NETWORK FUNCTIONS AS-A-SERVICE
OVER VIRTUALISED INFRASTRUCTURES

GRANT AGREEMENT NO. 619520

Deliverable D2.21

Overall System Architecture and Interfaces

Editor G. Xilouris (NCSRD)

Contributors Eleni Trouva (NCSRD), Aurora Ramos, Felicia Lobillo (ATOS), Pedro Neves, Jose Bonnet, Joao Monteiro, Antonio Gamelas (PTIN), Beppe Coffano, Marco Di Girolamo (HP), Jordi Ferrer Riera, Eduard Escalona (i2CAT), Michael McGrath, Vincenzo Riccobene (INTEL), Dora Christofi, Georgios Dimosthenous (PTL), Georgios Gardikis (SPH), Mamadou Sidibè (VIO), Paolo Comi (ITALTEL), Athina Bourdena, Evangelos Markakis, Evangelos Pallis (TEIC), Panagiotis Papadimitriou (LUH), Antonio Pietrabissa, Donato Macone, Francesco Delli Priscoli, Martina Panfili (CRAT), Yacine Rebahi (FRAUNHOFER), Antonio Cimmino, Piyush Harsh (ZHAW).

Version 1.0

Date July 31th, 2014

Distribution PUBLIC (PU)

Executive Summary

Deliverable D2.21 is the second specification document of the T-NOVA project, which presents the current outcomes of Task 2.2 with regard to the design and specification of the T-NOVA overall system architecture. The aim has been to produce an architectural proposal which fulfils most key NFV-related requirements, implements all use cases defined in D2.1 and at the same time allows implementation in a relatively short timeframe with reasonable technical complexity.

The first step is the survey of currently proposed NFV architectures, including industry initiatives (CloudNFV, HP OpenNFV, Intel/Tieto, ALU CloudBand), research projects (Mobile Cloud Networking, CONTENT, NetIDE, UNIFY) as well as the current specifications, drafts and trends in ETSI NFV ISG.

Considering the use cases and requirements laid out in D2.1 as well as the state-of-the-art in integrated NFV architectures, including standardisation trends, a high-level overall architecture is proposed, which encompasses all the component entities of the T-NOVA system. The architecture is structured in a layered logic, broken down into four layers:

- Infrastructure (NFVI) layer, which includes the physical and virtual elements (compute and storage nodes, hypervisors, data-centre and transport networks) in order to host the elements of the T-NOVA service. Two main domains are foreseen, the NFV data-centre infrastructures (NFVI-PoPs) and the transport networks.
- Infrastructure Management layer, which involves all the management entities capable to virtualise the infrastructure assets and manage them
- Orchestration layer, which includes the T-NOVA Orchestrator, which coordinates the underlying management entities in order to establish and manage the end-to-end T-NOVA services
- Marketplace layer, which includes all the entities which promote multi-actor interaction and facilitate commercial exploitation of the system

Compliance with the current ETSI NFV ISG vision is sought at all stages of system design, especially concerning the terminology and the main architectural blocks, while several extensions are proposed, especially at the Marketplace domain.

Following, the sequence of interactions between subsystems for each use case is presented. These sequences are expressed as UML sequence diagrams and, via them, it is validated that all defined use cases can be realised within the proposed architecture.

Finally, several architectural variations are discussed i.e. alternative architecture configurations and extensions which are considered so as to match diverse operators' and users' needs. Among the extensions considered are: interfacing with a BSS/OSS, multi-provider NFVI-PoPs, support of legacy non-SDN DC infrastructures, support of multiple T-NOVA SPs with a single broker, and multi-SP federation.

Table of Contents

List of Figures

1. INTRODUCTION

Deliverable D2.21 is the second specification document of the T-NOVA project, which presents the current outcomes of Task 2.2 with regard to the design and specification of the T-NOVA overall system architecture.

Most challenges in the field of Network Functions Virtualisation are associated with management operations, such as automated provision, configuration, optimisation, monitoring and rescaling. For this purpose, the key factor towards accelerating NFV adoption is not associated with a single specific technology, but rather with a complete end-to-end management framework which will fulfil all the key NFV-related requirements from the network operators' as well as the customers' point of view.

In this context, this deliverable considers a) the use cases and requirements laid out in D2.1 as well as b) the state-of-the-art in integrated NFV architectures, including standardisation trends, in order to derive a set of design principles and propose a high-level overall architecture which encompasses all the component entities of the T-NOVA system. Compliance with the current ETSI NFV ISG vision is sought at all stages of system design, especially concerning the terminology and the main architectural blocks, while several extensions are proposed, especially at the Marketplace domain.

The deliverable proceeds as follows: Section 2 presents an overview of the NFV architectures, including industry initiatives, research projects as well as the current ETSI specifications. Section 3 presents the overall T-NOVA system architecture, also including the rationale behind main design choices, as well as a high-level description of the main subsystems and interfaces. Section 4 describes the service lifecycle and the sequence of interactions which implement the system use cases. Section 5 discusses architectural variations and configurations, according to various stakeholder needs, and finally Section 5 concludes the document.

2. NFV Architectures Survey

The following sections aim to survey a number of integrated NFV enabling architectures, as proposed by R&D projects currently running, industry frameworks and solutions as well as efforts from Standardisation Bodies related to NFV.

2.1. ETSI ISG NFV

IT and Networks industries have been combining their complementary expertise and resources in a joint collaborative effort to reach broad agreement on standardized approaches and common architectures which address identified technical challenges, are interoperable and have economies of scale.

As a result, a network operator-led Industry Specification Group (ISG) with open membership was setup in the last quarter of 2012 under the umbrella of ETSI to work through the technical challenges of Network Functions Virtualisation (NFV).

However, it should be noted that ETSI ISG NFV does not provide standards but rather produces guideline documents. The ETSI ISG NFV delivers its findings in the form of Group Specifications not in the form of European Norms (EN) or Technical Standards (TS). The outputs are openly published and shared with relevant standards bodies, industry Fora and Consortia to encourage a wider collaborative effort. If misalignments are detected, the ETSI ISG NFV will collaborate with other SDOs in order to meet the requirements.

The NFV ISG also provides an environment for industry to collaborate on Proof of Concept (PoC) platforms in order to demonstrate solutions, which address the technical challenges for NFV implementation and to encourage growth of an open ecosystem.

2.1.1. High-level NFV framework and reference architecture

The NFV concept envisages the implementation of network functions (NFs) as software-only entities that run over the NFV Infrastructure (NFVI). Figure 1, published in October 2013 by the ETSI ISG NFV in its document on global architecture, illustrates the high-level NFV framework, where three main working domains can be identified:

- Virtualised Network Function (VNF), as the software implementation of a network function which is capable of running over the NFVI.
- NFV Infrastructure (NFVI), which includes the diversity of physical resources and how these can be virtualised. NFVI supports the execution of the VNFs.
- NFV Management and Orchestration (NFV MANO), which covers the orchestration and lifecycle management of physical and/or software resources that support the infrastructure virtualisation, and the lifecycle management of VNFs. NFV MANO focuses on all virtualisation-specific management tasks necessary in the NFV framework.

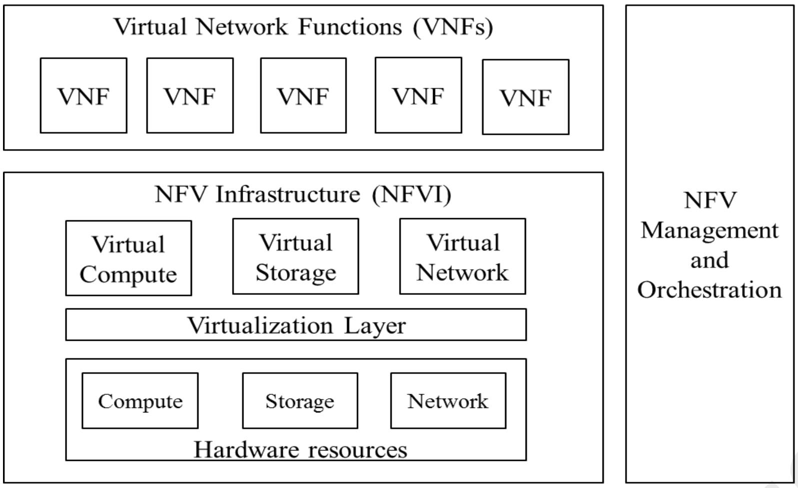

Figure 1. ETSI High-level NFV framework[1]

The NFV architectural framework handles the expected changes that will probably occur in an operator's network due to the network function virtualisation process. Figure 2 shows this global architecture, depicting the functional blocks and reference points in the NFV framework.

