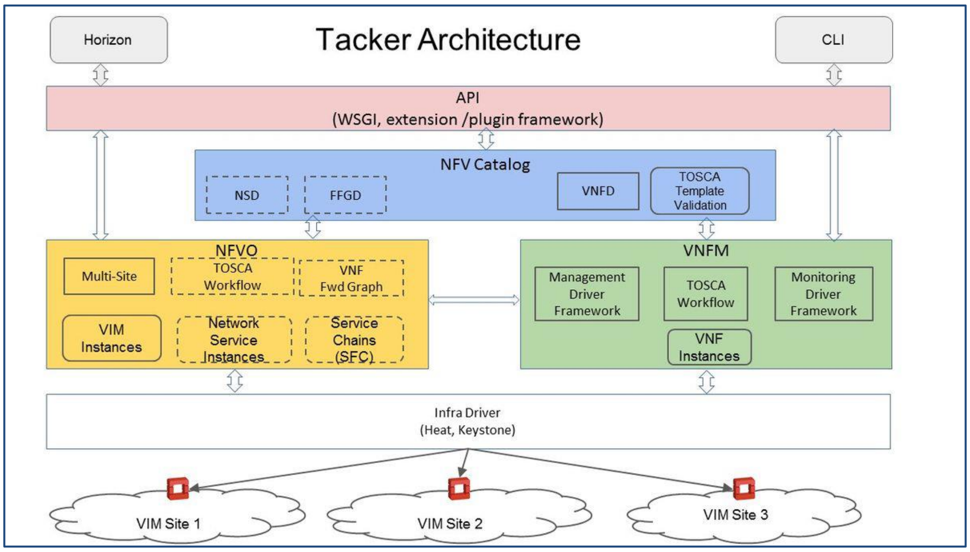



**Figure 6 - Tacker architecture**

The VNFD Catalog is based on the OASIS TOSCA *Simple Profile for NFV* specification [11], which is the data model for the templates used by Tacker to on-board VNFs into the catalog. This is a difference with respect to T-NOVA.

At provisioning time, a translator converts the TOSCA template statements into HEAT template format. The HEAT template is then passed on to Nova as in the standard OpenStack flow.

The VNF configuration phase (in the VNFM) is managed by a configuration driver. Configuration management is structured as a plug-in framework where each VNF vendor can write its own configuration driver. Hence, it supports the dedicated VNFM model. Tacker has also a SDN Controller plugin, allowing to use SDN controller's southbound interfaces to push the configurations onto VNFs (e.g. by OpenDaylight netconf/yang).

As far as the monitoring module, Tacker has some off-the-shelf loadable drivers (e.g. icmp-ping, http-ping, etc.), but at the same time it allows VNF vendors to integrate their own monitoring drivers handling specific metrics. Integration with Ceilometer is also planned.

Figure 7 shows the Tacker SFC architecture [12]. At this moment, SFC is implemented by an ODL SFC driver, programming the OVS instances from the ODL controller (based on the VNFFGD information), and optionally passing further configurations to the VNFs via netconf/yang. Other options (e.g., using a Neutron SFC driver going through Neutron to OVS SFC agents in the compute nodes) are not implemented yet.

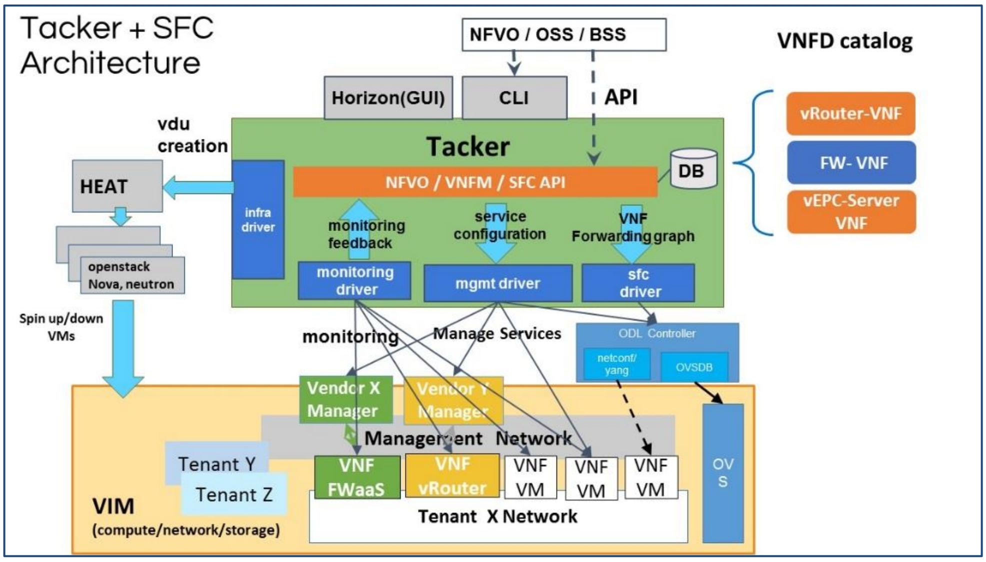



**Figure 7 - Tacker SFC architecture**

### 2.1.4.1. T-NOVA Positioning vs. Tacker

Compared to T-NOVA, against the advantage of being an OpenStack incubated project, Tacker shows at the moment a couple of potential shortcomings:

- It misses a northbound interface towards a Marketplace layer (a competitive advantage for T-NOVA);
- The choice of TOSCA as descriptor format is to date not in line with the ETSI NFV ISG orientation. The latest TOSCA Simple Profile for NFV (CSD 03) only partially implements the ETSI NFV MANO requirements and the latest specifications produced by the ETSI ISG. This situation could evolve in the near future, when the OASIS group will produce and release the CSD 4.

## 2.1.5. OPEN-O

Open-O is a full stack approach where open orchestration bridges the gap between NFV and SDN. In this context, Open-O offers E2E service orchestration, with service providing agility by orchestrating both the NFV as well as the connection parts of the e2e service. The architecture of Open-O is illustrated in the Figure 8 below. Although considered that Open-O is aligned with ETSI MANO architecture, it acknowledges the need to address OSS/BSS with a set of portals to make it easier to create and deploy new services, and to provide a "Design–Time Environment" for new services. MANO also doesn't cover connectivity issues, mainly with the legacy networks. Currently the provided code suffers from code base heterogeneity, quality and errors. The release is currently at v1.0, with installation available via containers of VMs. The second release of Open-O is planned for end of April 2017.

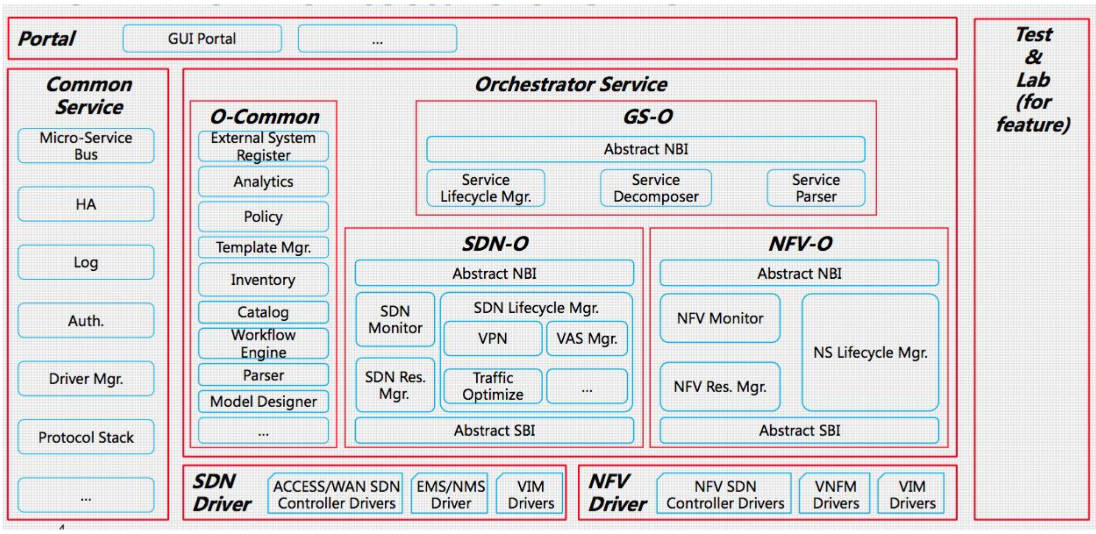



**Figure 8 - Open-O Architecture Overview**

#### 2.1.5.1. T-NOVA Positioning vs Open-O

As the first release of Open-O was made available on November 2016 and while all the main technical WPs of the project had finished, a deep analysis of Open-O was not possible. From the architectural documents accompanying the release of Open-O, it appears that Open-O has a much more "network" flavour embedded mainly due to its inherent support for SDN, via the SDNO component available in the Orchestration Service. The concept of "drivers" at the southbound interfaces allow modularity and interfacing with a variety of network and computing technologies.

The similarities to T-NOVA are exhausted in the use of microservices architecture, support for multi-pop environments with separate administration domains and the support for VNF specific VNFMs via the SBI abstractions.

## 2.2. Standards Developing Organizations

### 2.2.1. ETSI NFV

With respects to the ETSI NFV ISG roadmap, T-NOVA can be positioned as one of the first absolute implementations of the MANO reference architecture, and the very first one providing a marketplace layer, enabling the delivery of a real VNFaaS scenario inclusive of an initial business ecosystem model. Capitalizing on the research, design and implementation work realized during the project, T-NOVA partners (as better described in WP8 deliverables D8.21 and D8.22) have provided direct contributions to the ETSI ISG, namely to EVE and IFA working groups, further reinforcing the linkage between T-NOVA and the whole ETSI NFV standardization effort.

T-NOVA supported the evolution of some areas where its innovation was particularly valuable to enhance the current elaboration performed by the ETSI ISG. One of these is the WAN connectivity layer, fundamental to enable multi-PoP provisioning of VNFs and network services. Here T-NOVA filed contributions encompassing SDN across multiple VIMs, and integration of the NFV platform with WAN connectivity services, leveraging the work done on the WICM component. Other areas where T-NOVA offered its contributions include virtual networking, VNF data modelling, lifecycle management, multitenancy, and the Marketplace novelty.

To successfully cap the interaction with ETSI, a T-NOVA PoC proposal demonstrating the Marketplace and TeNOR integration into a 2-PoP provisioning case has been filed to ETSI, accepted in August 2016 and publicly showcased at the EWSDN event [13] (The Hague, October 2016). So, T-NOVA is leaving significant trails of its work over the whole range of ETSI NFV deliverables.

In the most recent period, when T-NOVA was already at a quite advanced stage, ETSI decided to support and host the Open Source MANO initiative[2], which in May 2016 delivered its Release 0 code, with a Release 1 pre-announced by end 2016. Surely OSM has taken benefit from its later inception, nevertheless T-NOVA has some innovative aspects which can still be considered unique - details can be found in section 2.1.2. At any rate, the successful acceptance of the PoC submission is the best proof of the value acknowledged by ETSI to the main results accomplished by T-NOVA to flesh out the global open NFV landscape.

#### 2.2.1.1. Analysis of recent ETSI NFV specifications

From the beginning, T-NOVA architecture has used ETSI NFV as the main reference model. However, in several aspects, T-NOVA has diverged from ETSI NFV specifications, either because of the specific characteristics of T-NOVA (e.g. main focus on VNFaaS, support of the marketplace module), or simply because stable ETSI specifications did not exist yet when T-NOVA components and interfaces were implemented.

The remaining of this section aims at clarifying the positioning of T-NOVA in relation to a number of recent ETSI NFV specifications (namely from IFA and EVE Working Groups), which map directly to components or interfaces developed by T-NOVA. This can be seen as an update of a similar analysis that was provided in D2.21 [14] (released in the project first year) and updated in D2.22 [15] (released in the project second year). A pictorial representation of the correspondence between recently published ETSI specs and the T-NOVA architecture is attempted in Figure 9.

[2] http://www.etsi.org/technologies-clusters/technologies/nfv/open-source-mano

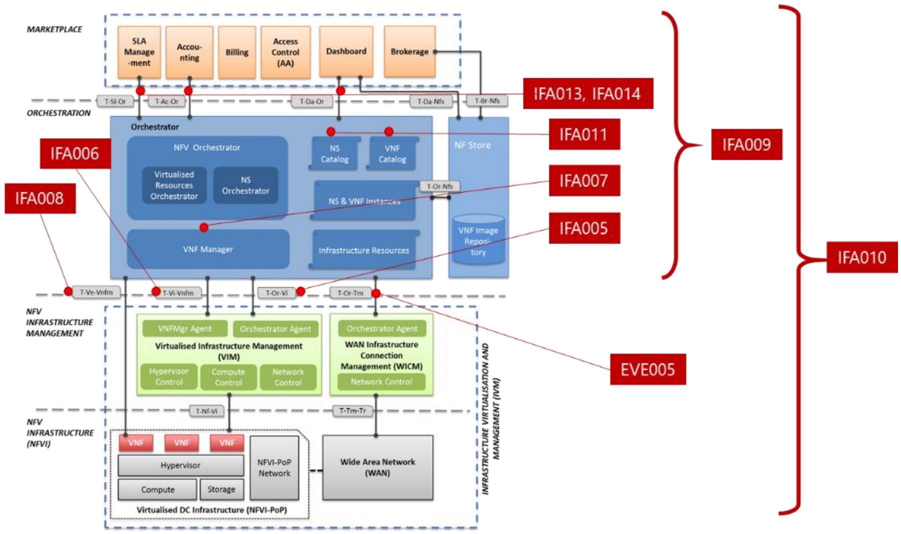



Figure 9 - Mapping of ETSI specs into T-NOVA architecture

**IFA005 Or-Vi reference point - Interface and Information Model Specification**

The IFA005 document [16] overviews the interfaces and information elements associated with the Or-Vi reference point, which is used for the exchange of those information elements between the NFV Orchestrator (Or) and the Virtualised Infrastructure Manager (VIM).

T-NOVA has specified the MANO requirements in Deliverable D2.32 (Specification of the Infrastructure Virtualisation, Management and Orchestration – Final) [17]. The ETSI reference point implementation is referenced as T-Or-Vi in this document and supports the below mentioned interfaces' requirements:

- Software Image Management: VNF images are uploaded into the NF Store by the users (e.g. VNF developers), and then passed to the NFVO Orchestrator, which passes them to the adequate VIM (in a multi-PoP scenario) when an instantiation is needed;
- Virtualised Resources Information Management: supported through the usage of an Infrastructure Repository, which becomes automatically aware of the resources made available in any of the considered PoPs;
- Virtualised Resources Management: see previous bullet point;
- Virtualised Resources Reservation Management: not supported;
- Virtualised Resources Change Notification: implemented the Interface Repository, which gets notified and reacts accordingly whenever new resources are made available in the PoPs to be considered;
- Virtualised Resources Reservation Change Notification: not supported;
- Virtualised Resources Performance Management: implemented through monitoring and with the single purpose of automatically scaling out/in instantiated services, and not in an abstract 'performance management' scope;
- Virtualised Resources Fault Management: implemented indirectly, by the NFV Orchestrator to be able to see that a failure has occurred (but not, e.g., the automatic reaction of rebooting a VNF component that has failed).

Besides this interface, T-NOVA has also implemented an interface with the WAN – WIM (WAN Infrastructure Manager), as per the ETSI terminology; WICM (WAN Infrastructure

Connectivity Manager) as per T-NOVA terminology – which allows the basic establishment of a connection between two different PoPs.

**IFA006 Vi-Vnfm reference point - Interface and Information Model Specification**

The IFA006 document [18] provides an overview of the interfaces and information element associated with the Vi-Vnfm reference point. The Vi-Vnfm reference point is used for exchange of information elements between the Virtualised Infrastructure Manager (VIM) and VNF Manager (VNFM).

As stated above, T-NOVA has specified the MANO requirements in Deliverable D2.32 (Specification of the Infrastructure Virtualisation, Management and Orchestration – Final) [17]. The ETSI reference point implementation is referenced as T-Vi-Vnfm in this document. The base for the definition of the T-Vi-Vnfm interface was the phase 1 ETSI NFV released documents. According to these documents, two distinct operational choices could be followed. The first one is the use of direct Or-Vi interface for control of the VIM resources and Software Image Interface and the second is the use of indirect control exploiting Vi-Vnfm interface. T-NOVA selected the first method, thus a large part of the requirements related to Vi-Vnfm are implemented for the T-Or-Vi implementation. In this context, the T-Vi-Vnfm interface is responsible for the exchange of infrastructure monitoring information either through explicit request by the Orchestrator or through periodic reporting initiated by the VIM. The types of data exchanged over this interface include status, performance and utilisation of infrastructural resources.

More precisely, the below mentioned interfaces' requirement are discussed:

- Software Image Management
- Virtualised Resources Information Management
- Virtualised Resources Management
- Virtualised Resources Reservation Management
- Virtualised Resources Change Notification
- Virtualised Resources Reservation Change Notification
- Virtualised Resources Performance Management
- Virtualised Resources Fault Management

The document provides refined information on the interfaces exposed per component (i.e. VNFM and VIM), as well as the Information Model used. It is apparent that T-NOVA has implemented an operational subset of the specified functionalities, considering the timeframe for implementation but also the planned objectives for the project.

**IFA007 Or-Vnfm reference point - Interface and Information Model Specification**

The IFA007 document [19] overviews the interfaces and information elements associated with the Or-Vnfm reference point, which is used for the exchange of those information elements between the NFV Orchestrator (Or) and the Virtual Network Functions Manager (VNFM).

As stated above, T-NOVA has specified the MANO requirements in Deliverable D2.32 (Specification of the Infrastructure Virtualisation, Management and Orchestration – Final) [17]. The ETSI reference point implementation is an internal interface and supports the below mentioned interfaces' requirements:

- Virtualised Network Function (VNF) Package Management (produced by NFVO, consumed by VNFM): implemented through the use of a VNF Descriptor (meta-data) and the VNF image file (although not in one single file);
- VNF Lifecycle Operation Granting (produced by NFVO, consumed by VNFM): not implemented, due to the VNFM to be considered as an internal component of the whole NFVO Orchestrator – this would be one of the evolutions needed to support the case of an externally provided VNFM;
- Virtualised Resources Management (produced by NFVO, consumed by VNFM): implemented through the (optimal) VIM and WIM resources, by checking the Infrastructure Repository for their availability, characteristics, etc.
- VNF Lifecycle Management (produced by VNFM, consumed by NFVO): implemented through the (optimal) allocation of VIM and WIM resources in each instantiation request, automatic scaling out/in, etc.;
- VNF Lifecycle Change Notification (produced by VNFM, consumed by NFVO): not implemented in TeNOR, feedback reaches TeNOR only indirectly, through monitoring;
- VNF Performance Management (produced by VNFM, consumed by NFVO): implemented by monitoring, which follows a set of parameters the VNF developer defines in the VNF descriptor and the NFVO subscribes for each instance that is created;
- VNF Fault Management (produced by VNFM, consumed by NFVO): part of the monitoring cycle, but not explicitly for fault management;
- VNF Configuration Management (produced by VNFM, consumed by NFVO): implemented through the mAPI module; VNF Indicator (produced by VNFM, consumed by NFVO): implemented as the monitoring cycle, in which the VNF related monitoring parameters are subscribed by the Orchestrator upon an instantiation request.

**IFA008 Ve-Vnfm reference point - Interface and Information Model Specification**

IFA008 [20] is a really important specification because it defines the interfaces between the VNFs and the VNFM, which is the Ve-Vnfm reference point. In T-NOVA this corresponds to the T-Ve-Vnfm. The ETSI reference point actually includes both the Ve-Vnfm-em reference point, used for exchanges between EM and VNF Manager, and the Ve-Vnfm-vnf reference point used for exchanges between VNF and VNF Manager. The first big difference regarding T-NOVA is the following: due the evolutionary approach of the project, the EM functional block is not included in T-NOVA leaving the possibility of flexible solutions for the management aspects of the VNFs. Therefore, it is possible to compare only the Ve-Vnfm-vnf part.

A general remark about IFA008 is that with this specification ETSI missed a big opportunity. In fact, it was opportune to anticipate as soon as possible the definition of this document being it necessary to easily interoperate new VNFs with a generic VFNM. On the contrary, the specification was released only last October and the big issue is that it is not the Stage 3 specification and it is not usable. IFA008 provides only a general description of the possible operations and the related information elements. Besides the specification contemplates a great number of interfaces: VNF Lifecycle Management, VNF Performance Management, VNF Fault Management (produced by VNFM and consumed by VNF); VNF Indicator and VNF Configuration (produced by

VNF, consumed by VNFM). Only a small subset of them are deemed mandatory in practical situations.

The final result of this complexity is that we still lack a real specification and this implies delays in the progress towards an open NFV ecosystem. Whenever a new VNF shall be managed it is necessary to apply customizations and VNF vendors can continue to claim that their specific VNF Manager is mandatory.

It is important to note that a new specification is under development in the ETSI group, the GS NFV-SOL 002: RESTful protocols specification for the Ve-Vnfm Reference Point. The foreseen approval date is the end of July 2017.

T-NOVA has specified the T-Vnf-Vnfm interface taking into account the output of the ETSI NFV Phase 1 and according to the T-NOVA MANO requirements included in Deliverable D2.32 (Specification of the Infrastructure Virtualisation, Management and Orchestration – Final) [10], which was delivered in October 2015, i.e. almost one year before the IFA008 ETSI document. Comparing the two specifications, it is possible to say that the T-NOVA approach is aligned with the ETSI vision and that possible deviations concern secondary aspects.

The first difference, as previously stated, regards the interfaces with the EM. The second one regards the possibility for the VNF to retrieve Fault and Performance information from the VNFM.

T-NOVA adopts REST APIs and is in line with the future evolution of IFA008, which in the Stage 3 specification available next year, namely NVF SOL002, will adopt the REST protocol.

**IFA009 Report on Architectural Options**

IFA009 [21] is essentially an "exercise", studying and presenting several alternatives (options) for the assignments of functionalities to the various building blocks of the NFV architecture.

T-NOVA has considered most of these alternatives, well before this document was released, and has a clear and justified positioning for most of them. In particular:

- *Choice between generic and VNF-specific VNFM:* T-NOVA defined a generic VNFM, adopting a vendor-neutral API (mAPI) for the communication between VNF and VNFM (Ve-Vnfm).
- *Choice between having the resources managed by the NFVO or the VNFM:* in T-NOVA, the resources needed are identified by the NFVO (and allocated by the VIM upon request)
- *Choice between various NFVO functional splits:* in T-NOVA, network service orchestration is coupled with resource orchestration. The Infrastructure Repository maintains a global view of the infrastructure resources and the Service Mapping module identifies the resources needed. These capabilities correspond to the ETSI Resource Orchestrator functional module.
- *Choice between EM functions as VNFCs (embedded in the VNF) or separate VNF:* T-NOVA mostly selected the first option, arguing that, in many (especially simpler) VNFs, it is absolutely valid to consider the EM functionality (e.g. web front-end or SNMP agent) embedded in the VNF itself.

**IFA010 Functional requirements specification**

IFA010 [22] departs from the high-level concepts and requirements as set in GS NFV 001-004 [23], published in 2013, and attempts to define a comprehensive set of functional requirements for the NFV Management and Orchestration components.

T-NOVA has specified the MANO requirements in Deliverable D2.32 (Specification of the Infrastructure Virtualisation, Management and Orchestration – Final) [17], which was delivered in October 2015, i.e. almost six months before the ETSI document.

Comparing the two sets of specifications, it can be seen that, generally, T-NOVA is well aligned with the ETSI vision on MANO, at least in terms of functional requirements, associated with critical steps of VNF lifecycle management, such as creation, on-boarding, update, monitoring, management and termination. Some minor deviations are presented below. These deviations mostly concern the NFVO/VNFM functionalities; the requirements for the VIM layer are almost identical.

First, ETSI foresees a "resource reservation" model, under which customer resources are committed but not allocated. T-NOVA MANO allocates resources only upon request (on demand) for the specific service. Also, ETSI defines that resources and tenants may be grouped, and groups of resources can be assigned to group of tenants. Also, tenants may have specific resource quotas. Although in T-NOVA these requirements have not been explicitly included, similar capabilities can be implemented using the native Openstack features.

Second, ETSI includes several requirements for fault management and on-demand healing. T-NOVA supports fault notification based on alarms, yet no specific fault management component has been included in the MANO stack.

On the other hand, T-NOVA includes requirements not adequately covered by ETSI. These refer mostly to the interface with the Marketplace (or OSS/BSS), which are critical in order to deploy the MANO environment in an operational concept. Also, T-NOVA covers more thoroughly the networking domain, including explicit requirements for the WICM, thus effectively addressing the service connectivity component.

**IFA011 VNF Packaging Specification**

The IFA011 document [24] provides requirements for the structure and format of a VNF Package describing the VNF properties and associated resource requirements.

The VNF package in ETSI is the artefact delivered to service providers by VNF vendors and it contains all of the required files and meta-data descriptors required to validate a VNF and manage its lifecycle: standardized meta-data descriptors describing the virtualized resource requirements, the design constraints and other dependencies; the definition of the operational behaviour including VNF lifecycle events.

T-NOVA adopts a more agile model, based on the T-NOVA Marketplace but anyway it leverages the VNF package performing the on-boarding of the VNF package from the Function Store where independent Function Providers make available their VNFs.

A relevant component of the package is the VNF Descriptor. T-NOVA adopted a VNF Descriptor based on ETSI NFV MAN001 [25], published in December 2014. Therefore, it is not aligned with the last specification that was issued in October 2016: namely ETSI GS NFV-IFA 011 V2.1.1 (2016-10) [24]. The main differences between the two specifications are listed in a document of the ETSI NFV SOL WG [26]. In IFA011 it is possible to find new information elements or information elements with a different semantic. Nevertheless, it is possible to state that for the validation of the T-NOVA concept these differences are not important. Anyway, the T-NOVA team continuously monitored the progress of the IFA011 specification providing different contributions.

It is important to point out that a liaison on NFV descriptors exists between ETSI NFV and OASIS. Therefore, there is an ongoing work both on VNF Descriptor (VNFD) and Network Service Descriptor (NSD). It is foreseen that ETSI NFV will adopt the TOSCA profile and a new specification will be published as GS NFV-SOL 001. The foreseen date of availability is October 2017. It is possible that this specification will be an endorsement of a specification under development by an ad-hoc group of the OASIS TOSCA Technical Committee. With the availability of this specification it will possible to obtain a precise definition of: the VNF Package, the structure and the format (archive structure, file naming conventions, MIME types for file contents, etc.).

**IFA013 Os-Ma-Nfvo reference point - Interface and Information Model Specification**

T-NOVA introduced the NFV marketplace concept aiming at opening the NFV market to third party developers for the provision of VNFs, a concept that fell outside the technical view of ETSI NFV. At the same time, for the service provider perspective, T-NOVA Marketplace defined some interfaces between the Marketplace and the NFVO (TeNOR) that were covering the Os-Ma ETSI reference point when ETSI did not provide yet details about the information through this reference point. T-NOVA Marketplace, though not implementing a proper OSS/BSS system, includes some OSS/BSS functionalities addressing particularities of NFV (i.e. billing, accounting, SLA monitoring, Authentication, Authorisation, and Accounting (AAA)) that can be a good input towards the real integration between NFV orchestration and existing OSS/BSS systems.

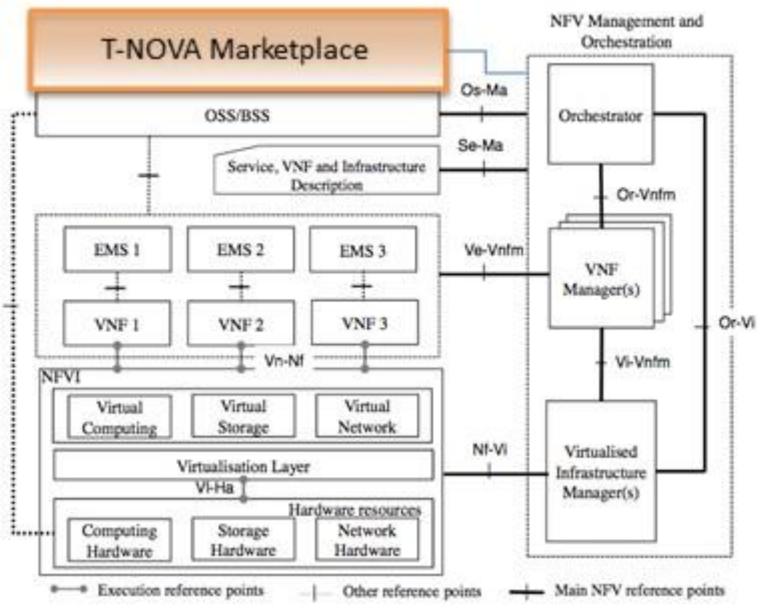



**Figure 10 - ETSI NFV architecture**

Based on the information provided now by IFA013 [27] in Oct'16 we can map and compare current ETSI Os-Ma-Nfvo reference point - Interface and Information Model Specification to T-NOVA interfaces between the Marketplace and TeNOR in the following way:

- Network Service Descriptor (NSD) Management: T-Bsc-Or to on-board, enable, disable, updated, delete and query NSDs. As future roadmap for T-NOVA, analogous operations are defined by ETSI for PNFs.
- Network Service (NS) Lifecycle Management and Lifecycle Change Notification: T-Ac-Or to create a NS instance identifier and update and query a NS, T-Da-Or to terminate a NS, TSs-Or to instantiate a NS. Future roadmap for T-NOVA can be a NS scaling triggered by the service owner or by policies coming from the OSS/BSS.
- NS Performance Management: T-Sl-Or to collect and report performance (SLA) information (that in T-NOVA is later use to create penalties/rewardings) and partially done through T-NOVA NSD creation, since thresholds for NS performance management are included as part of T-NOVA NSD. ETSI groups these operations by a PM (Performance management) Job.
- NS Fault Management: T-Ac-Or. T-NOVA accounting module receives a notification of failure in case a NS has not been instantiated correctly. Other events related with NS instances are also covered. More complex alarms system can be considered for future work roadmap in T-NOVA.
- VNF Package Management: T-Da-Fs and T-Fs-Or to on-board, enable, disable, delete, query fetch a VNF package.

**IFA 014 Network Service Templates Specification**

T-NOVA NSD (Network Service Descriptor) specification from the Marketplace perspective requirements was built as a result of applying TMForum SID model to ETSI NFV information model. This was done adding on top of the ETSI NSD [25], a related customer facing Network Service description, including fields in the NSD that allow the business interaction among the stakeholders that interact in the T-NOVA Marketplace, namely SLA specification, and pricing. This made that T-NOVA manages a unique Network Service Descriptor shared between customer facing catalogues and orchestration catalogues, so the business service offerings are mapped directly to service deployment by TeNOR. Each layer (marketplace and TeNOR) will be using descriptor fields relevant for their operations.

The latest specification by ETSI about the NSD [28] provides some further details that were not in place when T-NOVA NSD was created, however, we can find still some differences related with the business activity explained in the previous paragraph:

- T-NOVA NSD took the assurance parameters proposed by ETSI in order to build an actual service assurance scheme that allows the customer to be economically compensated in case the SLA has not been met [28]. And those *SLAs are linked to price offerings*, which is something that ETSI does not address.
- Another future roadmap that we find in ETSI NSD is related to the *Network Service Deployment flavour*, which was identified in previous work in T-NOVA [29]. ETSI uses the Gold, Silver, Bronze notation for the definition of flavours of a particular NS that is composed by a number of VNFs and a Connectivity Service, while in T-NOVA is used to name a group of technical parameters for the SLA specification. However, that notation, as it is defined at the moment, does not correspond to any particular networking principle/rule common to all the possible network services compositions. The analogy that we can think is coming from the relevant usage of the three-colour marker in networking. The

table below attempts to provide a generic framework for the definition of this approach.

**Table 2 - Generic Networking service deployment flavours**

| Flavour name | Properties |
|---|---|
| Gold | - Highest Priority Service<br>- Scaling requirement in terms of resources are taken into account (always available)<br>- Network traffic QoS equal to EF or at least AF1x or whatever the supported service differentiation allows<br>- Access to the IT resources should be prioritised |
| Silver | - Statistical Prioritisation for the service<br>- Guarantee the minimum requirements in terms of resources as those are specified by the NS, however able to provide additional resources in case they are available on the NFV-PoP.<br>- Network resources could follow the established Assured Forwarding class service differentiation that has the same notion as the above for the IT resources<br>- Access to the IT resources could be prioritised among different Silver services from multiple tenants or the same tenant (complicated) |
| Bronze | - Equal to a Best Effort service but with an asterisk<br>- For IT resources No scaling is allowed<br>- For Network resources No calling is allowed and the traffic is always mapped to Best Effort class.<br>- The system guarantees the IT resources required for the service to be operational. |

In this way we would have a service deployment flavour that is defined independently of the kind of service, based on policies and priority rules in case of congestion, which is aligned with what is understood in the networking world for a deployment flavour, unlike the expected performance of the specific service by means of QoS parameters.

Other information elements included in ETSI Network Service Template specification and its comparison to T-NOVA one are the following:

- *VNFFGD:* The information in the descriptor, is relevant to links that are connecting VNFs that in the case of T-NOVA may or not be residing on different PoPs. Those links are QoS relevant and their descriptors should be extended or correlated to the specifics contained in the VLD regarding Virtual Link (VL) QoS parameters. For VLs that span across PoPs the entity responsible for the preservation and enforcement of QoS policies is the WAN Infrastructure and Connectivity Manager (WICM). In T-NOVA for simplification QoS has not been thoroughly tackled due to limitation in the utilized technologies at the PoPs.
- *VLD*: one of the information fields specified as part of the VLD by ETSI are QoS related parameters for the Virtual Link, e.g. latency, jitter, packet loss. In T-NOVA for sake of simplicity it was assumed that the virtual links are not adding QoS issues, and it is something that it is considered as part of the future roadmap of T-NOVA.
- *Scaling:* Scaling is a very complex topic in the application of NFV in current state of the art systems, due to stringent requirements posed by the very nature of the Network Services. Even for a simple scale out of a network service (e.g. spawn of a new instance of a VNF) a lot of network re-configuration needs to take place or vice versa a change in the configured Committed Information Rate (CIR) requires also scaling of the VNFs in order to cope with the new network requirements. For the moment only scaling out of VNFs is considered in T-

NOVA assuming that the network resources are in-place or adequate for the proper operation of the NS.

- *PNFD*: this was out of scope of T-NOVA which was only focused in NSs composed by VNFs and not their integration with physical network functions. It could be considered as future roadmap building upon T-NOVA solution

**EVE005 Report on SDN Usage in NFV Architectural Framework**

EVE005 [30] provides an overview of SDN in relation to ETSI NFV architectural framework, describes common SDN/NFV design patterns, describes different scenarios in which SDN technology might be integrated into the NFV framework and derives a number of recommendations targeted at present and future ETSI NFV work and generally all NFV-related activities.