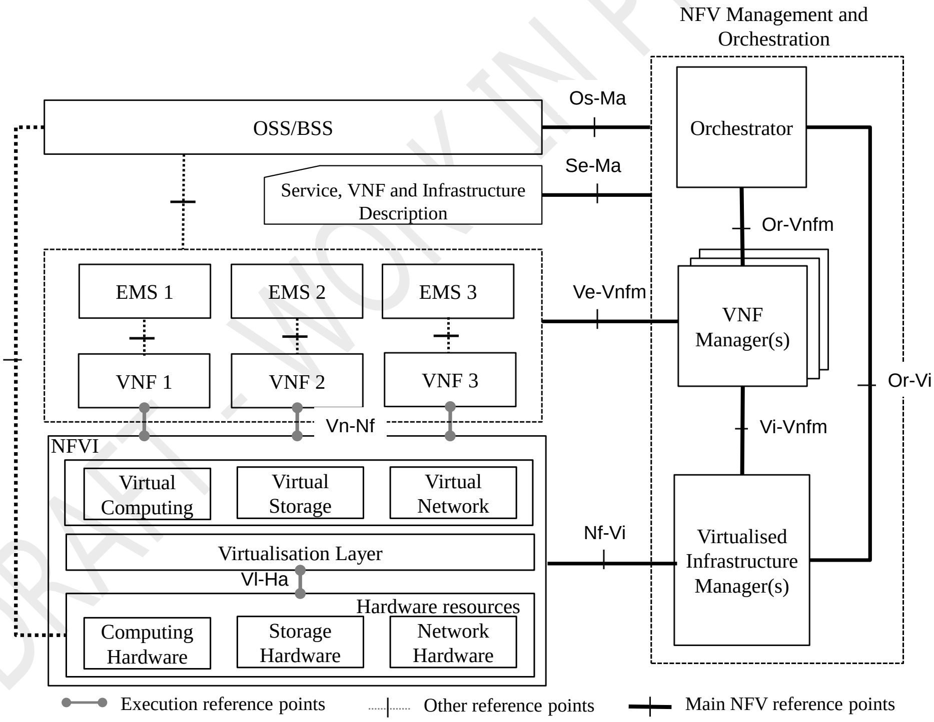

Figure 2. ETSI NFV reference architectural framework[2]

The architectural framework shown in Figure 2 focuses on the functionalities necessary for the virtualisation and the consequent operation of an operator's network. It does not specify which network functions should be virtualised, as that is solely a decision of the owner of the network.

[1] Source: gs_NFV002v010101p - NFV - Architectural Framework
[2] Source: gs_NFV002v010101p - NFV - Architectural Framework

Three ETSI NFV ISG Working Groups have been established following the identification of the above-mentioned domains (i.e. NFVI, SWA and MANO[3]).

2.2. TM Forum

TM Forum (TMF) [1] is a global trade association of service providers and suppliers with the overall objective of progress and success in the digital economy. Key focus areas include trying to reduce costs and risks, improving business agility and growing business through knowledge, tools, standards, training and practical advice. Its activities are organized in three strategic programs: agile business & IT, customer engagement and Open Digital. In relation to the first of them, TM forum works in 4 key areas: Business & IT transformation, Business metrics and KPIs, Cybersecurity and Managing virtualized Network and Services.

In the context of this last mentioned area is where TMF has recently kicked off a major new project with the aim of creating a blueprint for a new generation of service provider support systems to achieve business agility when delivering virtual network and services; it is the zero-touch orchestration, operations and management (ZOOM) project. [2]

The TMF ZOOM project, already in its first stages, has settled its objectives mainly to contribute in the definition of the new virtualized operations environment to delivery of virtualize network services, while allowing the interaction between physical and virtual components to dynamically assembled into personalised services, achieve agility in business and operations practices, define new security approaches to protect infrastructure, functions and services, and complement on-going work of ETSI and other standard organizations working together to provide the management platform and transformation guidance to support successful deployment of NFV.

In this direction, the technical reports released so far by TM Forum ZOOM related to NFV are the following:

- **TR229 ZOOM/NFV User Stories** [3]

This report provides a snapshot of the User Stories that have been identified by the ZOOM project team and encompasses:

- Scenarios being developed in The TM Forum NFV Catalyst program [4]. The projects under this program we have looked at in T-NOVA are the following:
    - Orchestrating SDN and NFV While Enforcing An SLA Over A WAN.
    - CloudNFVTM: Dynamic, Data-Driven Management and Operations.
    - NFV Management Ecosystem.
    - Service Bundling In A B2B2X Marketplace
- Requirements in ETSI and ATIS Reports [5,6] [7] [8].

[3] Note that WGs work at an architecture or functional level, not at an implementation or physical level.

- Agile brain storming sessions amongst Service Providers active in the ZOOM Project.

T-NOVA looked at this TMF work in the previous Task T2.1 when defining use cases, business scenarios, roles involved and initial SLA issues.

- **TR227 TM Forum Specifications relevant for MANO Work** [9]

This document was developed by TMForum with the aim of providing feedback to the ETSI NFV ISG MANO workgroup MANO GS (V0.3.6, 2014-02) about the possible applicability of TM Forum specifications to areas that ETSI is working on. The expertise of the content of the MANO document was provided by TM Forum members who are also engaged in ETSI MANO. The contents were reviewed by the ZOOM Project to validate the TM Forum assets.

The document contains a description of the set of TM Forum documents that are relevant for MANO work. It identifies areas where each TM Forum document can help to standardize the information presented and interfaces of the MANO reference points. These areas, which are under the Business & IT transformation TMForum area are as follows:

- Information framework description (SID): A technology-neutral information model that defines entities and their relationships in a managed environment.
- Business process framework (ETOM): A hierarchical catalog of the key business processes required to operate a service-focused business.
- Integration framework description, that proposes, among others, the creation of a:
    - catalog of Business Services based on Service Oriented principles;
    - platform or domain-based enterprise architecture that provides the business agility required to compete in today's market; and
    - definition of critical standard interfaces that speed integration.

These reports will be taken into account in T-NOVA mainly when defining the marketplace, where a business service catalog will be developed, the service description scheme, SLA management, etc. Extended information about these TMForum activities will be included in the following deliverables.

- **TR228 TM Forum Gap Analysis related to MANO Work** [10]

This technical report is the answer of TMForum's members to the latest NFV Management and Operations (MANO) architecture document, published on July 8 [11], providing an updated GAP. It focuses in two aspects of ETSI work: the MANO interfaces and the MANO information description elements, comparing them with the TM Forum Frameworx APIs and the Logical Resource and Service models, respectively.

Analysing the document, we gather that one of the topics that TM Forum identifies that is missing from MANO, is a detailed implementation model on how to manage operational and business support systems in a hybrid legacy and virtualized environment, something that ETSI is not addressing so far.

In line with this result TMF, ZOOM points out further key future objectives for NFV scheme such as focusing on end-to-end virtual network and operations management, working on network functions virtualization (NFV) procurement requirements, and ensure security and quality of service across operators' networks.

Though the scope the interface between the MANO architecture and the existing OSS/BSS system of operators is not within T-NOVA scope, T-NOVA will provide a first step on the direction of this research line by means of the implementation of the marketplace, which will implement some of the functionalities of a BSS system of an operator, and what will be a first input for latest studies in the interoperability with OSS/BSS existing systems, that TMF ZOOM intends to address in the future.

2.3. CloudNFV

CloudNFV [12] is an open platform for implementing NFV based on cloud computing and Software Defined Networking (SDN) technologies in a multi-vendor environment. The involved companies are: 6WIND, CIMI Corporation, Dell, EnterpriseWeb, Overture Networks, and Qosmos. It has been recently accepted as a proof of concept (PoC) in the frame of ETSI NFV ISG.

2.3.1. Motivation

CloudNFV builds on the NFV ISG work, in order to validate it within the broadest possible framework of service creation and operations, and to incorporate recent critical revolutions such as the cloud and SDN. The project motivation stems from the fact that it considers NFV ISG's scope too large to progress it in time. In this concept CloudNFV proposes an implementation, an extension of ISG principles to ISG adjacent domains.

2.3.2. Overview

CloudNFV, still preserving the alignment with the NFV ISG of ETSI, deploys a mixture of virtual network functions, cloud application components, real network devices and services, and multi-operator federated services. It supports flexible management views based on services, virtual devices, or virtual functions using a service structure modelled on TMF's GB922 hierarchy [13]. The activity is committed to support the interfaces specified by the NFV ISG but also to provide open access to services, composition, deployment, and management features outside the NFV scope.