From the point of view of T-NOVA, it is interesting to note that EVE005 is the first ETSI NFV specification that addresses the role of the WIM (WAN Infrastructure Manager), of which a rough initial description had been provided in the original ETSI MANO specification [25]. In EVE005 the WIM is considered as part of the solution to enable multi-VIM/multi-PoP scenarios, particularly as a way of leveraging SDN to control traffic across WAN domains.

On a superficial analysis, ETSI NFV WIM and T-NOVA's WICM could be considered equivalent, as they both handle connectivity between NFV domains across WANs. However, the WICM goes significantly beyond the WIM (thence, the different name). Main differences to be noted are:

- The interface between the Orchestrator and the WICM (T-Or-Tm) is a fundamental T-NOVA architecture reference point, as specified in Deliverable D4.22 [31] and a fully functional interface, as shown by several T-NOVA demos. By contrast, ETSI's WIM is at this stage little more than a placeholder for a component to be specified. It is not even clear at this stage if the interface between ETSI MANO NFVO and the WIM will be a new reference point on its own, or a special case of the Or-Vi reference point.
- The WICM is especially targeted at handling VNFaaS, in which VNFs are consumed by customers who are supposed to also subscribe to some form of WAN connectivity service – in the case of enterprise customers, typically a L2/L3 IP/MPLS VPN – actually, in many cases, VNFaaS (e.g. Firewall as a Service, Transcoding as a Service) can be seen as an add-on to a pre-existing connectivity service. Therefore, the integration of WAN connectivity services with VNFaaS services is a key WICM requirement. The integration of WAN connectivity services and VNFs is so far beyond the scope of ETSI NFV WIM.

## 2.2.2. TMForum

As part of T-NOVA SOTA analysis, TMForum activities [32] were explored in previous work in T-NOVA [15] [33] (mainly related with NFV by means of TMForum ZOOM project [34]), which are aligned with the overall objective of TMF towards the growth of business and business agility in the telco work. From TR228 TM Forum Gap Analysis related to MANO Work [35], it was gathered that one of the points missing from NFV

MANO, was a detailed implementation model on how to manage operational and business support systems in a hybrid legacy and virtualized environment. T-NOVA has provided a first step on the direction of this research line by means of the implementation of an NFV marketplace, which implements some of the functionalities of a BSS system. What can be an important first input for latest studies is the still pending full business interoperability of 5G with OSS/BSS existing systems, which TMF ZOOM was planning to address in the future [32]. More recent TMF reports state that though CSPs (Communication Service Providers) aim to use SDN and NFV technologies to improve agility and reduce costs, the problem of having these technologies operationally deployed and managed to meet the business objectives remained unsolved [36] [37] [38]. For instance, features identified by TMF literature that would be in the path of T-NOVA roadmap evolution are: closed loop assurance, including specific SLAs compatible with NFV, extended accounting systems to manage the use of both physical and virtual resource and support of complex pricing scenarios and revenue sharing models derived from 5G multi-provider scenarios.

Therefore, T-NOVA can be positioned as one of the first designs and implementations of an NFV marketplace layer that enables business transactions for VNFaaS scenarios. One of the key novelties of T-NOVA marketplace ahead of TMForum is the novel SLA management framework specific to NFV, which was built upon the SLA considerations of TMForum for cloud services. Another novelty is the new revenue sharing model between Service Providers and VNF developers as part of the same value chain in the NFV ecosystem by means of an NFV accounting component compatible with NFV orchestrator ETSI NFV compliance (TeNOR).

The overall specification of the implementation of T-NOVA Marketplace was submitted to TMF ZOOM project as potential contribution [39] in March 2016 [40]. The initial feedback from TMForum was the T-NOVA Marketplace to be considered scheduled for Vancouver ZOOM meeting (12th July) [41] as a candidate use case for the HNWPaaS (Hybrid Network Platform as a Service) discussion, however we were finally notified that its most relevance potential was found towards the Red team Procurement and onboarding discussion. T-NOVA Marketplace came up in the context of a Red Discussion on on-boarding and an Open Source Community workshop held in Kista Stockholm (13th Oct), and it is listed now in the on-going action items planned to be discussed within R17, as it can be seen in [42]. Therefore we can state that besides the functionalities on SLA and accounting/billing within T-NOVA explained above, current TMForum on-going work for which T-NOVA is relevant is the process of on-boarding services. It is expected that T-NOVA front-end functionalities to create VNFDs and NSDs including their commercial offering and their onboarding into service orchestrator (TeNOR) will be relevant within that context, from what we expect further feedback in a few months.

### 2.2.3. MEF

MEF's Lifecycle Service Orchestration (LSO) is intended to provide an agile approach to automate the service lifecycle across multiple network domains involved in the delivery of an end-to-end connectivity service. Within each domain, the network infrastructure may be implemented with traditional WAN technologies, as well as NFV and/or SDN. LSO, in combination with SDN and NFV, is designed to enable MEF's Third Network vision, based on network-as-a-service principles and agile delivery of assured connectivity services orchestrated across network domains between physical and virtual service endpoints.

LSO supports service delivery orchestration against one or multiple delivery systems, including WAN controllers, SDN Controllers, EMSs (Element Management Systems), NFV Orchestrators, SDN Orchestrators, to fulfil customer orders or service control requests.

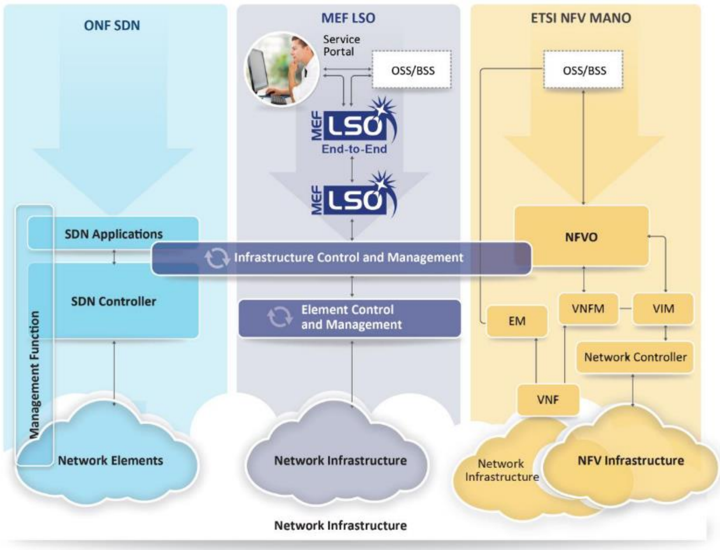



**Figure 11 - Relationship of MEF LSO with ONF SDN and ETSI NFV MANO**

The relationship between LSO and NFV/SDN is depicted in Figure 11 [43]. From an NFV perspective, the NFV Orchestrator essentially provides APIs that enable the abstraction of elastic data center resource management requirements (managed through VNFMs and VIMs) to instantiate or modify network functions and network services. Thus, the MEF LSO layer can request the dynamic instantiation of network functions or network services and delegate the management of the data center resources to the respective NFVO.

MEF and T-NOVA interests are different, in the sense that MEF is mainly focused on WAN connectivity services and views NFV and SDN essentially as enablers of agility and service automation, whereas T-NOVA is mainly focused on the VNFaaS service model, of which WAN connectivity is one of the key ingredients. Common to both MEF and T-NOVA visions is the seamless integration of WAN connectivity services with VNFs and VNF services. On the other hand, through the marketplace concept, T-NOVA can claim to go a step beyond MEF's LSO and Third Network visions, by enabling not only the agile delivery of network connectivity services, but also the creation of an open market of virtual network functions.

In terms of interoperability, T-NOVA platform (i.e., orchestration, VIM, NFVI layers), essentially as a specialized NFV MANO platform, could be integrated in a wider LSO environment in a relatively straightforward manner, provided that the required API adaptations are carried out.

## 2.3. R&D Projects

### 2.3.1. 5G-PPP Projects

T-NOVA has settled an important technical basis about NFV orchestration, NFV business aspects and others towards several 5G ongoing projects. Details are provided below in this section. Also, T-NOVA outputs are relevant to the 5G-PPP association, having been demonstrated that NFV is one of the key technological enablers for the real uptake of 5G [44]. Several 5G-PPP Working Groups (WG) [45] have looked at T-NOVA results, especially the Software Networks WG (which is led by the Atos group that is part of T-NOVA consortium) [46]. Further details on the envisaged future relevance of T-NOVA for 5G are provided in section 5.1.

#### 2.3.1.1. Sonata

SONATA [47] is a Horizon 2020 project, part of 5G-PPP Phase 1 cluster. The main goal of the project is to dramatically reduce the time to market for networked services by shortening service development and deployment. This is achieved by providing a Software Development Kit (SDK) that provides the developer with the necessary tools to quickly obtain feedback about the Network Service or Virtual Network Function he/she is developing using an emulator, a Service Platform that accepts, validates and deploys those services and functions and provides monitoring data back to the SDK, thus allowing the services or function to be fine-tuned. New versions of the same service or function can then be uploaded to the Service Platform and their instances updated, thus reducing both the time it takes to make an initial deployment and all the remaining deployments after.

The SONATA Service Platform builds on top of TeNOR, T-NOVA's open-source orchestrator, taking advantage of the micro-service oriented architecture of the later. This architectural style fits perfectly into the platform needs, especially into one of the innovations it presents: instead of uniform Network Service Manager and VNF Manager, SONATA allows developers to write their own managers, Service Specific Managers (SSMs) and Function Specific Managers (FSMs), and submit them together with their service or function. These SSMs/FSMs are specialized in a specific aspect (e.g., placement, scaling, etc.) and can therefore change in a very efficient manner the way that aspects of a certain service or function are managed. When no SSM/FSM is provided, a default one is activated when the service or function is instantiated. SSMs and FSMs are really micro-services that are plugged into the platform, with the necessary authentications and authorizations taken care of. For example, a Placement SSM/FSM in SONATA can therefore play the role the Service Mapping module plays in T-NOVA.

In T-NOVA we have also focused solely in OpenStack as the VIM, while SONATA pursues the possibility of supporting other kinds of VIMs, like the container based Kubernetes (this is still work in progress).

While in T-NOVA the monitoring feedback cycle is closed (within TeNOR), in SONATA, and because we have the SDK, this feedback can be extended to the developer directly, making him/her aware of the performance of his/her service or function instances. This can dramatically accelerate the feedback the developer has and therefore allow for more often improvements, taking advantage of the short deployment times.

#### 2.3.1.2. 5GEx

5GEx [48] is a Horizon 2020 project, part of 5G-PPP Phase 1 cluster. Its goal is to enable the creation and deployment of *cross-domain* services, seamlessly orchestrating computational and connectivity resources coming from different technical and administrative domains. 5GEx targets a reference architectural design, a reference prototypal implementation and an extensive trialing on a multi-partner sandbox, as well as the definition of proper business models complementing the technology and supporting its actual market deployment.

Among the key enablers for 5GEx to accomplish its objectives is the ability to integrate NFV resources (VNFs and network services) into the scope of its multi-domain orchestrator (MdO). Analyzing its fundamental use cases, on the way to design and specify its architectural model, 5GEx identified the need to create resource slices consisting of black-box, service-wrapped VNFs – or, in other words, VNFaaS kind of resources. T-NOVA was identified as the best reference platform to elaborate the VNFaaS use case category, providing a starting point for the design and possibly for the first prototypal implementation. So, there is a clear synergy and logical connection between T-NOVA and 5GEx.

In particular, 5GEx has leveraged the T-NOVA marketplace layer, as reference component to design and implement its interface I1 (front-end to cross-domain service customers). The T-NOVA data model for network services has been taken by 5GEx as starting point, extended by the addition of cross-domain related information. The T-NOVA open Marketplace software has been leveraged to build the 5GEx service creation component, catalogue management and SLA management by properly readapting its southbound interface to properly interact with the 5GEx MdO.

Bottom line, as far as 5GEx and cross-domain orchestration are concerned, T-NOVA has definitely proven to be a relevant technology enabler. To 5GEx, the Marketplace is the most valuable T-NOVA component, since it offered a solution for a key part of 5GEx architecture, where at the moment no effective alternatives have been identified. TeNOR has also been evaluated as baseline component for the multi-domain orchestrator (MdO). In the end, 5GEx has not directly reused TeNOR code, since some MdO requirements (especially concerning the lower layer connectivity aspects) were not fully met by TeNOR. However, in the long term 5GEx vision, single domain orchestrators like TeNOR will leave aside the MdO, in a layered architectural model where intra-domain orchestration will not be reimplemented, and the MdO will act at a higher layer.

#### 2.3.1.3. SESAME

H2020 SESAME project [49] addresses the placement of network intelligence and applications in the network edge through NFV and Edge Cloud Computing; and the consolidation of multi-tenancy in communications infrastructures, allowing several operators/service providers to engage in new sharing models of both access capacity and edge computing capabilities.

The key innovations proposed in the SESAME architecture focus on the novel concepts of virtualising Small Cell (SC) networks by leveraging the paradigms of a multi-operator (multi-tenancy) enabling framework coupled with an edge-based, virtualised execution environment. SESAME falls in the scope of these two principles and promotes the adoption of Small Cell multitenancy, i.e., multiple network operators will be able to use the SESAME platform, each one using his own network 'slice'. Additionally, it endorses the deployment of SCs with some virtualized functions, with each Small Cell containing also a micro-server through appropriate fronthaul technology. The micro-server running the virtualised functions with the SC forms the Cloud- Enabled Small Cell (CESC). Many CESCs form the 'CESC cluster' capable to provide access to a geographical area with one or more operators. The CESC offers computing, storage and radio resources. Through virtualization, the CESC cluster can be seen as a cloud of resources which can be sliced to enable multi-tenancy. Therefore, the CESC cluster becomes a neutral host for mobile Small Cell Network Operators (SCNO) or Virtual SCNO (VSCNO) which want to share IT and network resources at the edge of the mobile network. In addition, cloud-based computation resources are provided through a virtualised execution platform. This execution platform is used to support the required VNFs that implement the different features/capabilities of the Small Cells (and eventually of the core network) and the cognitive management and Self-X operations, as well as the computing support for the mobile edge applications of the end-users.

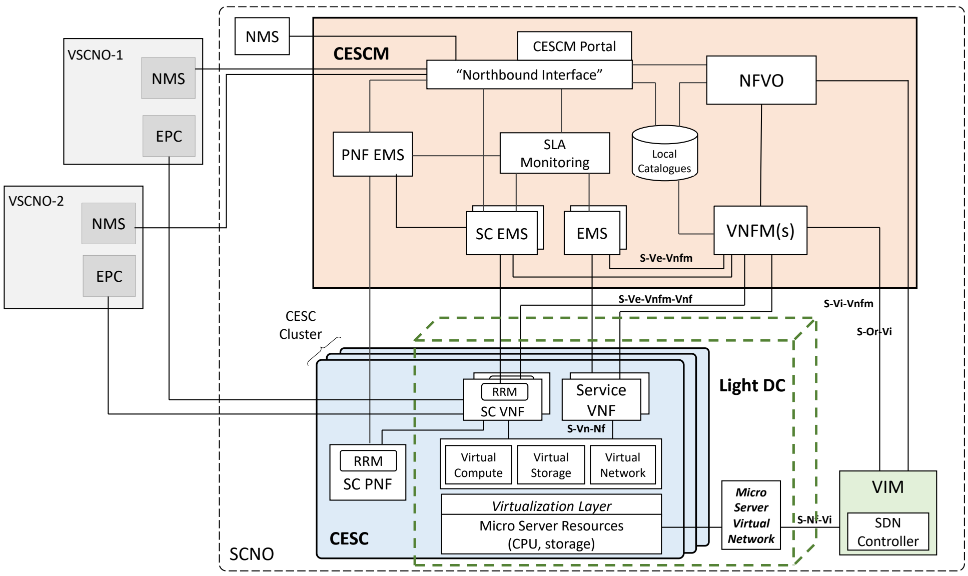



**Figure 12 - SESAME overall architecture**

The CESC clustering enables the achievement of a micro scale virtualised execution infrastructure in the form of a distributed data centre, denominated Light Data Centre (Light DC), enhancing the virtualisation capabilities and process power at the network edge. Network Services (NS) are supported by VNFs hosted in the Light DC (constituted by one or more CESC), leveraging on SDN and NFV functionalities that allow achieving an adequate level of flexibility and scalability at the cloud infrastructure edge. More specifically, VNFs are executed as Virtual Machines (VMs) inside the Light DC, which is provided with a hypervisor (based on KVM) specifically extended to support carrier grade computing and networking performance. Over the provided virtualised execution environment (Light DC), it is possible to chain different VNFs to meet a requested NS by a tenant (i.e. mobile network operator). Note that, in the context of SESAME, a NS is defined as a collection of VNFs that jointly supports data transmission between User Equipment (UE) and operators' Evolved Packet Core (EPC), with the possibility to involve one or several service VNFs in the data path. Therefore, each NS is deployed as a chain of SC VNFs and Service VNFs.

Finally, the CESC Manager (CESCM) is the central service management and orchestration component in the overall architecture figure. It integrates all the necessary network management elements, traditionally suggested in 3GPP, and the novel recommended functional blocks of NFV MANO. A single instance of CESCM is able to operate over several CESC clusters, each constituting a Light DC, through the usage of a dedicated Virtual Infrastructure Managers (VIM) per cluster. Figure 12 illustrates the overall SESAME architecture, including main components and reference points.

SESAME holds within its consortium five main T-NOVA partners (namely i2CAT, NCSRD, ATOS, ITALTEL and ZHAW). Therefore, the technology selection for the implementation of NFV Orchestrator is T-NOVA TeNOR. This decision impacts also the selection of the data model to be followed, adapted of course to the requirements of the project, as SESAME also employs PNFs in the chain.

Beyond this very important decision, T-NOVA VNFs are considered to be ported and adapted to the needs of the project, i.e. ARM based architecture for the NFVI-PoP.

## 2.3.2. Other European R&D Projects

### 2.3.2.1. VITAL

H2020 VITAL project [50] addresses the combination of Terrestrial and Satellite networks by bringing NFV into the satellite domain and by enabling SDN-based, federated resources management in hybrid SatCom-terrestrial networks. The project primarily focuses around three key application scenarios: Satellite Virtual Network Operator (SVNO) services, Satellite backhauling and hybrid telecom service delivery. The solutions developed in VITAL enable improved coverage, optimised communication resources use and better network resilience, along with improved innovation capacity and business agility for deploying communications services over combined networks.

VITAL addresses the development of a hybrid architectural framework, the required mechanisms to enable virtualization of SatCom network components, including performance optimisation and implementation of a number of virtualised functions, and the design of an SDN-enabled, federated resources management framework, embedding strategies and algorithmic solutions to provide end-to-end communication services. A high-level view of the overall architecture for SDN/NFV-enabled satellite ground segment systems is given in Figure 13, illustrating its functional building blocks and the reference points among them.

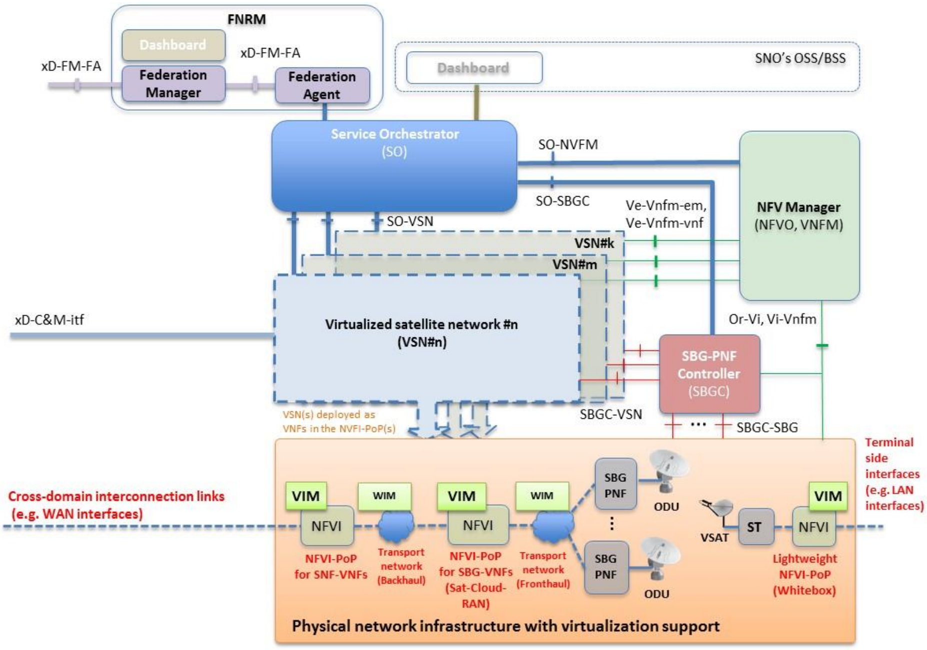



Figure 13 - VITAL overall architecture

The VITAL system architecture is composed of the following building blocks:

- **Physical network infrastructure with virtualization support**. This building block consists of the virtualization-capable physical network elements on top of which Virtualized Satellite Networks (VSNs) are deployed. This infrastructure includes the NFV Infrastructure-Point(s) of Presence (NFVI-PoP(s)), the Satellite Baseband Gateways (SBGs), the Physical Network Functions (SBG-PNFs), the Satellite Terminals (STs) and the transport network between the several NFVI-PoPs.

- **Virtualized Satellite network (VSN)**. The VSN is a satellite communications network in which most of its functions are supplied as VNFs running in one or several of the NFVI-PoPs of the physical network infrastructure. Several isolated VSNs can be deployed over the same physical network infrastructure. The non-virtualized functions of a VSN are provided through one or several SBG-PNFs, which could be dedicated to a given VSN or shared among several VSNs. The operation of each VSN could be delegated to the customer/tenant, acting as a satellite virtual network operator (SVNO). Each of the VSNs may be customized to the customer/tenant's needs, including a variety of different network services running as VNFs (e.g. PEP, VPN, etc.). A detailed description of this building block is provided in Section 5.

- **Management components.** This contains the set of functional entities needed for the provision and lifecycle management of the VSNs. In particular, VSNs can be instantiated, terminated, monitored, and modified (e.g. scaled up/down, VNFs added/removed, satellite carriers added/removed, etc.) through the following management entities:
    - **NFV Manager**. This is the entity responsible for the management of the VNFs that form part of the VSN, taking care of the instantiation, the dimensioning and the termination of the VNFs.

- **Service Orchestrator (SO)**. The role of the Service Orchestrator (SO) within the VITAL architecture is mainly to provide service composition and to provide support for the OSS/BSS functionalities independently of the nature of the service (virtualized or not).
- **Federation Network Resource Manager (FNRM)**. This element is in charge of multi-domain service orchestration. It consists of two separate components: a Federation Manager (FM) and a Federation Agent (FA). The FM hosts the logic to federate different domains and orchestrating Multi-Domain Network Services (MD-NSs). The FA is a component intended to handle the heterogeneity of the various underlying orchestrators and management entities of each domain, interfacing them with the FM. In addition to the FM and FA, a dashboard/customer portal is included as part of the FNRM to perform MD-NSs deployment, instantiation and orchestration.
- **SBG-PNF Controller (SBGC)**. The SBGC manages the pool of SBG-PNFs. Through the SBGC, the SO can request the allocation of SGB-PNFs resources (e.g. forward/return channels) for a given VSN. To that end, the SBGC is in charge of slicing the resources of the SBG-PNF so that a logically isolated portion of those resources are allocated to a particular VSN. In addition, the SBGC may provide a SDN abstraction of the allocated resources so that control and management of these resources can be integrated within the VSN.

In relation to T-NOVA, Vital segregates the responsibility of the Service Orchestration (SO) from the general Orchestration concept. In T-NOVA part of the SO functionality is also delegated to the T-NOVA Marketplace, the place where the service composition is made. In this context, VITAL re-used and simplified the T-NOVA data model, to be suitable for the focus are of the project. The project does not make use of TeNOR as it uses a stripped-down implementation for the NFV orchestration, however the implementation of NFVI-PoP is based on the guidelines and best practices as proposed by T-NOVA. In addition, some of the considered VNFs are based on the open-source versions provided by T-NOVA.

## 2.3.3. R&D projects outside EU

In this section, three US projects are briefly analysed and compared with T-NOVA.

**NSF Network Functions Virtualization with Timing Guarantees**

The NSF project "Network Functions Virtualization with Timing Guarantees" [51] investigates building a scalable NFV platform that enables latency and throughput guarantees in cloud computing setting. In particular, this project aims to leverage on real-time systems techniques to fulfil latency and throughput requirements. This project complements T-NOVA system by extending NFV to include more performance constraints.

**ChoiceNet**

ChoiceNet [52] proposes an Internet architecture where consumers can buy network services (e.g., connectivity, network functions) from providers through a marketplace. The consumers can combine different service to fulfil their needs. They can also act as resellers to other consumers. In this respect, ChoiceNet provides an architecture which is similar to T-NOVA but with more focus on traditional network layer features (e.g., routing, quality of service) and on the economic relationship between the network entities. In contrast, T-NOVA focuses on providing network functions and in-network processing.

**Network Function Virtualization Using Dynamic Reconfiguration**

The project "Network Function Virtualization Using Dynamic Reconfiguration" [53] aims to develop a new hardware-based technology that enables scalability and programmability. This project investigates the implementation of network functions using field-programmable gate array (FPGA) that can be dynamically reconfigured and customized based on the changes on the network functions' requirements. In comparison to T-NOVA, this project aims to provide hardware-based implementation of NFs, whereas T-NOVA focuses on software implementation of NFs.

## 2.4. Industry initiatives

In this section two relevant industry initiatives in the NFV space are briefly described. Also, it should be noted that D8.13, one of the final T-NOVA deliverables, will include information on relevant commercially available solutions.

### 2.4.1. AT&T ECOMP

Enhanced Control, Orchestration, Management & Policy (ECOMP) is the keystone of AT&T's Domain 2.0 (D2) Program, on which the company expects to virtualize 75 percent of the network infrastructure by 2020 [54]. A key goal of the platform is the ability to add features quickly and reduce operational costs.

The ECOMP architecture builds on the ETSI NFV architectural framework. ECOMP extends the scope of ETSI MANO by including Controller and Policy components and increases the scope of ETSI MANO's resource description to include complete meta-data for lifecycle management of the virtual environment. Another noticeable difference compared to ETSI NFV model is that Element Management Systems are not part of the ECOMP model, instead enhanced FCAPS capability have been incorporated within the MANO-like area.

In March 2016, AT&T announced the release of the ECOMP platform as a way to re-architect the network based on a new software-centric paradigm [55]. A few months later, it was announced that ECOMP would become an open source offering managed by the Linux Foundation, with the goal of making it the telecom industry's standard automation platform for managing virtual network functions and other software-centric network capabilities [56].

From a T-NOVA point of view, ECOMP may be seen as a possible approach to build an NFV platform and a different variation of ETSI NFV architectural framework. ECOMP is optimized to deliver an open management platform for defining, operating and managing a wide range of products and services based upon virtualized network and infrastructure resources and software applications, whereas T-NOVA is mainly focused on VNFaaS environments and address VNFaaS-specific requirements and challenges.

### 2.4.2. CORD

Central Office Re-architected as a Datacenter (CORD) [57] aims at bringing datacenter economics and cloud flexibility to the telco Central Office and to the entire access network, by building an open source service delivery platform that combines SDN, NFV, and elastic cloud services. ONOS, OpenStack, Docker, and XOS are components of CORD, all running on merchant silicon, white-box switches, commodity servers, and disaggregated access devices.

CORD is an independently funded software project hosted by The Linux Foundation, supported by an ecosystem that includes ON.Lab and a group of organizations such as AT&T, China Unicom, Comcast, Google, NTT, SK Telecom, Verizon, Ciena, Cisco, Fujitsu, Intel, NEC, Nokia, Radisys and Samsung.

CORD aims at not only replacing today's purpose-built hardware devices with more agile software-based counterparts, but also enable a rich collection of services, including access for residential, mobile, and enterprise customers.

T-NOVA NFVI-PoP could be materialized as part of the new Central Office, as envisioned by CORD. Amongst the multiple domains of use of the CORD platform and the potential set of innovative services that may be enabled for residential, mobile, and enterprise network customers, T-NOVA VNFaaS concept, with a view to deploying virtualized network functions to be consumed "as-a-Service" by customers appears to be an obvious candidate. Since CORD is especially oriented towards the operators' network edge, the edge NFVI-PoP, as previously described, seems to fit particularly well this vision.

# 3. DEPLOYMENT GUIDELINES

This section addresses issues related to T-NOVA deployment. The first section is focused on the deployment of T-NOVA over an existing commercial grade VIM, namely Helion OpenStack Enterprise Edition 2.1.2, HPE's commercial OpenStack release, with a view to assessing the flexibility of the T-NOVA platform to be integrated in "real-world" technical environments, different from the purpose-built T-NOVA pilot infrastructures. Section 3.2 explores the deployment of T-NOVA in environments that are extensions or variations of the basic T-NOVA model. Section 3.3 is focused on the interoperation with legacy connectivity services and the impact of NFVI-PoP location. Finally section 3.4 outlines a possible roadmap for T-NOVA incremental deployment in service provider infrastructures, again taking into account the requirement of interoperation with legacy connectivity service models.

## 3.1. Guidelines for deployment over existing NFV infrastructures

During the last 12 months of T-NOVA, the consortium and the European Commission accepted the proposal from HPE to run an additional pilot, fully complementary and not interfering with the main pilot described in deliverables D4.52 and D7.2. This new pilot has been setup and run on a small scale local infrastructure lab at HPE Italy premises. Its purpose was to obtain additional learnings to the ones coming from the main pilot, specifically to experiment the deployment of T-NOVA software components into a different infrastructure, or in other words evaluating T-NOVA deployment on a proprietary (commercial) VIM and NFVI. An additional related rationale was to make a better analysis of T-NOVA usability in "real world" scenarios, from the side of an adopter external to the project consortium, not just on a theoretical point of view, but reaching some conclusions through hands on experience. In other words, the goal was to walk in the footsteps of a user willing to utilize T-NOVA, starting greenfield from the open source code released on GitHub.

The chosen target VIM was HPE Helion OpenStack Enterprise Edition 2.1.2, HPE's commercial OpenStack release, based on the Kilo version of OpenStack, like the main T-NOVA pilot. This way, the alignment and comparability of tests was better granted, avoiding to introduce issues only depending upon the OpenStack version differences.

The outcomes expected by this extra pilot were an assessment of the potential integration effort, and main related issues, of T-NOVA with existing infrastructures different from the one used in the project, identifying possible gaps to fill in perspective of post-project evolutions and employment of the T-NOVA software. As additional goal, the pilot provided quite valuable indications for HPE own exploitation planning.

The HPE pilot encompassed four different sequential phases:

- Hardware infrastructure setup and configuration
- Helion OpenStack deployment
- T-NOVA components deployment

- Testing based on T-NOVA use cases

Each phase is briefly reported below. Then, a recap of the acquired learnings will be presented.

**Hardware infrastructure setup**

The pilot was run on a hardware testbed so composed:

- 1x **C3000 HPE ProLiant Blade Enclosure** with:
  - **6x HPE ProLiant BL460c servers**
    - 2x quad-core Intel Xeon E5540 CPUs
    - 64/48 GB Ram (controllers/compute nodes)
    - 2x 300 GB local disks (15K rpm)
    - 2x 10 Gbps physical NICs (up to 8 logical)
  - Flex 10 Virtual Connect modules
    - 2x 10 Gbps Ethernet uplinks
    - 2x 8 Gbps Fiber Channel uplinks
- 1x HPE 5120 Network switch
- 2x Brocade 8Gb SAN Switches
- 1x HPE MSA 2324 Storage Array
  - 3.6 TB total disk space

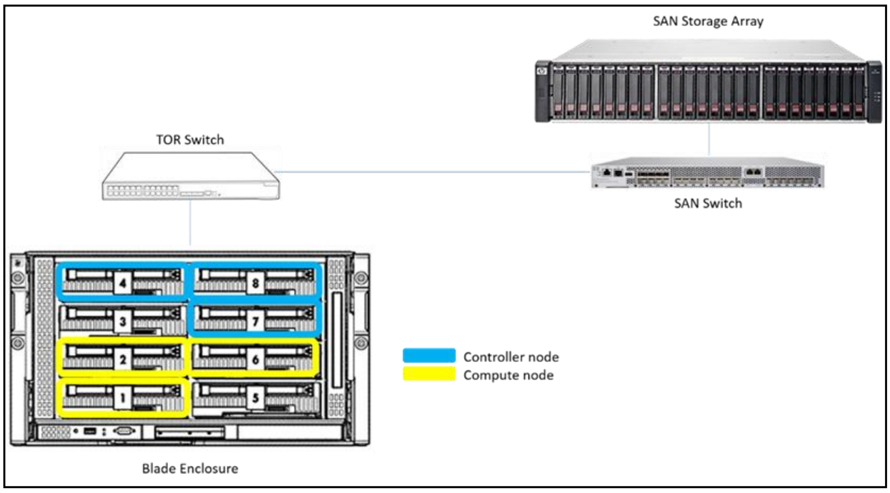



**Figure 14 - HPE pilot hardware testbed**

Helion OpenStack 2.1.2 has tighter hardware requirements than the Community OpenStack, so it has been necessary to use a solution with a control plane composed by a three-node cluster. As depicted in Figure 14, the Helion cluster was also composed of three compute nodes, each one equipped with 48 GB of RAM.

Storage was provided by a SAN Storage Array. To each node, different types of LUNs (Logical Unit Number) were exposed:

- Controller nodes:
  - 2 x 300GB: used for Swift object storage

    - 1 x 600GB: used for Cinder block storage
- Compute nodes:
    - 1 x 500GB: reserved for Nova Compute

The testbed was configured for external network connectivity through a top of rack switch (model HPE 5120).

**Helion Openstack deployment**

Among the other added values respect to vanilla (community) OpenStack, Helion OpenStack provides to the cloud administrator several tools for managing the cloud infrastructure deployment. Following the paradigm of Infrastructure as Code, the hardware characteristics and interconnections need to be described in YAML (Yet Another Markup Language) template files, so that the cloud installation is automated through the execution of Ansible playbooks. Both these playbooks and YAML files are versioned within a version control system (GIT), therefore it is quite simple to perform the cloud installation, maintenance and configuration upgrades.