In the frame of ETSI NFV ISG PoC, a validation of CloudNFV framework will be performed, which defines different scenarios, in an implementation integrating service creation, deployment, and management. The PoC is based in part on Metaswitch's open-source project Clearwater IMS project, which [12] provides an open source IMS suite.

CloudNFV maps ETSI specifications with TM Forum's Information Framework (SID) GB922 and Integration Framework (GB942), and defines an architecture that enables DevOps (development and operations, SW development method based on the collaboration between SW developers and IT operation professionals), service velocity, elastic scalability, business agility.

2.3.3. Architecture

CloudNFV architecture is based on management and orchestration applications built around an agile data/process model called Active Virtualization, which provides for order/contract and policy storage ("Active Contract") and resource state information ("Active Resource") provided by EnterpriseWeb. Service orders are optimized through Active Virtualization then provisioned on cloud infrastructure using Overture Network's Ensemble Service Orchestrator, which instantiates the virtual network functions through OpenStack Nova and connects them using OpenStack Neutron. The overall architecture is depicted in Figure 3.

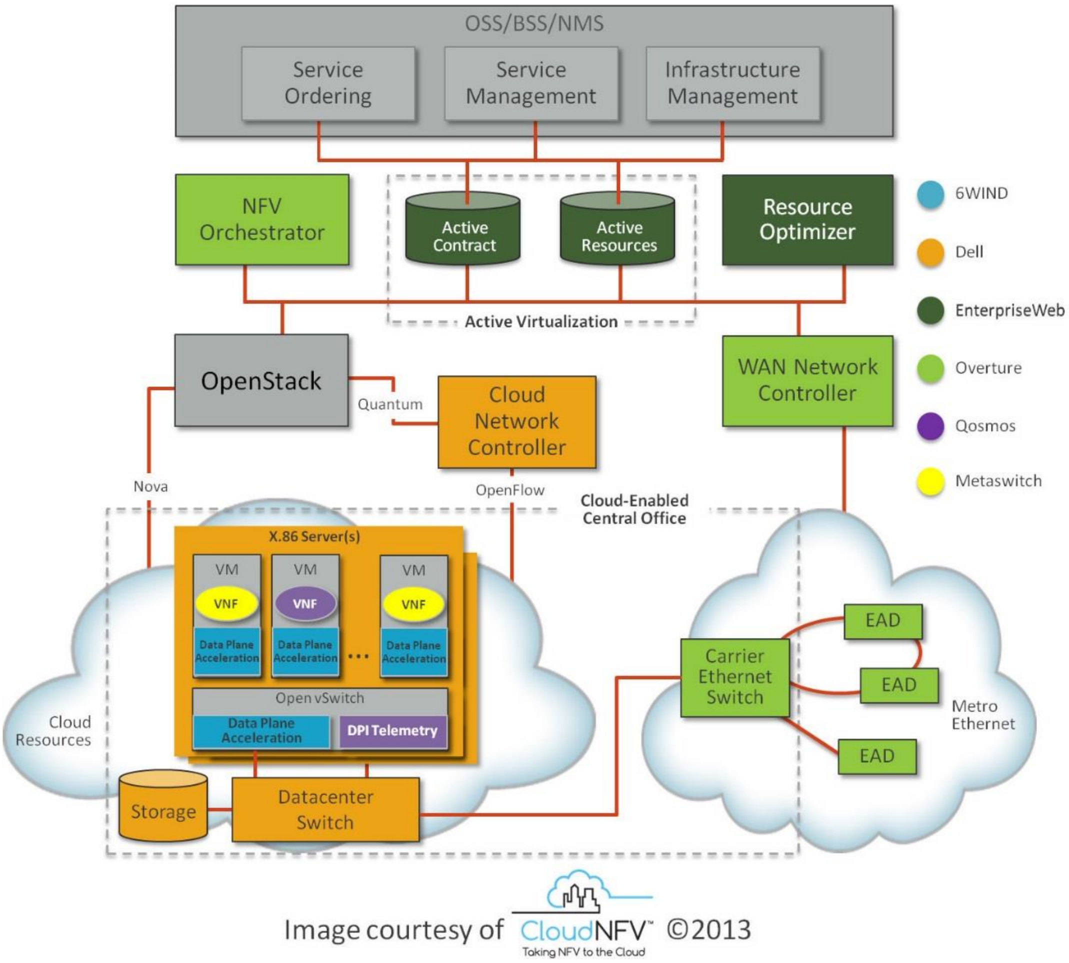

Figure 3. CloudNFV Architecture

The architecture is designed to support open interfaces for carrier federation at both infrastructure and orchestration levels. This provides the capability to CloudNFV to deploy assets on a per-customer basis and also as Infrastructure Services (shared between users). Moreover mixed physical devices and virtual functions configurations are supported.

2.4. HP OpenNFV

OpenNFV [14] is a comprehensive project launched by HP, built around a proposed open reference architecture, encompassing a service portfolio, and enforced by an ecosystem of ISVs, NEPs and application developers.

HP architecture is aligned with the ETSI model, and HP has a number of active contributors in the NFV ISG. OpenNFV main components are a NFV Infrastructure and a NFV Orchestrator module, in turn based on HP Converged Infrastructure and HP Converged Cloud propositions. It also capitalizes on the SDN role, and on HP's SDN technology assets. It is a modular architecture, basically vendor agnostic and allowing a modularized approach to NFV take-up. A high level picture of OpenNFV architecture is presented in Figure 4.

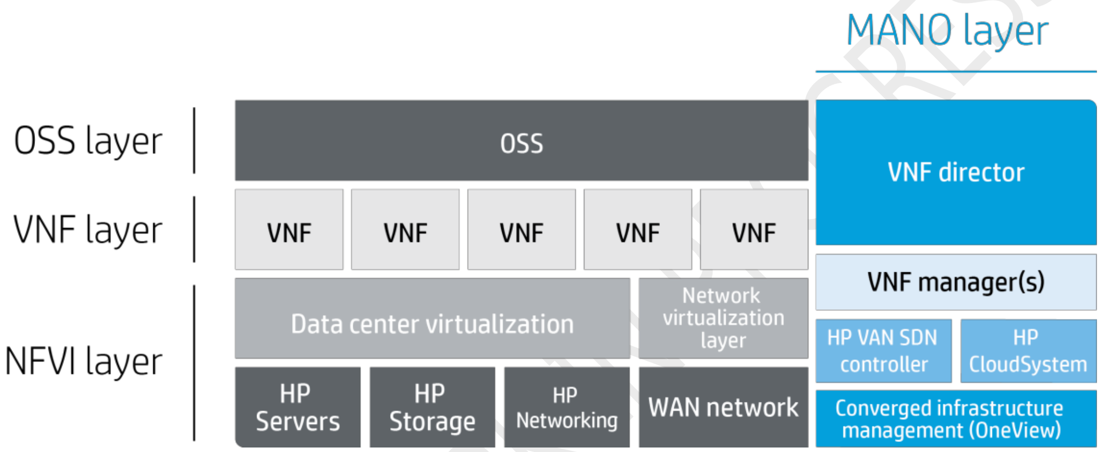

Figure 4. HP OpenNFV architecture

Figure 5 shows a functional view of the OpenNFV architecture.

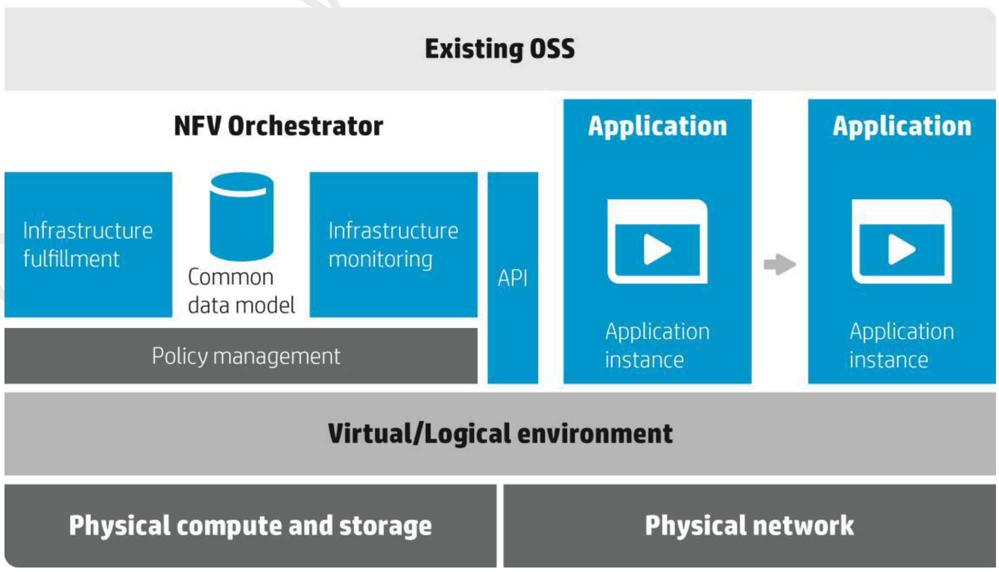

Figure 5. HP OpenNFV Functional Architecture

The virtualized NFV infrastructure layer includes compute, storage and networking plus their element management components. It's a logical infrastructure layer, designed to support both physical and virtualized resources at the same time. The NFVI layer offers pre-integrated modules supporting virtualization and performance enhancement technologies like SR-IOV and DPDK, to ensure that the hardware horsepower is suitable to seamlessly virtualize network functions without performance degradation. Multiple hypervisors can be supported, like VMware ESX, Microsoft Hyper-V, and Linux KVM.