**T-NOVA components deployment**

Once the Helion OpenStack VIM has been correctly installed and configured, the next step was installation of the T-NOVA components on top of it. A selected list of components was deployed to the testbed, in order to allow the execution of a full end-to-end functional test without introducing excessive and unneeded overhead, not providing significant incremental value to the pilot results. The chosen components were:

- Network Function Store (NFS)
- Marketplace
- TeNOR (with Gatekeeper as the authentication tool)

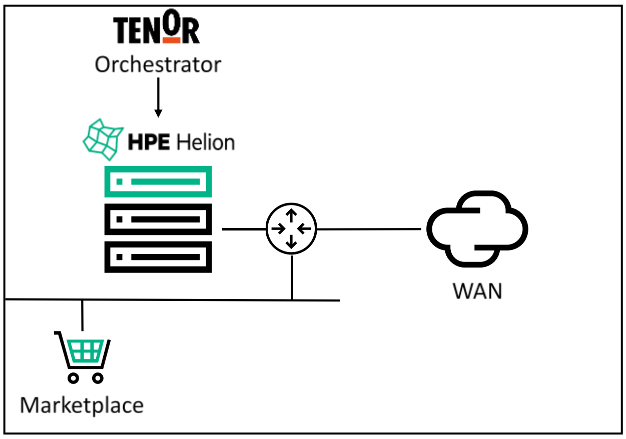



**Figure 15 - T-NOVA deployment on HPE pilot**

**Testing**

With the T-NOVA deployment completed, a summary of the use cases UC1, UC2 and UC3 was executed, going through the following steps:

- A Function Provider uploads a VNF image to the NFS;

- The Function Provider enters into the Marketplace dashboard, and defines the VNFD correspondent to the previously uploaded VNF image, performing the VNF addition to the catalogue;
- The Service Provider enters into to the Marketplace dashboard, and defines a new Network Service using the VNF previously added to the catalogue;
- A customer accesses the Marketplace and purchases an instance of the previously defined Network Service;
- The TeNOR Orchestrator instantiates the Network Service as requested, and the correct execution of the service is verified.

The test was performed using the PxaaS VNF developed by T-NOVA partners. HPE is planning to continue developing the last version of the pilot after T-NOVA conclusion, by upgrading to the most recent Helion releases, and performing even more complex tests (VNF scaling, multi-VNF network service, multi-PoP instantiation).

**Lessons learnt**

As previously stated, the main scope of this pilot was to test the T-NOVA framework against a commercial enterprise-grade VIM, and detect T-NOVA strengths and weaknesses in this utilization scenario.

Overall, the main strength point of the T-NOVA framework turned out to be its modular architecture, where each component has several interfaces. Thanks to this approach, it is possible for a T-NOVA adopter to integrate even only a subset of T-NOVA components, as done in this pilot, getting them to coexist with existing modules in ETSI-compliant environments. This can surely help to increase the initial pervasiveness of T-NOVA.

On the other hand, the pilot allowed to detect some attention points, which can occur when T-NOVA is deployed on a commercial grade VIM like Helion OpenStack. Compared to the community software, commercial solutions are ruggedized especially on the performance assurance, security and reliability aspects, sometimes leading to different design choices, which can in some points hinder the deployment of T-NOVA. The pilot highlighted in particular some difficulty with the monitoring framework and the SDN controller deployment. In more detail:

- **Monitoring**: the T-NOVA monitoring framework is depending upon Ceilometer. The future positioning of Ceilometer with respects to the OpenStack evolution is still not so clear. This is due to the fact that Ceilometer suffers serious performance issues and it is not very reliable in terms of scalability, as also outlined in the deliverable D4.41 [58]. For this reason, Ceilometer support is getting dropped out from several OpenStack commercial distributions (Helion OpenStack among them), to be replaced by proprietary or more robust open solutions (Monasca). From the experience done in the HPE pilot, there is significant risk that when using T-NOVA on a commercial VIM, the ability of fully exploiting its monitoring functionality can be hindered or partially compromised.
- **SDN controller**: Another potential limitation to date is the use of an open-source SDN controller (OpenDaylight) to control both intra- and inter- PoP

networking connectivity. The majority of commercial VIMs relies upon their own implementation of SDN controllers, and does not guarantee a full support to open source APIs. This was exactly the case with Helion OpenStack: albeit HPE is platinum member and top contributor to the OpenDaylight community, OpenDaylight is not officially supported by the Helion release employed in the pilot (mostly for performance reasons), which in general does not support 3rd party Neutron plug-ins. This limitation will probably be overcome in the next releases of Helion, but this issue could still persist for some time until OpenDaylight really achieves a commercial and telco-grade readiness level. This problem can bring functional shortcomings to the current T-NOVA version.

Lesser issues had to be solved to get TeNOR working with Helion OpenStack. Some configurations had to be modified, since, using the out-of-the-box T-NOVA orchestrator, the platform was actually not able to deploy VNFs. To deploy VNF instances, in fact, TeNOR requires some particular authorizations that in Helion OpenStack cannot be granted to non-admin tenants. Again, this is the typical hindrance you can come across when using a commercial grade VIM. In particular, the incurred roadblocks concerned:

- Definition of new OpenStack flavours: in OpenStack terminology, a flavour is the set of virtual hardware resources (virtual CPUs, disk and RAM amount) reserved to a given provisioned VM instance. This is an essential feature to guarantee the ability of provisioning different VNF flavours, since each flavour defined in a VNF Descriptor should be mapped to a corresponding OpenStack flavour. In Helion OpenStack this feature is disabled by default for security reasons, and it needed to be selectively enabled, at least for the tenant directly managed by TeNOR.
- Security settings: T-NOVA code uses deprecated APIs to modify the security group of the tenant hosting TeNOR. Helion OpenStack does not implement the deprecated APIs, hence these settings had to be manually crafted using the Helion dashboard, by creating specific rules for opening TCP/UDP ports. The usage of deprecated APIs can clearly bring similar issues with any commercial VIM, thus it is recommended to avoid it in the future.
- Enabling external repository for images: this is another feature by default disabled by Helion OpenStack and other commercial VIMs. Since in T-NOVA the image repository is an external component respect to the VIM, this feature needed to be turned on.

From the perspective of “fresh” user experience, the pilot was useful to better detect some imprecision or misalignment respect to code release in the publicly available documentation. In general, some useful suggestions came also to improve the description of T-NOVA global architecture and interdependencies among its components. Clearly, it has been a work-in-progress, since the pilot went on in parallel with the documentation refinement, and the status of documentation was well known to not be final. Nonetheless, this low hanging fruit represented an additional outcome provided by the additional pilot, which allowed to further improve the overall quality of the final T-NOVA release.

## 3.2. T-NOVA deployment use cases and scenarios for adoption

Early in the T-NOVA project, possible deployment cases of T-NOVA infrastructure were discussed. Thanks to the flexibility of the T-NOVA architecture to accommodate multiple variants or extensions, the concept of VNFaaS, as conceived by T-NOVA, can be materialized in different technological and business environments. Those cases are revised in this section to re-evaluate our initial view.

### 3.2.1. T-NOVA interfacing existing OSS/BSS

This is the scenario (Figure 16) where T-NOVA is deployed in various locations (NFVI-PoPs) and the transport network required for the interconnections between the PoPs needed all management and reservation requests to be realized through its OSS/BSS system. In this case, there is no direct interface of NFVO to the Network Management modules and all communication is performed via OSS/BSS.

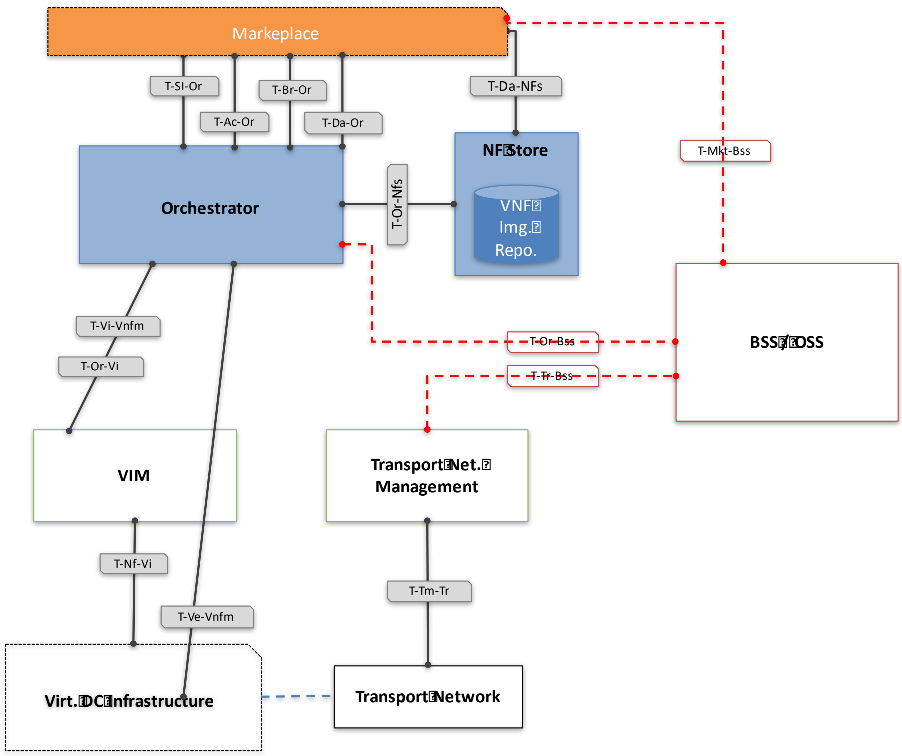



**Figure 16 - OSS/BSS deployment scenario**

The added value presented by this deployment is the fast deployment of NFVI-PoPs in certain locations and fast exploitation of existing, legacy network infrastructure for transport network connectivity.

### 3.2.2. Multiple NFVI-PoP providers

This scenario (Figure 17) is a very possible deployment scenario for NFV and in this extent for T-NOVA. As the CAPEX/OPEX costs for multiple PoPs across the network reach of a Network Operator might be too high, Network Operators will seek for opportunities of leasing infrastructures from public cloud/infrastructure providers. In this case T-NOVA will be required to operate over multiple PoP providers. T-NOVA since specifications epoch provides support for multiple PoP deployment, exploiting the hybrid cloud deployment (every PoP has its own consolidated management – VIM). Therefore, there is no cascading model use (i.e Openstack is deployed using cascaded model) or multi-regional deployment.

This scenario assumes that the communication with the BSS/OSS system of the Infrastructure Provider is needed to give access to the VIM for orchestration of the resources via the NFVO of T-NOVA. It is presumed that both share the same implementation for VIM (or at least the involved interfaces). Of course, if required the modular implementation of TeNOR may enable and support different implementations of the T-Vi-Vnfm / T-Or-Vi interfaces according to the technology used at the other VIMs. However, this is out of the scope of the project.

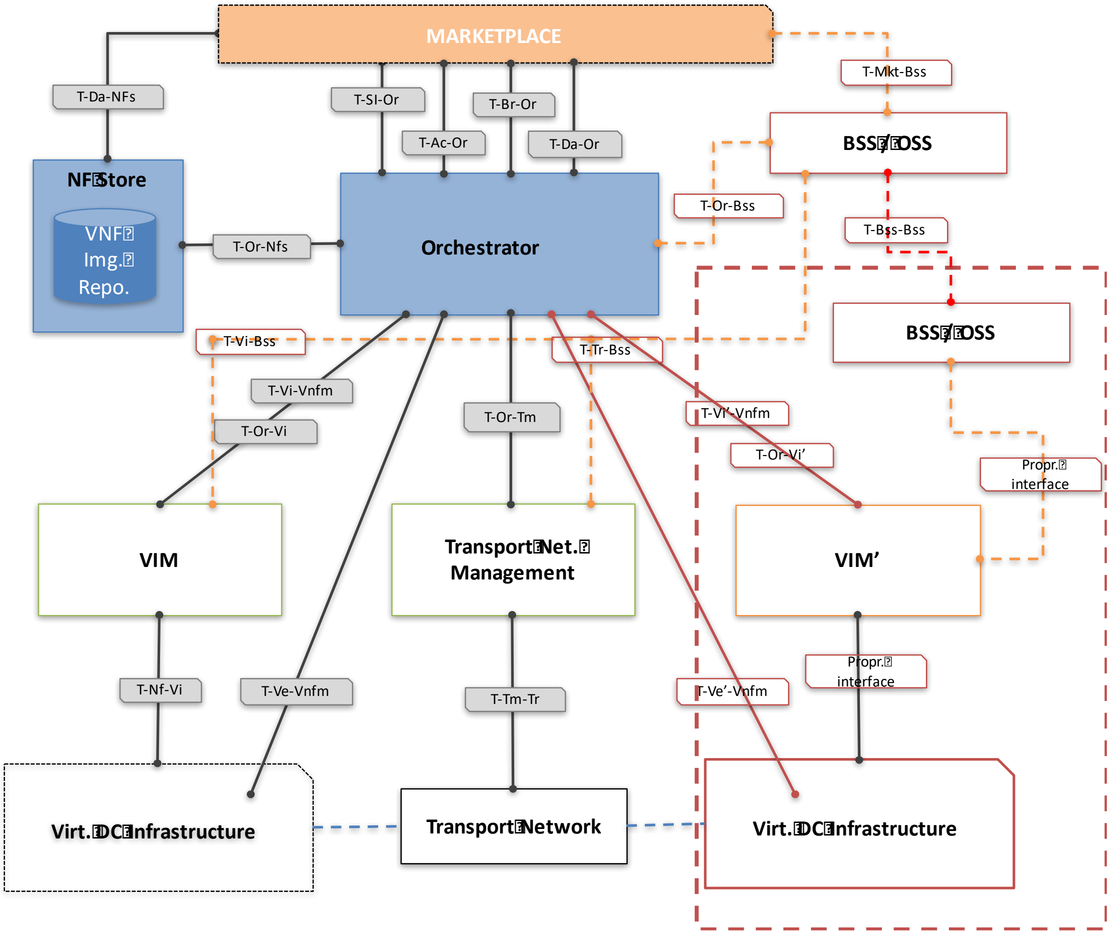



**Figure 17 - Multiple PoP Providers**

The added value by this use case is the exploitation of resources available from other Infrastructure Providers.

## 3.2.3. Multiple T-NOVA SPs (Orchestrator Federation)

This scenario (Figure 18), is applicable in the case that multiple T-NOVA enabled SPs (also capable of playing the broker role) are federated in order to provide end-to-end

Network Services. To enable the service provider federation through the T-NOVA architecture, it is envisaged that a new interface shall be added between the corresponding orchestrator components namely T-Or-Or. This interface will enable east-west communication communicating relevant information required for the provision of end-to-end NS. The technical challenges and implementation details for these interfaces are out of the scope of T-NOVA.

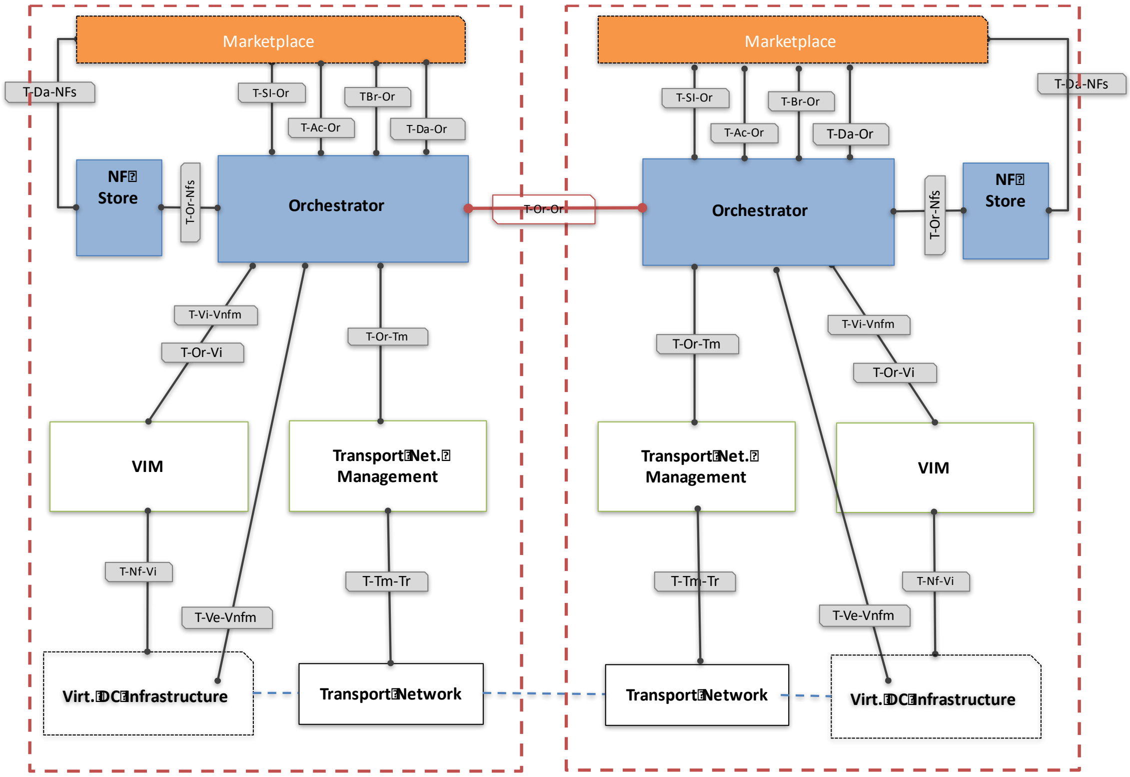



**Figure 18 - T-NOVA enabled SPs Orchestration**

The added value of this scenario is that the service offerings of each SP would be increased through the addition of new services that can be deployed through the functions of the other service provider in a transparent way

## 3.3. NFVI-PoP location and WAN connectivity: edge vs. core

Very often, the potential VNFaaS customers are already customers of a previous connectivity service, such as L2/L3 VPNs (very likely to be the case with enterprise customers). The widespread adoption of VNFaaS in the short to medium term requires the integration with standard enterprise WAN connectivity service models. The complexity of this integration largely depends on the number and location of NFVI-PoPs. Two basic scenarios can be defined about NFVI-PoP location:

- Edge NFVI-PoP: In this case, the NFVI-PoP is located at the edge of the service provider network, thus close to the customer attachment point, traditionally the so-called "Central Office" (roughly in line with CORD vision – see section 2.4.2), which may also coincide with the Provider Edge (PE). The VNF physical location is close to the respective logical location. Locating NFVI resources at the network edge helps reduce packet latency, which is particularly important for network functions requiring minimal or predictable latency, for reasons of quality of experience. On the other hand, utilization of network bandwidth is optimal thanks to local processing, thus avoiding the need to send traffic to a centralized data center and back to the originating point at the edge. The price to pay is usually the need to disperse NFV infrastructure across a potentially high number of physical locations.