On the network virtualization side, the architecture can seamlessly support automated provisioning of both physical and virtual networks. HP physical switches also support L2/L3 forwarding, bridging the gap between legacy and SDN-controlled networks. A comprehensive heterogeneous network management platform is also available.

HP Cloud System is the module that allows the NFVI layer to be viewed as a service, accessing the infrastructure resources through an OpenStack interface.

The NFV Orchestrator module (implemented by HP in the *NFV Director* component) implements the functions prescribed by the ETSI model MANO layer specification:

- Configuration of the virtualized computing infrastructure;
- Networking functions for the computing infrastructure to communicate with the physical network in an NFV application;
- Common data model for services and NFV instances (forwarding graph);
- Service and resources status management, including integration with existing OSS;
- Policy management: supervision of static (configuration) and dynamic (operation) policies.

The NFV Orchestrator is hypervisor-agnostic, so it can support different solutions both proprietary and open source.

2.5. Qosmos/Intel/Tieto

Intel has been an active player in supporting the development and evolution of NFV and SDN through industry and vendor specific initiatives. The network builders program for example is and industry initiative comprising of more than 70 companies. The goal of the program is to make it easier to build, enhance, and operate SDN/NFV-based infrastructure, while lowering capital and operating expenditures. The program publishes function specific architectures such as vEPC, vBRAS, vCPE etc. [15]

Intel has also worked with a number of industry partners to develop reference architectures for NFV. For example Intel has collaborated with Telecommunication Equipment Manufacturers (TEM's) provider QOSMOS and systems integrator Tieto to develop a reference architecture for NFV based on ETSI NFV ISG principles [16]. The architecture is designed to address key barriers to NFV adoption such as:

- Real-time performance
- Service Awareness - Allocation of resources to network functions based on the services they support

- Service Availability - Achieving the same level of service availability for end-to-end virtualised networks as non-virtualised.
- Network elasticity

The key components of the architecture are shown in Figure 6. This architecture was used to form the basis for a proof of concept (POC) that demonstrated a number of ETSI defined VNF use cases. The key components of the architecture are as follows:

Virtualised Network Functions – The PoC implemented reference software for LTE eNodeB and EPC (MME, SGW, and PDN GW), along with Tieto's Diameter Signalling Controller (DSC), which was deployed as a VNF. Qosmos' intelligent DPI was included and could be deployed either within a VNF or as a standalone virtual networking function component (VNFC). The associated Element Management System (EMS) for each VNF is integrated within the VNF subsystem, which monitors the operational condition of the VNF's as part of the overall Telecommunications Management Network (TMN).

NFV Management and Orchestration – This sub-system is responsible for the management of VNF deployments and lifecycle. The SDN controller is also contained within this sub-system with responsibility for flow control to enable intelligent networking. The architecture is orchestrator-agnostic however the reference implementation is based on OpenStack. The network management solution interfaces to OpenStack via Heat are used for automation and deployment.

Network Operations (OSS/BSS) – The architecture and reference implementation supports a NETCONF interface for OSS and a Cloud Infrastructure Management Interface (CIMI). The intent is to demonstrate that new VNFs can be deployed and managed from an end-to-end perspective within a telco cloud environment.

Cloud Platform – This sub-system is based on OpenStack, which has been modified by Tieto to include telco-grade supervision, statistics, diagnostics, fault and performance management capabilities.

SDN Networking – The SDN controller is compatible with OpenStack Neutron and supports SDN networking and legacy network management system (NMS) integration, and provides supervision, statistics and performance management. OpenFlow is used to communicate between the SDN controller and the Open vSwitch, which is managed by the OpenFlow Controller.

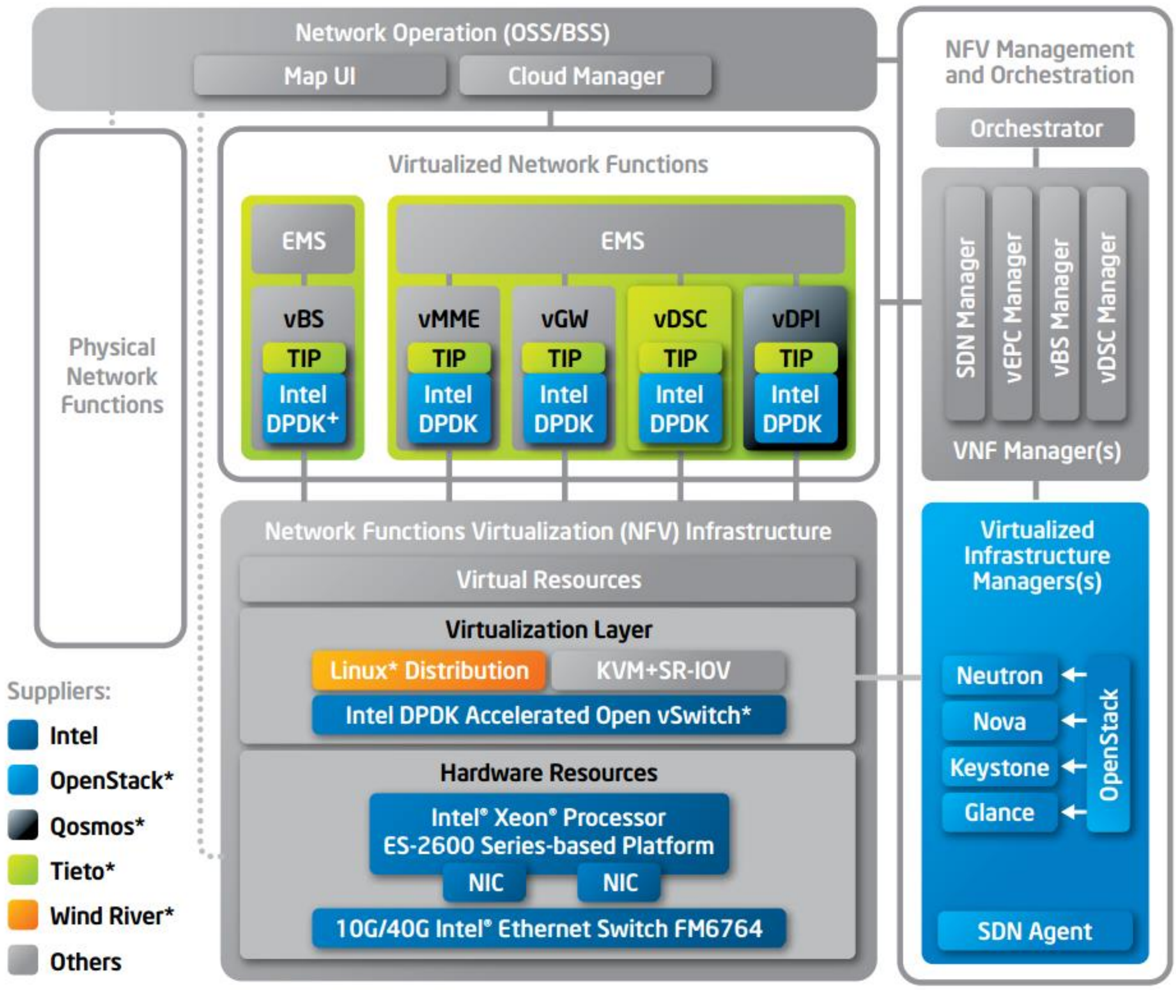

Figure 6. Intel NFV High Level Architecture

The designed architecture was validated through a PoC implementation and was used to illustrate three specific ETSI NVF use cases:

- Virtual Network Functions as a Service (VNFaaS)
- Virtualisation of a Mobile Core Network (EPC) and IMS
- Virtualisation of a Mobile Base Station (RAN)

2.6. Alcatel-Lucent CloudBand

The CloudBand NFV platform [17] aims at transforming carrier-grade service provider (B2B proposal) networks with distributed footprint into a single, manageable, virtual cloud.

2.6.1. CloudBand Overview

The overall architecture of CloudBand is illustrated in Figure 7.

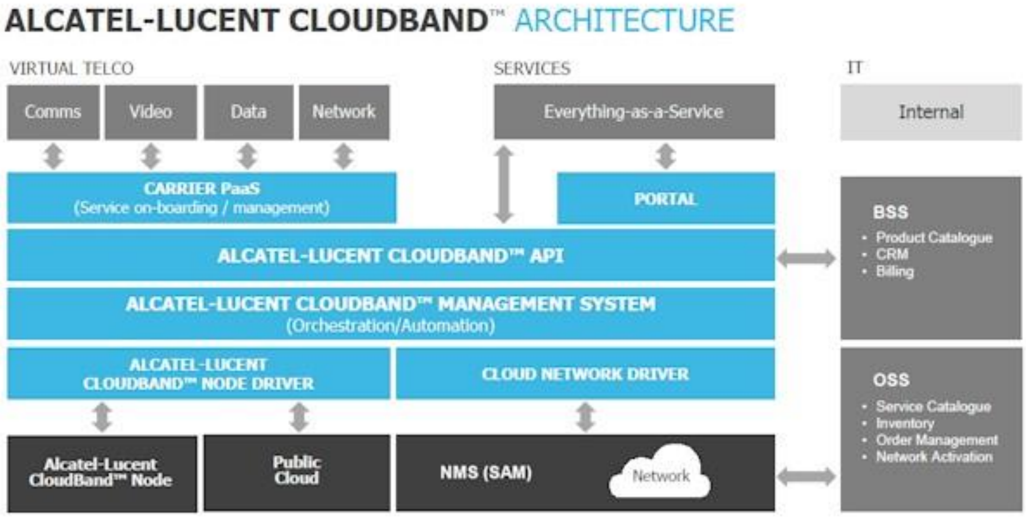

Figure 7. CloudBand Overall Architecture

The major characteristics of the architecture in Figure 7 are:

- Mono-vendor solution NFV / Cloud management
- Possible expansion to Everything as a Service offer[4]
- Cloud Infrastructure accesses via API from NFV and BSS
- Interface to public cloud
- NMS includes management of the CloudBand Network
- Multi-vendor support for cloud nodes

In addition, concerning the deployment, a distributed cloud infrastructure can accommodate a large number of small and medium datacentres, placed in different sites to spread services across multiple locations.

CloudBand also comes with a carrier PaaS (cPaaS) management tool that manages application lifecycle automatically and on-boards VNFs in the cloud. It hides the complexities of infrastructures and OS. It can automate and optimise application services like IMS or any other carrier grade service and facilitates operators to concentrate on other aspects of the application lifecycle namely provisioning, monitoring, healing and scaling (as shown in Fig 8).

The management of the PaaS allows defining rules for the placement, SLA, placement Zones, monitoring, Cloud resources, High availability (HA) and redundancy, tracking the full lifecycle of the deployed applications.

[4] The platform is supposed to be flexible to accommodate other services in the paradigm XaaS beside the ETSI VNF.

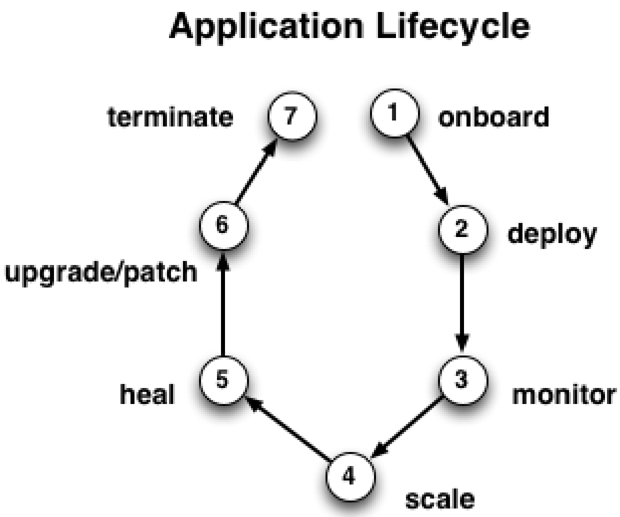

Figure 8. CloudBand Application LifeCycle

2.6.1.1. Analysis of key components

Orchestration and Automation

CloudBand orchestrates and automates cloud resources across the network and geo-distributed data centers. It decouples existing physical compute and storage hardware and network resources into a virtualization layer, which runs at the service providers' own data center. Administrators allocate and control resources to users and applications from one, carrier-grade pool, eliminating the complexity of managing multiple systems: the underlying **Cloud Node Driver** pools processing and storage -- like traditional cloud solutions; the **Cloud Network Driver** pools network resources.

Cloud Node Driver

It is a mediator layer that supports different node implementations. The driver manages communications, handles resource provisioning and de-provisioning, verifies compliance with the operating systems and conveys monitoring information from the Cloud Node to the CloudBand Management System. It supports: Alcatel-Lucent Cloud Node, supported via the CloudStack API, supports KVM hypervisor and others, Federated clouds, such as Amazon EC2.

Cloud Network Driver

Operators can create separate, virtualized and distributed LAN networks on demand across multiple locations, enabling seamless connectivity to their corporate VPN, through VPN stitching (easy like VM creation). Cloud Network Driver abstracts and automates network provisioning and monitoring while being agnostic to any third-party network equipment and/or software-defined network (SDN) implementation. It supports the **OpenStack Neutron** module. "My Networks" is the manager of the network from the portal. They can view all VPNs associated with the CloudBand account, and can add, edit and view CloudBand networks. The supported features are: Dynamic IP Allocation, Managed and Unmanaged Networks, Network-as-a-service, VPN Stitching, Route Domains, Network Uplink Services, Hybrid IP Networks, IPv6 Support for Guest VMs, Network Performance Monitoring, IP Address Management, QoS Profiles, Network Auditing, Virtual Routing and Forwarding (VRF) Peering, VLAN Uplink, Interfaces to Network Management Platforms (multi-vendor), Networked-cloud APIs

Analytics and Monitoring

All metrics are collected every minute from the cloud driver, and are stored in the Cassandra open source distributed database management system.

Available reports include: Monitoring Metrics (Cloud Node CPU-Mem-storage and VM CPU allocations), Resource Allocation Predictions, Alerts, and Resource Heat Map.

2.6.1.2. Integrated services

Load Balancing-as-a-service (LBaaS)

CloudBand's load balancing-as-a-service (LBaaS) offers load balancing (HTTP and HTTPS) across distributed network, a service that can be purchased and deployed immediately. LBaaS distributes traffic across several VMs with an active/active[5] configuration. The CloudBand Management System implements the load balancer as two highly available VMs on two different servers, but exposes a single virtual IP for service. The load balancer constantly monitors VM health and provides traffic distribution to these VMs based on the number of TCP connections.

Object Storage-as-a-service

Object storage is used for automating and streamlining data storage in the cloud. Separate atomic units of storage are assigned a unique identifier, used to retrieve the object without needing to know the data's physical location. CloudBand implements open-source ceph distributed network storage, which uses the reliable, autonomic, distributed object store (RADOS) at the lower layer, storing data of unlimited object size. Objects can be any item stored, including metadata and attributes. CloudBand exposes object storage via its storage subsystem and is automatically installed during node provisioning.

2.6.1.3. Automated Lifecycle Management - Descriptors

The CloudBand platform includes an automated lifecycle management. With the virtualization of the telecommunications industry, service providers have the opportunity to introduce tools similar to cloud automation as part of their operational arsenal.

[5] In Active/Active mode two or more servers aggregate the network traffic load and working as a team distributes it to the network servers. The load balancers can also remember information requests from users and keep this information in cache. Should the user return looking for the same information the user will be locked onto the load balancer that previously served them and the information provided again from the cache without the network server having to respond therefore reducing network traffic load.