- Centralized NFVI-PoP: In this case, VNF physical location is supposed to be distant from the respective logical location. The insertion of a VNF in the data path usually requires traffic to be redirected in the WAN and for this reason is usually not transparent from the point of view of an existing WAN connectivity service. A remote NFVI-PoP typically serves multiple PEs. Compared to the distributed Edge NFVI-PoP model, this approach requires fewer NFVI-PoPs and enables economies of scale, but the price to pay is greater traffic steering complexity, higher WAN bandwidth consumption and higher end-to-end latency.

The pros and cons of the edge and core deployments depend on several aspects, namely whether the function acts on control plane or data plane, and in the latter case, whether it plays and endpoint or a transit function. Control plane VNFs are supposed to be accessed simply by the respective IP address, therefore from the network point of view, standard IP routing is all that is required to reach a control plane VNF. On the contrary, data plane VNFs are supposed to be inserted in the data path, which means that the deployment of a new data plane VNF is non-trivial from a network point of view, as it usually requires data traffic to be redirected to the respective NFVI-PoP.

As mentioned above, two types of data plane VNFs should be distinguished: Endpoint VNFs, i.e. VNFs that are TCP/UDP endpoints, accessible to end users through a public or private IP address, e.g. Web proxy; Transit VNFs, i.e. VNFs that are inserted inline in the customer data path, i.e. are neither the source nor the destination of the data path, e.g. firewall, packet inspection, transcoding.

Transit VNFs represent the most challenging case from a network point of view because inserting a transit VNF when an existing connectivity service is in place usually requires traffic to be redirected "on the fly" – obviously, the more distant the NFVI-PoP is, the more challenging is this traffic redirection. For this reason, transit VNFs should be deployed at the network edge.

## 3.4. Roadmap for incremental deployment

Incremental deployability is usually a key requirement to guarantee the successful adoption of an emerging technology. Generally speaking, a technology is incrementally deployable when tangible benefits can be extracted from an initial small-scale deployment, without a massive infrastructure upgrade. In particular, VNFaaS, as envisaged by T-NOVA, requires incremental deployability to enable initial technology adoption and a progressively widespread adoption by the relevant stakeholders.

Although multiple scenarios can be described about the practical deployment of T-NOVA VNFaaS, in many cases it will likely be offered as an add-on or enhancement to a previously existing connectivity service. For this reason, seamless coexistence with legacy connectivity service models is required. Taking into account that enterprise customers will be the primary target of the service, integration with L2/L3 IP/MPLS

VPNs (which still represent the foundation of most enterprise networks today) should be seen as a key requirement in the short/medium term. In the interest of facilitating incremental deployment, transition to SDN-enabled WAN infrastructure should not be a requirement for the initial deployment of T-NOVA technology, however likely it may be in the long run.

An **incremental 3-phase approach** for VNFaaS deployment is proposed below. For illustrative purposes, a simple enterprise network, based on a L2/L3 VPN variant, represented in Figure 19, is used as starting point.

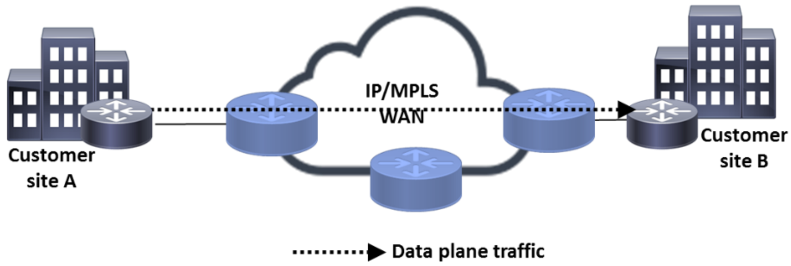



**Figure 19 - VNFaaS incremental deployment: starting point**

### 3.4.1. Phase 1 – Fully separated control

Roughly speaking, phase 1 corresponds to the deployment of a T-NOVA VNF service, supposed to be a transit VNF (e.g. traffic classifier) following the request made by the user through the marketplace portal. In Figure 20, the new infrastructure corresponds to the shaded area. It should be noted that the remaining infrastructure, including the IP/MPLS VPN, is inherited from Figure 19 and ideally should not be impacted in any way. The only common element between the legacy infrastructure and the new infrastructure is the WICM node, controlled by the WICM, which is inserted in the data path to perform traffic redirection if/when needed.

Ideally, to minimize the impact of traffic redirection, the NFVI-PoP should be located at the network edge, close to the customer. This suggests that NFVI-PoPs will tend to be disseminated at the network edge, starting in a limited number of selected areas, supposed to have the greatest potential to become VNFaaS early adopters, and progressively expanding as the demand grows.

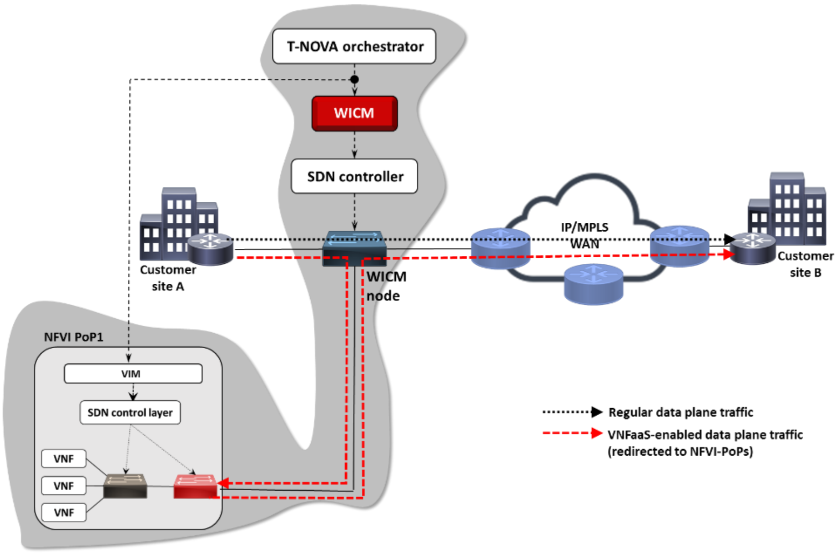



**Figure 20 - VNFaaS deployment phase 1: WAN / VNFaaS full separation**

The existing connectivity services (L2/L3 VPN) are not supposed to be affected in any way by the deployment of VNFaaS. This means that although connectivity services and VNFaaS could be offered to customers as an integrated service package, internally they are still handled as separate services, managed by separate OSS/BSS tools. The NFV orchestrator only cares about the management of the NFV infrastructure and the VNF lifecycle.

The WICM node (under the control of the WICM) is supposed to be an SDN switch in charge of redirecting data plane traffic required by the subscription/unsubscription of VNF services, in a way that is totally transparent to any existing connectivity service (e.g. L2/L3 VPN) that might be already in place.

By avoiding any impact on existing legacy connectivity services, incremental deployability is guaranteed. This solution enables a progressive deployment of the service and minimizes the impacts on pre-existing connectivity services but suffers from the fact that the integration of WAN and VNFaaS service components is done to a very limited extent only, based on separate management tools and mechanisms.

## 3.4.2. Phase 2 – Integration of WAN connectivity and VNFaaS

In order to overcome the shortcomings of the previous scenario, integrated management of WAN connectivity and VNFaaS is put in place. The WICM can play a pivotal role by controlling not only special-purpose WICM nodes, but also PE nodes, where the logic of the WAN connectivity service usually resides. The WICM is able to modify the configuration of PE routers if traffic steering rules have to be modified, basically acting as an OSS system. However, IP/MPLS and NFVI still represent two technologically different domains, the WICM being the only common element.

Since the PE routers can be controlled by the WICM, it is now possible to deploy NFVI-PoPs wherever they make more sense, so the previous constraint that NFVI-PoP should be located at the network edge, as close to the VNF logical location as possible, is no longer strictly required. To illustrate this point, in Figure 21 a second VNF is deployed in a second NFVI-PoP (NFVI-PoP 2), which requires traffic steering in the WAN to be adapted.

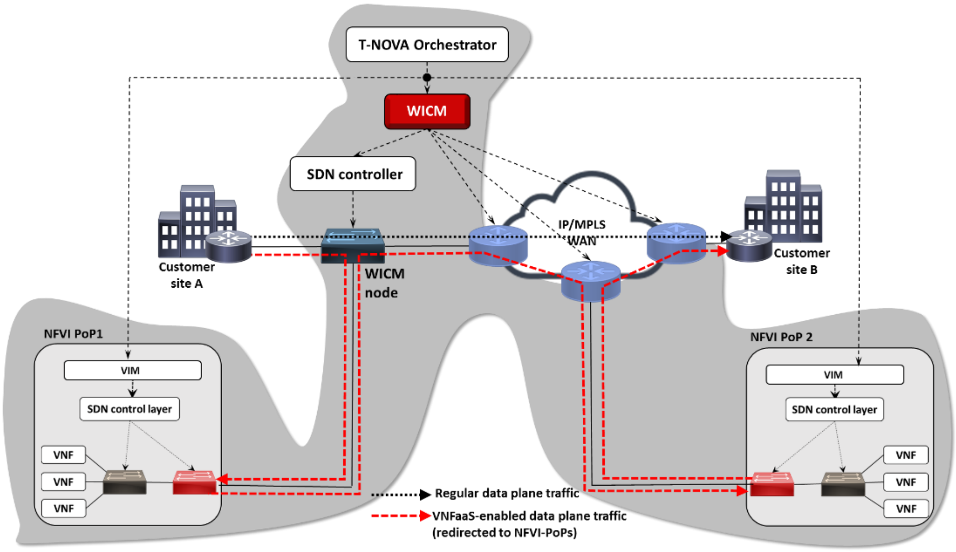



**Figure 21 - VNFaaS deployment phase 2: WAN / VNFaaS limited integration**

## 3.4.3. Phase 3 – SDN-based WAN

Phase 3 is based on the assumption that in the long run it will be possible to manage and control WAN connectivity and VNFs in an integrated way, thanks to the widespread deployment of SDN technology in the WAN to enable automation and quick responsiveness to subscription/unsubscription of services. This implies the progressive replacement of traditional WAN connectivity services to new service models enabled by SDN.

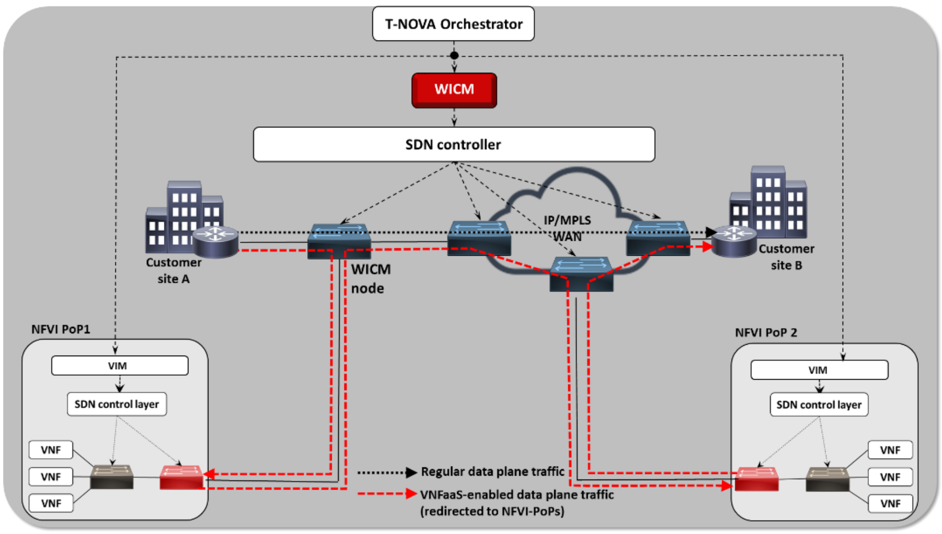



**Figure 22 - VNFaaS deployment phase 3: SDN-based WAN**

# 4. OVERALL EVALUATION OF RESULTS AND LESSONS LEARNT

This section is focused on the evaluation of results. Section 4.1 summarizes the results of T-NOVA system validation, conducted in Task 7.2, based on the predefined system use cases, comprising the complete service lifecycles and considering the requirements previously defined in the initial T-NOVA deliverables. Section 4.2 lists of main lessons learnt in several domains of the project activities.

## 4.1. Evaluation of results

This section starts with a comparison of the T-NOVA system requirements with the validation results.

Deliverable D2.1 [59] elaborates on the requirements that arose from the system use cases that have been documented in the same deliverable (D2.1 [59], Section 5). These requirements are formally listed in the table of (D2.1 [59], Appendix A). The following list summarises the achievements based on the originally specified requirements:

- **NFV service request.** T-NOVA provides the mechanisms to simplify the creation and reconfiguration of VNFs for customers. In detail, customers use templates provided in the Function Store relying on VNFs advertised by the SPs. Resource constraints (i.e., connectivity / bandwidth and compute demands) will be resolved by T-NOVA.
- **NFV service mapping.** The T-NOVA service mapping comprises multiple algorithms with different objectives to map the virtual topology as requested by the customer to the substrate networks. Common objectives used in T-NOVA are service cost minimization and revenue maximization for the customer and the provider, respectively.
- **NFV service deployment.** Following NFV service mapping, the assigned computing, network and storage resources are prepared for the deployment of the NFV service. This step comprises, for example, the installation of packet forwarding entries.
- **NFV service scaling.** T-NOVA provides scaling mechanisms to cope with changing service demands. In detail, scale-out acquires additional resources and sets up new instances while scale-in consolidates unused instances and releases resources.
- **Resource discovery.** T-NOVA collects up-to-date information about the network topology, the bandwidth utilisation as well as the utilisation of the computing and storage resources across the network infrastructure to allow efficient resource mapping.
- **Resource isolation.** T-NOVA relies on the isolation methods (e.g., CPU and traffic scheduling) provided by the hypervisors of the OpenStack-based compute nodes.
- **Resource efficiency.** Dynamic changing computational demands of VNFs raise the need for resource reallocation and optimization to be able to use resources

efficiently. Hence, VNF templates are associated with workload profiles that allow demand estimations.

- **Resource monitoring.** The implemented monitoring considers resource consumption per VNF as well as the infrastructure utilization with alert mechanisms for overutilization states.
- **SLA monitoring.** T-NOVA keeps track of service metrics and the SLA requirements. Any violations in SLAs are promptly reported to the Marketplace dashboard.
- **Billing.** T-NOVA takes into account different types of SLAs and billing models to find a price. The generated revenue is then shared among the involved function providers.
- **Secure communication and Broker authentication.** Common authentication and encryption methods are used in T-NOVA to ensure secure communication between the different parties.

Based on the predefined system use cases (that comprise complete service lifecycles) and taking into account all of the aforementioned requirements, procedures for validation were developed (D7.2 [60], Section 4). A multi-PoP pilot deployment was setup to conduct the testing procedures in the final year of the project. The results presented in (D7.2 [60]) substantiate the claim that all originally formulated requirements, according to (D2.1 [59]), were met.

Beyond the coverage of system requirements, a quantitative evaluation of the different system components has taken place and detailed results are available in (D7.2 [60], Section 4). One of the main metrics in the system-level evaluation is run time. The results allow the following classification:

- Interaction with the system, monitoring, processing of requests and updates require typically a wait cycle in the magnitude of 100s of milliseconds.
- Instantiation of new services, including VM booting and configuration is finished after 1 to 3 minutes.
- Advertisement and uploading of images requires, for example with a 2GB image, 2 minutes.

Finally, a long list of demonstrations made visible the practical value of the T-NOVA Proof-of-Concept (D7.2 [60], Section 5). Those demonstrations allowed potential users to gain experience in the future use of network functions as a service over virtualized infrastructures.

## 4.2. Lessons learnt

Undoubtedly, technologies such as SDN and NFV and service models such as VNFaaS represent today major catalysts towards industry transformation and innovation. T-NOVA has accomplished a number of relevant results in these areas. Lessons learnt in the course of these three years can be useful to other projects working in the same or related areas. Thus, this section provides a summary of the main findings and lessons learnt. It should be noted that lessons learnt from the deployment of T-NOVA platform over a commercial grade VIM (HPE Helion OpenStack), already reported in section 3.1, are not repeated here.

**Table 3 - T-NOVA lessons learnt**

| Topic | Lessons Learnt | Responsible WP and Deliverables including Technical Details |
|---|---|---|
| NFV Architecture | We identified that certain complex VNF would require refined granularity in their lifecycle control than the generic VNFM would provide. Therefore the Orchestrator design is modular and included the support for VNF specific VNFMs. This capability is also expressed in the VNF Descriptors. In addition, the NFV Architecture supported by T-NOVA identified the need for an additional element, the WAN Infrastructure Connectivity Manager, responsible for managing and controlling the traffic steering towards the PoPs from the end-users and also between PoPs.<br>The use of the middleware API, allowed T-NOVA to decouple the integration of VNFs management interface from TeNOR (similar to the abstraction layer used by SDN controllers to hide specific driver implementation from the nourthbound interface). This approach was validated by using different protocols to manage VNFs transparently from TeNORs point of view. Moreover, the VNF developers work to integrate their VNFs in T-NOVA was significantly reduced by only having to model the VNFs management interface in the descriptors. | WP2, WP5 (D2.22, D5.2) |
| NFV MANO | ETSI's MANO Framework served the project well, in terms of definition of concepts and terms used, but there are still many details that require further and deeper definitions. For example, in the VNF lifecycle, already addressed above, there are still some relevant loose ends to be solved. The frontier between the NFV MANO and the classical OSSs/BSSs is also not clear, with functions clearly crossing (system) borders, thus making difficult a coherent implementation of such functions. | WP3 (D3.1, D3.2) |
| NFV Resource Orchestration | The project has chosen to base TeNOR's, T-NOVA's open-sourced NFV Orchestrator, architecture in micro-services. We believe this has been the right choice, making available different and independently deployed services in smaller components than usual. But having to deal with many URLs and ports introduces a barrier to learn and use TeNOR that can be avoided/circumvented by the adoption of containers (e.g., Docker), which makes micro-services easier to deploy. Also, bear in mind that micro-services are not just a technological choice: by providing smaller micro-services, specially when associated to containers, we increase the number of people that are able to contribute, since each micro-service can be done in a different technology stack than any other. | WP3 (D3.1, D3.4) |
| Descriptors-Catalogues | In order to build the Os-Ma interface according to ETSI terminology, we concluded that a unique Network Service Descriptor is preferable to be shared between customer facing catalogues and orchestration catalogues. In this way, business service offerings including SLAs and pricing can be mapped directly to service deployment in the interest on business agility and operability. Each layer will be using descriptor fields relevant | WP3, WP5, WP6 (D2,32, D2.42, D3.42,D5.2, D6.1) |

| | | |
|---|---|---|
| | for their operations. Currently, Cloud applications templates employ a mature model for the automation of lifecycle management. One example are the intrinsic functions such as the "get attribute" which enables the use of specific instance deployment information to configure applications, e.g. IP address. By using this approach, common in cloud environments, we were able to use descriptors to programmatically manage the VNFs lifecycle. | |
| Infrastructure Repository | New versions of the OpenStack result in challenges for the repository due to changes in the NOVA DB schema. A better approach would be to use the NOVA API to retrieve the infrastructure related information, however this is currently not possible as full access to infrastructure related information is not provided by the API. Need to extend NOVA API in future releases to address current limitations. OCCI standard was used to implement REST API for repository. However OCCI creates a flat schema which does not allow the relationships in the resources to be expressed in a hierarchical manner. Required changes to OCCI implementation to address this limitation (output was compliant, implementation not fully compliant. OCCI standard needs to be updated to support hierarchical expression of resource relationships. | WP3 (D3.2) |
| Monitoring | Ceilometer has proven rather unstable and eventually its use was abandoned during the last phases of the project. At least one internal DNS service per PoP turned out to be necessary, so that VNFs could resolve the IP of the monitoring endpoint, in order to dispatch the metrics. It is preferable (and more scalable) to produce alarms at VIM level, rather than centrally at the Orchestrator. Also, in large deployments with tens of NFVI-PoPs and potentially thousands of VNFs, only alarms/events should be communicated, for the sake of scalability. VNF anomaly detection (via statistical processing and potentially machine learning) turned out to be a very interesting and timely research topic | WP4 (D4.32) |
| Marketplace - Auctioning | Only few brokerage providers enable quality assurance capabilities (New Relic, Tapp, CloudKick, Rightscale, and Kaavo, Cloud Exchange Platform). With the exception of one (New Relic), all of those offerings focus on IaaS, which happens to be the most commoditized category of cloud services today. Coverage of optimization capabilities is even sparser. Moreover, only one brokerage provider addressing this type of capability (RightScale), which also happens to focus on only one type of cloud service (IaaS). The T-NOVA Brokerage module is the only brokering platform working with the NFVaaS concept achieving in this way benefits for the SP. More specifically, the T-NOVA SP has the ability to trade among a variety of FP's and receive the best available NFV for his service by taking into account the Infrastructure cost and the expected performance (SLA) of the NFV. The extension of Auctioning in a multi-Service Provider Network will make more sense as the constraint roles used inside T-NOVA do not provide the full capabilities of auctioning for VNFaaS concept. | WP6 (D6.2) |
| Marketplace - SLA Assurance | Based on the SLAs analysis done in T-NOVA, we conclude that Service Assurance in NFV is something complex due to different source that can affect the service performance, namely: | WP2, WP6 (D2.42, D6.4) |

| | | |
|---|---|---|
| | - Orchestration operation<br>- Cloud infrastructure<br>- Network<br>- VNF design<br>- Quality of software<br>Usually, in telecommunication services, SLAs can be seen as the minimum service acceptance level a customer would agree to be delivered by a communication service provider, though they are often vague, not end-to-end and unknown to the network, basically based only on coverage and connectivity. But SLAs in NFV requires a combination of network functions and cloud. This takes us to a problem of responsibility resolution between the actors involved in the value chain, e.g. service provider, cloud provider or the different VNF developers. A VNF will behave differently depending on the amount of resources allocated for its deployment (according to what will be described in the VNFD), this is why SLA should be taken as one of the key inputs to be considered for the service mapping (placement) problem. | |
| NF Store | The choice to realize the Network Function Store like a web service running on a tomcat web server might seems not the best but the observed performances, also in this implementation that use only one server instance, are good related to the required use. | |
| Workload Characterisation and KPIs | Workload characterisation is manually intensive and does not scale. We concluded that an automated approach is required to achieve the level of scalability to support wide scale VNF deployments. In addition we identified that allocation of infrastructure resources needs to optimised in the context of the KPIs e.g. latency, throughput that needs to be achieve for a given workload deployment | WP4<br>(D4.1) |
| NFVI-PoP Deployment | 1. From a hardware perspective commercial off the shelf (COTS) X86 servers have been used in the implementation of the testbed. However, the use of additional hardware platforms to provide specialized acceleration capabilities namely FPGA was also investigated and may form part of the final testbed implementation.<br>2. The absolute minimum requirement for the T-NOVA IVM layer deployment is four nodes. This minimum setup is tailored to development purposes. Setting up a T-NOVA deployment for testing and experimentation purposes requires a minimum of eight nodes, while a setup for pilot or demonstration requires a minimum of fourteen nodes.<br>3. OpenStack and OpenDaylight have been integrated successfully through the ML2 plugin for those specific releases:<br>- OpenStack Juno and OpenDaylight Helium SR2<br>- OpenStack Kilo and OpenDaylight Helium SR2<br>- OpenStack Kilo and OpenDaylight Lithium<br><br>Several additional steps not found in the official documentation were required to achieve the integration, which are documented in the annex part of D4.52.<br>The integration of OpenStack Liberty with OpenDaylight has been unstable. For building an NFV infrastructure (NFVI) and Virtualized Infrastructure Management (VIM), the use of Open Platform for NFV (OPNFV) project is a viable option as it | WP4<br>(D4.51, D4.52) |

| | | |
|---|---|---|
| | provides a solution to the OpenStack - OpenDaylight integration problem. | |
| IVM layer components deployment testing | Available open source testing tools have served us well in testing the functionality and performance of the T-NOVA IVM layer deployment. Specifically, OpenStack Tempest and OpenStack Rally suites were used to test the OpenStack deployment. Robot generic testing framework was used to implement tests evaluating the functionality of OpenDaylight controller and its integration with OpenStack and also, the distributed SDN control plane, monitoring framework and SFC components. | WP4 (D4.51, D4.52) |
| Service Function Chaining | 1. The Open Stack Nova anti-spoofing rules have impact on the SFC implementation, because it drops packets that do not carry the original source MAC/IP address. A workaround for this that was proposed as a patch is using the Noop driver to the nova-compute and neutron that stops the creation of iptables rules. **Lesson learnt:** when native OpenStack rules hamper the support of SFC, apply the patch and leave out security for future work.<br>2. The SFC solution works with VNFs that do transparent L2 forwarding from ingress to egress interfaces w/o altering the packet header, i.e. if L3 VNFs are used in the scenario, than the chain mechanism cannot guarantee that the packets will be received in the VNFs as expected. Therefore the VNFs that operate on layer >L2 and require packets in their original form and/or alter the packets on the egress port, can break the consistency in the chaining rules. **Lesson learnt:** There is not yet a generic solution for chaining all types of services and VNFs. If you want network functions chain, then use L2 VNFs, else you probably need a chain of SFC and a service composition.<br>3. Pure OpenFlow approach in SFC simplifies the implementation as it avoids overheads caused by header encapsulation (such as in the NSH specification), while preserving correctness. **Lesson learnt:** standards are good but not essential.<br>4. Neutron Port forwarding approach in SFC API has been a good and simple choice that has been also adopted (a posteriori to T-NOVA implementation) by the OSS community (ex: sfc in OpenStack) and the industry (ex. N-tuple matching packets in Juniper's OpenContrail). **Lesson learnt:** No need to wait for/adapt the OSS/industry's implementation, as long as you create an own solution that works. | WP4, WP7 (D4.31, D4.21, D4.22, D4.32, D4.5, D7.1, D7.2) |
| SR-IOV | SR-IOV provides the best performance in comparison of any of the fast path packet acceleration technologies currently available, achieve near line rate performance. However this comes at a cost of an inability to migrate. This limitation will need to be address in future releases of OpenStack to drive the generalised adoption of SR-IOV in Telco Cloud Environments. | WP4 (D4.1) |
| DPDK | To achieve near line rate performance the combination of VM/OVS DPDK/DPDK NIC should be selected. VNF applications need to be written natively to support DPDK such as the vTC. 1GB Huge pages should be setup. | WP4 (D4.1) |

| | | |
|---|---|---|
| OVS | Use only if the throughput requirement is <500 Mbps (assuming no other packet acceleration option used), otherwise use OVS DPDK. | WP4 (D4.1) |
| Core Pinning | Use in conjunction with NUMA pinning for dual socket or higher systems. Use for cache intensive VNFs. Use when deterministic SLA behaviour is required. Can be used to reduce the interplay effect between VMs deployed on the same compute node and the L1 and L2 cache misses, resulting in improved performance. Potentially avoid if live migration is required on a recurring basis. | WP4 (D4.1) |
| Service Mapping | Based on the results of the research carried out in Task 3.3 we can conclude that optimization based mapping algorithms provide reliable results and are feasible for implementation. Stochastic control based algorithms on the other hand show in general improved performance in highly congested states. The performances of the implemented optimization-based algorithm could be improved in the future if either more details on the service requests are available to the mapping algorithm (e.g. time when the service is expected to be active, average load expected, etc.) and/or by introducing a module for estimating overall statistical load characteristics. | WP3 (D3.3) |
| Accounting/billing/pricing | Based on the analysis done in T-NOVA marketplace business model we conclude that the most suitable billing model for NFV should follow a similar approach to billing for cloud services XaaS, which is Pay-As-You-Go model. Other considered billing models such as licensing model or subscription have been demonstrated to be not business wise since are not fair nor profitable for the Service Provider or Function Providers due to the cloud perspective business view that NFV has. | WP2, WP6 (D6.4) |
| SDN Controllers Clustering / Load Balancing | Based on the results of the research done in Task 4.2 on load balancing techniques in multi-controller scenarios, we can conclude that the OpenDaylight clustering infrastructure used to validate the proposed algorithms, is not reliable enough to work in real world conditions. Further improvements are needed to ensure data consistency among controllers as the size of the network increases (> 20 nodes) | WP4 (D4.22) |
| VNF Implementation: vSA | In the beginning, the development of the vSa was done and tested on VirtualBox environment, which does not account as a complete NFV infrastructure. Future work for the proposed vSA appliance included the deployment on a complete virtualized network environment, e.g. OpenStack, in order to be tested and validated in more complicated scenarios. The OpenStack deployment raised several networking issues in terms of automated VNF deployment, Service Function Chaining (SFC), traffic forwarding and inter-VM communication, required for the vSA to function properly. The automated and functional integration of the vSA to OpenStack's networking environment, and more specifically to Neutron service was non-trivial and remains to be substantiated and implemented as Neutron at the moment does not offer much freedom and flexibility on arbitrary traffic steering. With other respects, even load balancing and packet filtering (or firewalling) have some similarities, the LBaaS functionality of OpenStack is unable to provide appropriate load balancing for the vSA. As long as | WP2, WP5 (D5.2, D5.3) |

| | | |
|---|---|---|
| | performance is concerned, the use of Unikernel technologies and microservices should optimize the operations of the vSA | |
| VNF Implementation: vHGW | In this project, we had the opportunity to start the design and implementation from scratch. We used this freedom to select novel tools with a view to ease the development cycle and the deployment of our solution. The two main aspects were containerization and double-orchestration.<br>This first important aspect revolved around automation by using a micro-service architecture pattern powered by Docker Containers. First, the development was carried out using familiar environments like the Java and Python programming languages as well as productivity enhancer such as the Eclipse and PyCharm IDEs and the maven/setuptool/virtualenv environments for unit tests and bootstrapping. Once the program were stable, they were placed in containers to be tested on an acceptance environment. Once the acceptance test were validated, those containers were pushed to the DockerHub repository to be downloaded at bootstrapping time in the VNF environment. One major advantage on this method was that we deployed a single Virtual Machine ISO file for each 6 VDU, with self-updating capabilities. In other words, updating the codebase on our source repository automatically updated the Docker containers containing the microservices, which in turn were injected into the VNF, with a substantial increase in productivity and automation compared to other developers that needed to deploy a new VM image each time the code was modified. It should be noted that containers allowed us to add another layer of abstraction in our solution without sacrificing performances.<br>Using classic cloud orchestration tools such as Saltstack has been a key tool to allow a smooth bootstrapring of our code in the VNF, circumventing transient platform instability, by retrying the embedding several time. We used a double orchestration mechanism for scaling: the high-level orchestration, including metric checking and SLA breach events were managed by the Tenor Orchestrator, which communicated with our own low-level orchestration solution based on Docker Swarm. This granted us with a lot of flexibility on the implementation, but also an easier debugging environment when our solution was tested on our local testbed where TeNOR was not deployed. | WP2, WP5 (D5.2, D5.3) |
| VNF Implementation: vPXaaS | vPXaaS was developed with the goal to provide a user-configurable VNF running on the T-NOVA platform. Although the monolithic VNF achieved this purpose, in the future, VNF development will be done with microservices in order to overcome the problem of instability due to software vulnerabilities, service failure and manual configuration errors. The VNF will also be based on a self-upgrading Linux distro, which is fit for cloud computing, instead of a general use distro such as Debian and Ubuntu Server. | WP5 (D5.2, D5.3) |
| NFV/WAN integration | The complexity of integrating a VNF with an existing network connectivity service (e.g. a L2/L3 IP/MPLS VPN) depends on the type of VNF – namely, whether it is supposed to be inserted transparently in the data path, in which case it is likely to have a | WP4 (D4.22) |

| | | |
|---|---|---|
| | direct impact on traffic steering. Thus, when it comes to assessing NFV/WAN integration issues, two basic VNF types should be considered: endpoint and transit VNFs. For the first group (e.g. web proxies, DNS servers, control plane functions) normal IP routing mechanisms should suffice and no specific actions related to traffic steering are required. On the contrary, transit VNFs (e.g. DPIs, traffic classifiers), supposed to be inserted transparently in the data path, usually imply a modification of the way traffic is steered across the network, which requires the intervention of the WICM module. In the latter case, the location of the NFVI-PoP hosting the VNF is an important factor to be taken into account, as it will determine how far traffic has to be redirected into the network, with potentially significant effects on latency and bandwidth consumption. | |

# 5. FUTURE RESEARCH CHALLENGES

This section identifies areas for future research beyond T-NOVA. Section 5.1 positions T-NOVA accomplishments in the framework of the industry evolution towards 5G and identifies specific T-NOVA contributions to that effort. Section 5.2 lists a number of topics for further study with a view to extending T-NOVA work and results.

## 5.1. 5G Positioning of T-NOVA

NFV and network service orchestration are key elements of the future envisioned 5G architecture. NFV is essential to realize the "network softwarization" step, integrating the "basic" network function virtualization inside an end-to-end service management and orchestration framework, and with more advanced programmable controllers enabling a seamless configuration and control of physical and virtual network functions. The initial architectural view released by the 5G-PPP WG [5] dubs Virtual Network Functions as a specific class of 5G Network Functions, and the base element to be extended into the novel concept of *programmable infrastructure*. Their scope is well beyond the "classic" NFV datacenter-hosted environment, encompassing on one hand the full transport and access layers, on the other hand the whole service layer inclusive of management and orchestration.

The plane-layered envisioned 5G architecture tends to drive a vision where the legacy controllers and orchestrators are not overwritten, but rather wrapped-up inside overarching management and orchestration layers speaking to the lower layers through abstracted APIs, oriented to resource slicing and heterogeneous infrastructure modelling. Hence, NFV orchestrators are slated to still play a role in the 5G holistic vision. T-NOVA, which created one of the first real open implementations of an NFV creation, orchestration and deployment framework, can give a great contribution to the next research and development steps, with both its implemented components its acquired know-how. Moreover, as experimented in the 5GEx project (see section 2.3.1.2. ), the T-NOVA Marketplace can provide foundational know-how to 5G projects for developing the service creation layer and the customer front-end.

## 5.2. Topics for future research

T-NOVA was one of the first projects related to NFV funded under FP7 framework. Given the large scope of the area of interest and the very limited set of available solutions, T-NOVA had to provide holistic approach in the implementation plan. As such, T-NOVA had to remain focused into its original promise to implement VNF-as-a-Service use case. In the process a lot of research challenges were tackled, remaining focused in the initial promise. A side effect of this approach, yield some areas to remain unexplored or less elaborated.