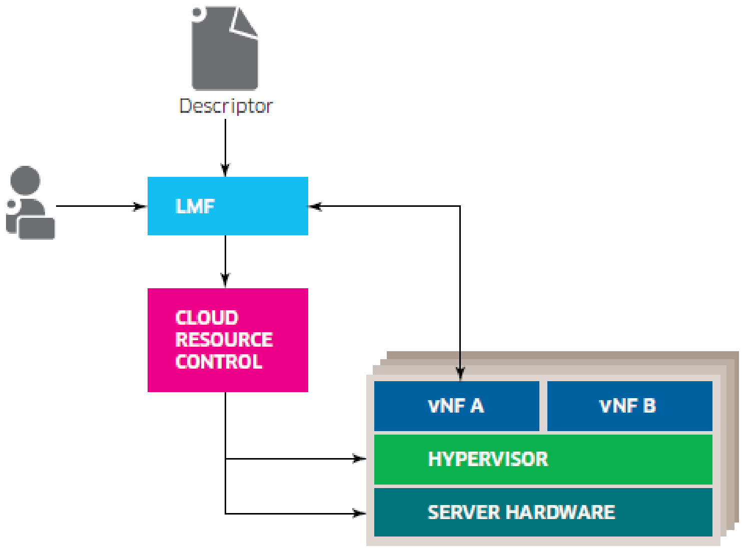

Figure 9. VNF deployment sequence

The process uses a descriptor that is generally provided by the VNF vendor. This descriptor defines the structure of the VNF (and sub-functions needed to run as an independent VM) and deployment and operational aspects, such as computation, storage and networking requirements.

These descriptors are mapped to requests to the cloud resource control to create VMs and to identify software images to be downloaded and initiated on those VMs. Once the VMs are up and running, the LMF configures parameters of the VNF components based on instructions in the descriptor.

Lifecycle management also plays a role in elasticity – scaling the VNF up and down-adding, removing or resizing VMs as needed. Triggered by either an operator decision or a VNF event, the LMF executes the required actions.

In principle, every virtualized network function could come with its own proprietary lifecycle management, but service providers would obviously benefit from a consistent approach across all VNFs. The use of tools for automated lifecycle management and convergence on one or a small set of compatible tools will provide significant benefits to both, service providers and vendors.

2.7. Project MCN

The Mobile Cloud Networking (MCN) [18] is a 3-year European co-funded (FP7) project, which started in November 2012 and will end in October 2015. The project approaches the integration between the Cloud and Telco worlds, making Operators benefit from the principles of virtualization.

The consortium comprises Network Operators, Cloud Providers, Vendors, University and Research Institutes, as well as SMEs.

2.7.1. Overview

The project focuses, in particular, on mobile operators. For this reason, the main target is to fully cloudify the whole components of a mobile network operation, namely (see Figure 10):

- the access (RAN - Radio Access Network);
- the core (EPC – Evolved Packet Core);
- the services (IMS – IP Multimedia Subsystem, CDN – Content Delivery Networks, DSS – Digital Signage);
- the Operational Support Systems (OSS) (Provisioning, Monitoring, SLA Management);
- the Business Support Systems (BSS) (CRM – Customer Relationship Management, Charging, Billing).

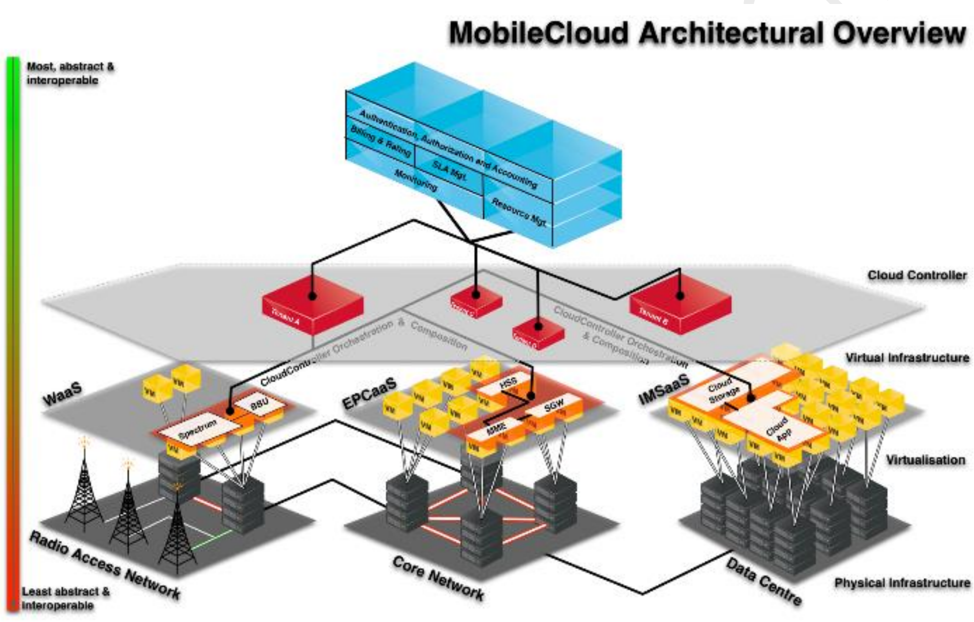

Figure 10. Mobile Cloud Networking Vision

Note: In the case of OSS/BSS case, the focus is two-fold. Firstly, support the operations and businesses of cloudified networks; secondly, the virtualization itself of the OSS/BSS platforms.

Beyond the virtualization, the project explores the "as a Service" (XaaS) concept, where the functions are provided as a full operational service. That means the customer of the service does not need to worry about implementation, deployment and dimensioning details of it. This approach allows an easy creation of "end-to-end (e2e)" services by composition of basic services. As an example, it can be considered the creation of an MVNOaaS service by the composition of RANaaS+EPCaaS+IMSaaS.

2.7.2. Motivations

The motivations for this project relate to the exploitation of cloud principles by the Telcos, such as flexibility, efficiency, pay as you go, or cost reduction. In particular, this can be explored by a sector that has been obliged to severely reduce margins and lose business that they traditionally controlled.

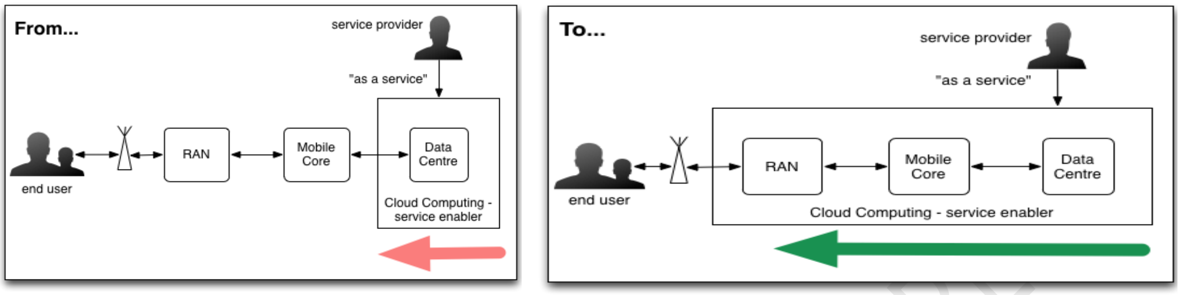

Figure 11. Extending the Concept of Cloud Computing Beyond Data Centres Towards the Mobile End-User

The primary motivations of the MobileCloud project are to:

- Extend the Concept of Cloud Computing beyond data centres towards the Mobile End-User.
    - “One Service (atomic): Mobile Network + Computing + Storage”
    - “On-Demand, Elastic, and Pay-As-You-Go”
- Enable a Novel Business Actor, the Mobile Cloud Provider
- Redesign the Mobile Network Architecture for Exploiting and Supporting Cloud Computing
- Deliver and Exploit the Concept of an End-to-End Mobile Cloud for Novel Applications

2.7.3. MCN Envisaged Services

The project focuses on the implementation of five main (MCN) Services, which are the ones to be exposed to customers. Beyond these MCN Services, other auxiliary (Support) Services are used to support the operation of the MCN Services in technical aspects, such as load balancing (LBaaS) or Database (DBaaS), and business aspects, such as SLA management (SLAaaS) or Rating, Charging and Billing (RCBaaS). Of course, all those are the basic resources, computation, and storage and network support services.

The **RANaaS** (Radio Access Networks as-a-Service) service is responsible to provide LTE native access radio to provide access to end users to an LTE network (core). RAN can be split into the RRU (Remote Radio Unit), which is responsible to broadcast the physical signals through the antennas, and the BBU (Base Band Unit), which is responsible process the signal and make all necessary computations. The RANaaS virtualizes the BBU part, moving the resources to the areas (e.g. eNBs) that need more computation power, making this way more efficient utilization of resources.

The **EPCaaS** (Evolved Packet Core as-a-Service) service is responsible to provide LTE connectivity to the end users, by forwarding traffic from the RAN to the Internet and vice-versa. The ECPaaS virtualizes the core as a whole by virtualizing its internal components (e.g. P-GW, S-GW, MME, etc.).

The **IMSaaS** (IP Multimedia Subsystem as-a-Service) service virtualizes the IMS internal components (e.g. P/I/S-CSCF, HSS, etc.) in a similar way as the EPC.

The **CDNaaS** (Content Delivery Network as-a-Service) service is responsible to provide a caching mechanism that can be used to cache popular contents to the end users.

The **DSSaaS** (Digital Signage System) service is responsible to provide a digital signage service. The virtualization allows the adaptation of the service to the usage performance.

The project focuses on different scenarios and business models. The following general use cases depicts the most significant use cases:

- **Service Providers** provide functional blocks in the "XaaS" model (RANaaS, EPCaaS, etc.)
- **Network Functions' Vendors and MNOs** are natural providers of Network Functions
- **Corporates, Utility Providers and MVNOs** are natural Customers of Network Functions

2.7.4. Architecture

Mobile Cloud Networking overall architecture design [19] is mainly governed by service oriented design principles. Every service in MCN has the same provisioning and lifecycle management pattern and architecturally follows the global MCN reference architecture.

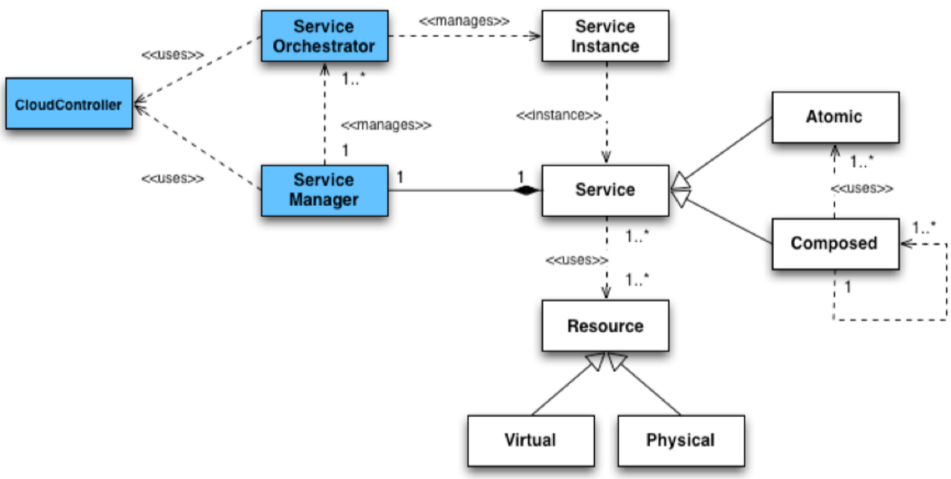

Figure 12. MCN Architectural Entities Relationship

A brief description of key components follows:

> **Service Manager:** provides an external interface to the Enterprise End User (EEU) and is responsible for managing service orchestrators, it has business and technical management functions.
>
> **Service Orchestrator:** it oversees the end-to-end orchestration of a service instance. It is responsible for managing the Service Instance and in particular its components (SIC), once it is created and running.
>
> **Cloud Controller:** provide the signaling and management interfaces to enable the common (northbound), and technology-specific (southbound) control planes. It provides both atomic and support services required for realizing SO needs. The main MCN architectural entities that interact most with the Cloud Controller are the SM and SO.

Service Orchestrator implementation in MCN project is service specific as it depends on the domain knowledge of the respective NF it is implementing. Some of the prominent NFs being virtualized and managed as a service in MCN are – EPC, IMS, DNS, OSS/BSS (RCB), AAA, CDN, etc.

An EEU (Enterprise End User) can request a service, which is in turn realized through composition of many services including atomic and other composed service. The MCN architecture through SM and SOs support such service composition, provisioning and runtime management in a standard manner.

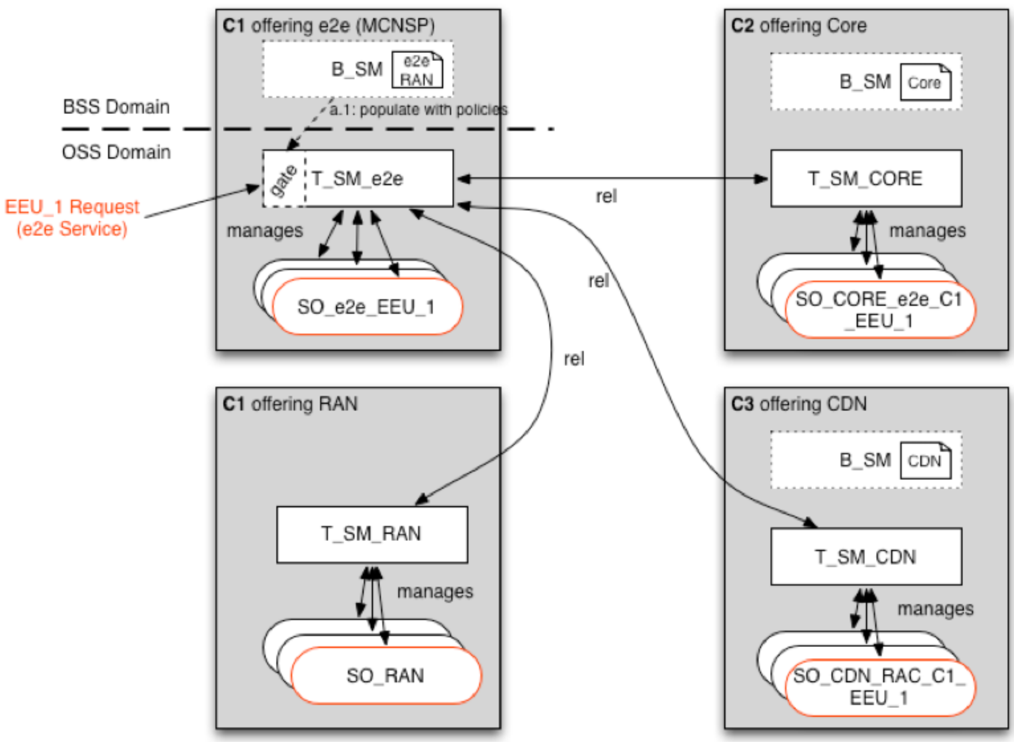

Figure 13. MCN Service Composition Scenario

Figure 13 shows how a complex service consisting of EPC, RAN, and CDN can be created in MCN. Since each NF is virtualized and the MCN architecture allows VNFs to be provisioned on demand, the business entities can thus easily offer such VNFs as a service and service provisioning (parameter setting, configurations, etc.) can either be achieved as managed or unmanaged offering by the business entity.

2.8. Project CONTENT

CONTENT (Convergence of Wireless Optical Network and IT Resources in Support of Cloud Services) is a 3-years European co-funded (FP7) project, which started in November 2012 and will end in October 2015 [20].

CONTENT aims at offering a network architecture and overall infrastructure solution to facilitate the deployment of conventional Cloud computing as well as mobile Cloud computing introducing new business models and facilitating new opportunities for a variety of business sectors.

The project consortium is composed by 6 partners three (3) of which are industrial partners (JUNIPER -, PRIMETEL and Nextworks) major players in the wireless and the optical network domains and the rest come from the academic world including two (2) European universities (UNIVBRIS, UTH) and two (2) research centres (AIT and i2CAT). i2CAT and PrimeTel also participate in the T-NOVA project.

2.8.1. Project Overview

The project focuses on developing a next generation converged network infrastructure to support the network of the future. The proposed infrastructure model will be based on the IaaS paradigm and will aim to provide a technology platform interconnecting geographically distributed computational resources, which can support a variety of Mobile Cloud services.

CONTENT will focus on a hybrid wireless solution based on Wi-Fi and LTE and a WDM access-metro network with frame-based sub wavelength switching granularity, incorporating active nodes that also support backhauling of the wireless access network. To support the IaaS paradigm, CONTENT will adopt the concept of physical resource virtualization across the technology domains. One of CONTENT’s objectives is to offer a rationalized cost and energy efficient network infrastructure suitable to support Cloud and mobile Cloud services.