With the introduction of the 5G research framework, NFV gained a lot of hype and is currently considered a foundation of the future 5G systems. The requirements imposed by the exploitation of NFV in the 5G systems are creating new challenges regarding latencies, lifecycle management, service orchestration, monitoring and DevOps. In addition, more challenges are arriving from the co-existence of NFV and SDN in the

NFVI-PoP deployments as well as the case of fine control of transport/WAN network resources for provision of end-to-end network services.

In the case of T-NOVA, Table 4 summarizes some future research topics that are relevant for each WP.

**Table 4 - Future research topics**

| Topic | Description | Relevant WP |
|---|---|---|
| Federation | The federation of multiple SPs that are NFV enabled in order to provide an end-2-end network service. This affects the east-west signalling, information exposure, interface standardization, service orchestration. | WP2, WP3 |
| VNF/NS Verification | 3rd party provided VNFs or composed NS need to be automatically verified before used in the operator environment. Frameworks that automate and speed up the procedures are required. This also involves DevOps procedures followed even for SP services and VNFs | WP6 |
| VNF/NS Validation | Workload characterization, NS validation is required to offer end-to-end services and SLA assurance to the customers. Frameworks and methodologies need to be put in place allowing sandbox environment experimentation before the final deployment, as well as in-situ monitoring for operation validation | WP4 |
| WAN Infrastructure Management | SD-WAN infrastructures are emerging offering programmability and SDN support. The role of SDN with NFV is highly critical for the reservation and provision of network resources between NFVI-PoPs and at the edges. Multi-admin domain issues are also relevant. | WP4 |
| Network Slicing | Network and computing resource slicing for facilitation of different verticals over the same physical infrastructure is the new trend in the research community and a requirement for future 5G systems. With the same idea QoS support and continuity across domains and technologies is also part of this topics. | WP4/WP7 |

| | | |
|---|---|---|
| Alternative VIM technologies – Interoperability | Current de-facto standard in the VIM implementation technologies is Openstack. However, there are other solutions emerging either based on container support (i.e Kubernetes) or industry based (i.e. VMware). The support of multiple VIM technologies at the NFVO level will be required in the future to facilitate interoperability. Expanding the current reach of SPs with resources that can be leased over public cloud infrastructures is considered critical in the future. Those infrastructures need to be as easily and autonomously controlled by the SP as its own in order to off-load NSs or VNFs there. Current public cloud infrastructure provides limited control over the resources they provide (i.e AWS). Interfaces and methods for overcoming these issues need to be proposed. | WP3/WP4 |
| Lightweight virtualization technologies | Alternative virtualization technologies need to be explored to cope with instantiation latencies and VM based VNF footprint (i.e. containers, unikernels). | WP5 |
| Security | Security aspects (such as trust, data leakage, authentication, etc.) need to be researched considering the multi-tenant environment for 5G services provision and the current challenges in cloud systems. | WP5 |

# 6. CONCLUSIONS

In the last three years, since T-NOVA kicked off, NFV has gained an increasingly prominent position in the telecommunications industry, as an innovation enabler and a key ingredient of next-generation networks, including 5G. During this period, countless initiatives and activities in similar or closely related areas have been launched; however, T-NOVA was unique in conceiving, building and demonstrating the full end-to-end lifecycle of VNFaaS services, including creation, on-boarding, deployment, operation, monitoring and scaling of network services, as well as implementing the complete set of functional components, namely the marketplace, orchestration layer, NFV infrastructure and end-to-end network connectivity. T-NOVA used ETSI NFV architectural framework as a starting point and foundational reference model, but the breadth of project accomplishments have gone well beyond the boundaries of ETSI NFV.

The experimentation and evaluation campaign conducted in the framework of T7.2 basically confirmed that the original requirements, as specified by WP2 in the first year of the project, have been successfully fulfilled by the T-NOVA platform.

Although NFV has evolved quite significantly in the course of the project lifetime, the concept of VNFaaS, as originally envisioned by T-NOVA, remains valid and still represents a potential catalyst for the uptake of NFV adoption and the emergence of new business models in the Telco industry, enabled by software and the "as-a-Service" paradigm.

The scope of the area of interest of NFV is very wide today. However, T-NOVA primary goal has always been the fulfilment of the original promise to implement and demonstrate the VNFaaS use case. Inevitably, in the interest of not losing focus, some areas remained unexplored or less elaborated. A number of topics for future research have been identified. For the projects willing to undertake some of these challenges, the lessons learnt in the course of T-NOVA activities will certainly provide useful guidance.

# REFERENCES

[1] OPNFV, "OPNFV Web Site," [Online]. Available: https://www.opnfv.org/. [Accessed December 2016].

[2] SDX Central , "OPNFV," [Online]. Available: https://www.sdxcentral.com/projects/opnfv/. [Accessed December 2016].

[3] C. Matsumoto, "With OPNFV, NFV Aims for an Open Source Speed Boost," [Online]. Available: https://www.sdxcentral.com/articles/news/opnfv-nfv-aims-open-source-speed/2014/09/. [Accessed December 2016].

[4] SDX Central, "ETSI (European Telecommunications Standard Institute)," [Online]. Available: https://www.sdxcentral.com/listings/etsi/. [Accessed December 2016].

[5] SDX Central, "How Does ETSI NFV Operate?," [Online]. Available: https://www.sdxcentral.com/nfv/definitions/who-is-etsi-network-functions-virtualization-nfv/. [Accessed December 2016].

[6] ETSI OSM, "Open Source MANO," [Online]. Available: https://osm.etsi.org/. [Accessed July 2016].

[7] RIFT.IO Inc, "RIFT.ware," [Online]. Available: https://riftio.com/riftware/. [Accessed July 2016].

[8] Telefónica Investigación y Desarrollo, S.A. Unipersonal, "OpenMANO," [Online]. Available: http://www.tid.es/long-term-innovation/network-innovation/telefonica-nfv-reference-lab/openmano. [Accessed July 2016].

[9] Canonical Ltd, "Jujucharms - Juju," [Online]. Available: https://jujucharms.com/. [Accessed July 2016].

[10] Openstack , "Tacker - OpenStack NFV Orchestration," [Online]. Available: https://wiki.openstack.org/wiki/Tacker. [Accessed December 2016].

[11] OASIS, "TOSCA Simple Profile for Network Functions Virtualization (NFV) Version 1.0," 2016.

[12] "Tacker + SFC: OpenStack NFV Orchestration Service Function Chaining," [Online]. Available: https://docs.google.com/presentation/d/18AGaiysVgHOd_flHVpObMO7zUkMjJGOQ98CUwkxU1xo/edit#slide=id.ge8ecbd4d4_44_13. [Accessed December 2016].

[13] EWSDN, "European Workshop on Software Defined Networks," 2016. [Online]. Available: https://www.ewsdn.eu/. [Accessed December 2016].

[14] T-NOVA, "D2.21 Overall System Architecture and Interfaces - Interim," 2014.

[15] T-NOVA, "D2.22 Overall System Architecture and Interfaces," 2015.

[16] ETSI NFV, "ETSI GS NFV-IFA 005 V2.1.1 Or-Vi reference point - Interface and Information Model Specification," 2016.

[17] T-NOVA, "D2.32 Specification of the Infrastructure Virtualisation, Management and Orchestration - Final," 2015.

[18] ETSI NFV, "ETSI GS NFV-IFA 006 V2.1.1 Vi-Vnfm reference point - Interface and Information Model Specification," 2016.

[19] ETSI NFV, "ETSI GS NFV-IFA 007 V2.1.1 Or-Vnfm reference point - Interface and Information Model Specification," 2016.

[20] ETSI NFV, "ETSI GS NFV-IFA 008 V2.1.1 Ve-Vnfm reference point - Interface and Information Model Specification," 2016.

[21] ETSI NFV, "ETSI GS NFV-IFA 009 V1.1.1 Report on Architectural Options," 2016.

[22] ETSI NFV, "ETSI GS NFV-IFA 010 V2.2.1 Functional requirements specification," 2016.

[23] ETSI NFV, "ETSI GS NFV 004 v1.1.1 Virtualisation Requirements," 2013.

[24] ETSI NFV, "IFA011 VNF Packaging Specification," 2016.

[25] ETSI NFV, "MAN001 Management and Orchestration," 2014.

[26] ETSI NFV, "TOSCA NFV/SDN Adhoc," [Online]. Available: https://www.oasis-open.org/committees/download.php/58476/VNFD%20mapping%20MAN001-IFA011.docx. [Accessed December 2016].

[27] ETSI NFV, "IFA013 Os-Ma-Nfvo reference point - Interface and Information Model Specification," 2016.

[28] ETSI NFV, "IFA014 Network Service Templates Specification," 2016.

[29] T-NOVA, "D6.4 SLAs and Billing," 2015.

[30] ETSI NFV, "EVE005 Report on SDN Usage in NFV Architectural Framework," 2015.

[31] T-NOVA, "D4.22 SDN Control Plane - Final," 2016.

[32] TM Forum, "TM Forum WebSite," [Online]. Available: http://www.tmforum.org. [Accessed 5 December 2016].

[33] T-NOVA, "D2.42 Specification of the Network Function," 2015.

[34] TM Forum, "TM Forum Zoom," [Online]. Available: https://www.tmforum.org/zoom/. [Accessed 5 December 2016].

[35] TM Forum, "TMF, TR228 TM Forum Gap Analysis related to MANO Work V1.0.1," 2014.

[36] TM Forum, "IG1117 OSS / BSS Futures Overview R14.5.1 Exploratory Report," 2015.

[37] Open Networking Foundation, "Impact of SDN and NFV on OSS/BSS. ONF Solution Brief," 2016.

[38] TM Forum, "What does it take to operationalize NFV and SDN?," [Online]. Available: https://inform.tmforum.org/nfv-it-transformation/2016/09/take-operationalize-nfv-sdn/. [Accessed December 2016].

[39] TM Forum, "TM Forum," [Online]. Available: https://projects.tmforum.org/jira/browse/ZOOM-46. [Accessed December 2016].

[40] T-NOVA, "D8.21 Interim Report on Standardisation Activities," 2016.

[41] TM Forum, "TM Forum," [Online]. Available: http://projects.tmforum.org/wiki/display/ZOOM/2016-07-12+Zoom-wide+Q1%2C+Q2+-+HNWSaaP+Meeting+notes. [Accessed December 2016].

[42] TM Forum, "TM Forum," [Online]. Available: https://projects.tmforum.org/wiki/display/ZOOM/R17+Candidate+Items. [Accessed December 2016].

[43] MEF, "The Third Network: Lifecycle Service Orchestration Vision," 2015.

[44] "5G Vision - The 5G Infrastructure Public Private Partnership: the next generation of communication networks and services," [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2015/02/5G-Vision-Brochure-v1.pdf. [Accessed December 2016].

[45] 5G-PPP, "5G-PPP Work Groups," 2014. [Online]. Available: https://5g-ppp.eu/5g-ppp-work-groups/. [Accessed December 2016].

[46] 5G-PPP, "Software Networks 5G-PPP Working Group Terms of Reference," 2015. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2015/12/5G-I-WG-SDN-NDF-Rev_07.pdf. [Accessed December 2016].

[47] Sonata project, "SONATA NFV: Agile Service Development and Orchestration in 5G Virtualized Networks," [Online]. Available: www.sonata-nfv.eu. [Accessed December 2016].

[48] 5GEx Project, "5GEx Web Site," [Online]. Available: http://www.5gex.eu/. [Accessed December 2016].

[49] SESAME project, "SESAME - Small cEllS coordinAtion for Multi-tenancy and Edge Services," [Online]. Available: http://www.sesame-h2020-5g-ppp.eu/. [Accessed December 2016].

[50] VITAL project, "VITAL (VIrtualized hybrid satellite-TerrestriAl systems for resilient and fLexible future networks)," [Online]. Available: http://www.ict-vital.eu/. [Accessed December 2016].

[51] National Science Foundation, “Network Functions Virtualization With Timing Guarantees,” [Online]. Available: http://www.nsf.gov/awardsearch/showAward?AWD_ID=1563873. [Accessed December 2016].

[52] ChoiceNet, “ChoiceNet: an Economy Plane for the Internet,” [Online]. Available: http://www.ecs.umass.edu/ece/wolf/ChoiceNet/. [Accessed December 2016].

[53] National Science Foundation, “Network Function Virtualization Using Dynamic Reconfiguration,” [Online]. Available: http://www.nsf.gov/awardsearch/showAward?AWD_ID=1525836. [Accessed December 2016].

[54] AT&T, “ECOMP (Enhanced Control, Orchestration, Management & Policy) Architecture White Paper, AT&T,,” [Online]. Available: http://about.att.com/content/dam/snrdocs/ecomp.pdf. [Accessed 5 12 2016].

[55] AT&T, “Our SDN Call to Action,” [Online]. Available: http://about.att.com/innovationblog/031516sdncall. [Accessed December 2016].

[56] M. Dano, “AT&T open sources ECOMP to Linux Foundation, hopes to make it industry's standard for SDN,” 13 July 2016. [Online]. Available: http://www.fiercetelecom.com/telecom/at-t-open-sources-ecomp-to-linux-foundation-hopes-to-make-it-industry-s-standard-for-sdn. [Accessed 5 December 2016].

[57] CORD, “CORD: Reinventing Central Offices for Efficency & Agility,” [Online]. Available: http://opencord.org/. [Accessed 5 December 2106].

[58] T-NOVA, “D4.41 Monitoring and Maintenance - Interim,” 2015.

[59] T-NOVA, “D2.1 System Use Cases and Requirements,” 2014.

[60] T-NOVA, “D7.2 Integrated Pilot and Evaluation Report,” 2016.

# LIST OF ACRONYMS

| Acronym | Explanation |
|---|---|
| 5G-PPP | 5G Infrastructure Public Private Partnership |
| AAA | Authentication, Authorisation, and Accounting |
| API | Application Programming Interface |
| BSS | Business Support System |
| CAPEX | Capital Expenditure |
| CESC | Cloud Enabled Small Cell |
| CESCM | CESC Manager |
| CORD | Central Office Re-architected as a Datacenter |
| DC | Data Center |
| DNS | Domain Name System |
| DoW | Description of Work |
| DPDK | Data Plane Development Kit |
| DPI | Deep Packet Inspection |
| ECOMP | Enhanced Control, Orchestration, Management & Policy |
| EMS | Element Management System |
| EPA | Enhanced Platform Awareness |
| ETSI | European Telecommunications Standards Institute |
| EVE | Evolution and Ecosystem |
| FA | Federation Agent |
| FCAPS | Fault, Configuration, Accounting, Performance, Security |
| FNRM | Federation Network Resource Manager |
| FP | Function Provider |
| FPGA | Field-Programmable Gate Array |
| FSM | Function Specific Manager |
| GUI | Graphical User Interface |
| HNWPaaS | Hybrid Network Platform as a Service |
| ICT | Information and Communications Technology |
| IFA | Interfaces and Architecture |
| IP | Internet Protocol |
| ISG | Industry Specification Group |

| | |
|---|---|
| KVM | Kernel-based Virtual Machine |
| L2 | Layer 2 |
| L3 | Layer 3 |
| LSO | Lifecycle Service Orchestration |
| MANO | Management and Orchestration |
| MdO | Multi-domain Orchestrator |
| MEF | Metro Ethernet Forum |
| MPLS | Multi-Protocol Label Switching |
| MWC | Mobile World Congress |
| NETCONF | Network Configuration Protocol |
| NFS | Network Function Store |
| NFV | Network Functions Virtualisation |
| NFVI | NFV Infrastructure |
| NFVO | NFV Orchestrator |
| NS | Network Service |
| NSD | Network Services Descriptor |
| OASIS | Organization for the Advancement of Structured Information Standards |
| ONOS | Open Network Operating System |
| OPEX | Operational Expenditure |
| OPNFV | Open Platform for NFV |
| OSM | Open Source MANO |
| OSS | Operational Support System |
| OVS | Open vSwitch |
| PE | Provider Edge |
| PoC | Proof of Concept |
| PoP | Point of Presence |
| PxaaS | Proxy as-a-Service |
| QoS | Quality of Service |
| RO | Resource Orchestration |
| SBGC | SBG-PNF Controller |
| SBG-PNF | Satellite Baseband Gateway - Physical Network Function |
| SC | Small Cell |

| | |
|---|---|
| SCNO | Small Cell Network Operator |
| SDK | Software Development Kit |
| SDN | Software-Defined Networking |
| SFC | Service Function Chaining |
| SID | Shared Information/Data Model |
| SLA | Service Level Agreement |
| SO | Service Orchestrator |
| SOTA | State Of The Art |
| SP | Service Provider |
| SSM | Service Specific Manager |
| TMF | TM Forum |
| TOSCA | Topology and Orchestration Specification for Cloud Applications |
| UI | User Interface |
| VIM | Virtualized Infrastructure Manager |
| VNF | Virtualized Network Function |
| VNFaaS | VNF as a Service |
| VNFFGD | VNF Forwarding Graph Descriptor |
| VNFM | VNF Manager |
| VPN | Virtual Private Network |
| VSN | Virtualized Satellite Network |
| WAN | Wide Area Network |
| WG | Working Group |
| WICM | WAN Infrastructure Connectivity Manager |
| WIM | WAN Infrastructure Manager |
| YAML | Yet Another Markup Language |
| YANG | Yet Another Next Generation |
| ZOOM | Zero-touch Orchestration, Operation and Management |