CONTENT project objectives are:

1. Seamless integration of wireless and wired optical access-metro network domains to provide end-to-end connectivity of computational resources with fixed and mobile users.
2. A cross-domain and technology virtualization solution allowing the creation and operation of infrastructure slices including subsets of the network and computational physical resources.
3. Support of dynamic end-to-end service provisioning across the network segments, offering variable QoS guarantees, throughout the integrated network
4. Cost and Energy Efficiency

2.8.2. Proposed CONTENT Architecture

CONTENT proposes a layered architecture with the aim to facilitate the main principles of its novel proposition i.e. cross-technology virtualization in support of optimised, seamless and coordinated cloud and mobile cloud service provisioning across heterogeneous network domains. The overall CONTENT architectural structure is illustrated in the figure below and includes the following layers:

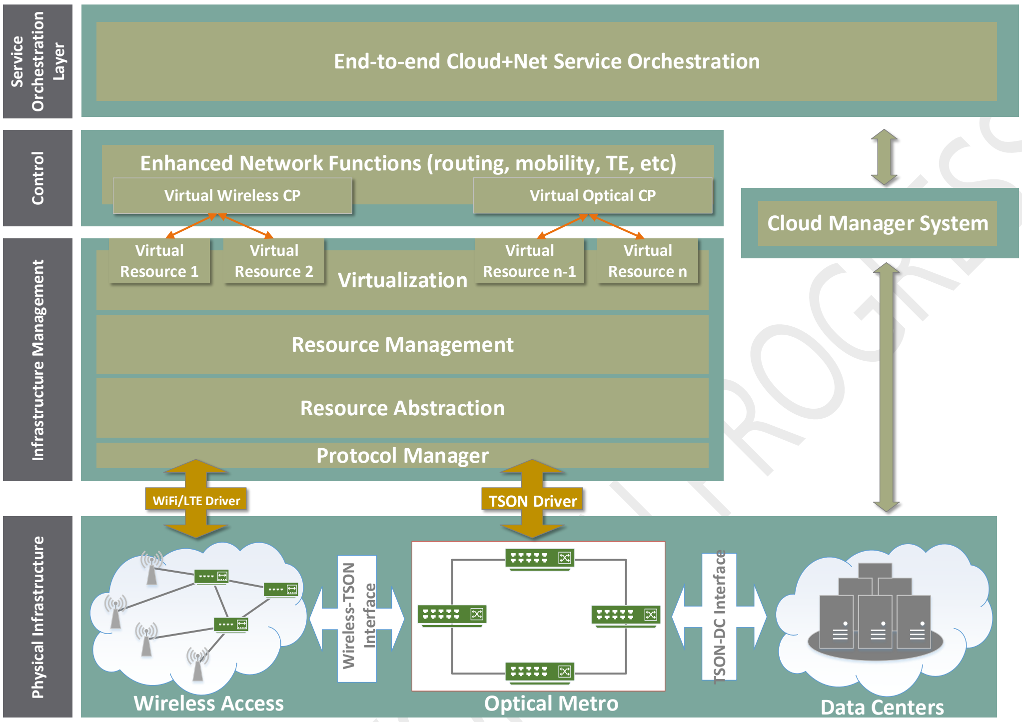

Figure 14. Overall CONTENT layered architecture

Heterogeneous Physical Infrastructure Layer: including a hybrid wireless access network (LTE/Wi-Fi) domain, and an optical metro network domain (TSON) interconnecting geographically distributed data centres, supporting frame-based sub-wavelength switching granularity.

Infrastructure Management Layer: has overall responsible for the management of the network infrastructure and the creation of virtual network infrastructures over the underlying physical resources. This involves functions including resource representation, abstraction, management and virtualization across the heterogeneous network domains. An important feature of the functionalities supported, is orchestrated abstraction of resources across domains, involving information exchange and coordination across domains.

Control Layer: responsible to provision IT and (mobile) connectivity services in the cloud and network domains respectively. The focus of the project is on the network side, where the control layer establishes seamless connectivity across heterogeneous technology domains (wireless access and optical metro) through a coordinated, end-to-end approach to support optimized performance, QoS guarantees as well as resource efficiency and sustainability.

Service Orchestration Layer: responsible for efficient coordination of the cloud and network resources, in order to enable the end-to-end composition and delivery of integrated cloud, mobile cloud and network services in mobile environments supporting the required QoE.

2.9. Project UNIFY

UNIFY (Unifying Cloud and Carrier Networks) project [21] has the goal of increasing the potential of virtualization and automation across the whole networking and cloud infrastructure. The project is focused on enablers of a unified production environment and will develop an automated, dynamic service creation platform, leveraging a fine-granular service chaining architecture. UNIFY proposes a service abstraction modeland a service creation language to enable dynamic and automatic placement of networking, computing and storage components across the infrastructure. It will develop an orchestrator with optimization algorithms to ensure optimal placement of elementary service components across the infrastructure. Moreover, UNIFY intents to research new management technologies and develop a Service Provider DevOps concept to address the dynamicity and agility of new services. It will investigate the applicability of a universal network node based on commodity hardware to support both network functions and traditional data center workloads. Therefore, the UNIFY consortium researches, develops and evaluates means to orchestrate, verify and observe end-to-end service delivery from home and enterprise networks through aggregation and core networks to data centers.

The project started in October 2013 and is in the course of finalizing the architecture.

2.9.1. Basic Concepts

UNIFY focuses on enabling virtualization and automation across the whole networking and cloud infrastructure in a unified manner. It envisions full network and service virtualization to enable rich and flexible services and operational efficiency.

Through the design of universal hardware architectures UNIFY aims at improving the intelligence and flexibility of the network and so open up opportunities for new converged fixed and mobile end-user service offerings, while also enabling advanced programmability and efficient virtualization, providing means to reduce the cost of new service creation and operation.

The service architecture will be based on a unified view of the infrastructure, covering both the traditional network and the data centre entities. The programmability framework envisaged by UNIFY will create unprecedented opportunity for innovation in the fields of cloud computing and networking. A lower technical barrier for sharing resources and services will unlock new business opportunities, which can be rapidly implemented on a unified production environment.

2.10. Project NetIDE

NetIDE (An integrated development environment for portable network applications) is focused on delivering a single integrated development environment (IDE) to support the whole development lifecycle of network controller programs in a vendor-independent fashion [22]. The project addresses the lack of a common approach to the development and deployment of SDN Controllers. The increasing numbers SDN frameworks being proliferated such as Pox, Floodlight, Ryu, Trema, OpenDaylight, etc are exacerbating this problem. Secondly there is a lack of integrated development support for SDN solutions. Finally there is a lack of critical services and tools for the network layer i.e. key services to support integration between development and testing of applications are still missing.

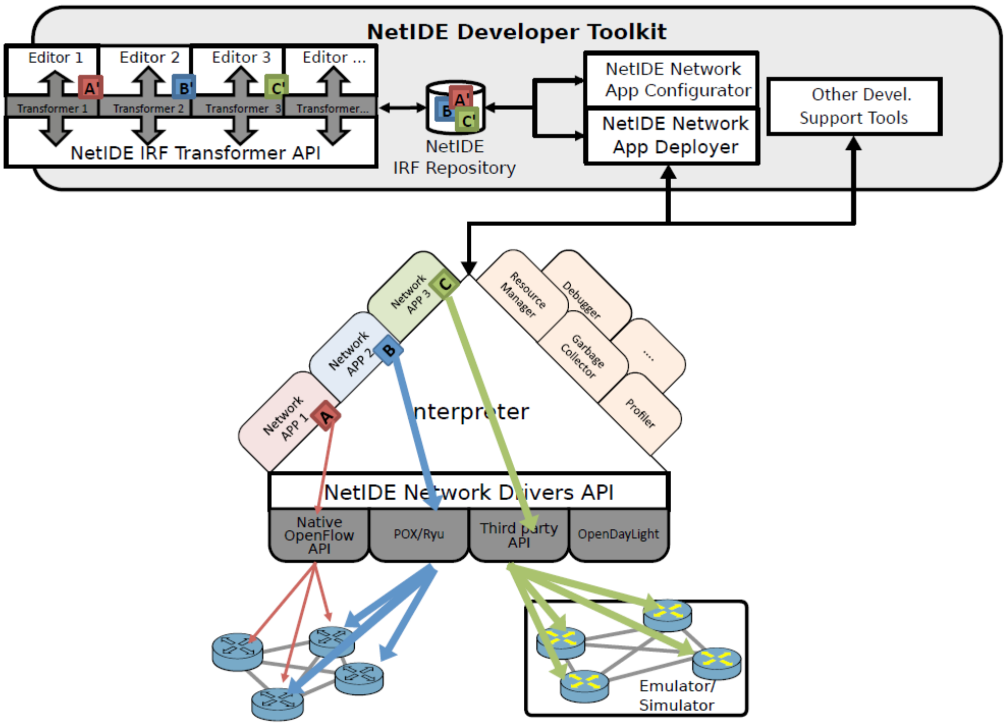

Figure 15. The NETIDE approach to SDN Controller Development

NetIDE is focused on delivering a single IDE to support the whole development lifecycle of network controller programs in a vendor-independent fashion as shown in the figure above. The project will deliver:

- A controller- and gear-independent approach to support the development of Network Apps
- Integrated development and tool support for the network design cycle in SDN environments
- New network layer services and the support for emulator-in-the-loop and simulator-in-the-loop configuration

2.10.1. Overall Architecture

The initial version of the proposed NETIDE architecture is shown in Figure 16. The high level architecture of NETIDE. It is proposed that Pyretic will be the development language for the abstracted controller which will be integrated with an NETIDE Interpreter via an Abstract API (IRF). The NetIDE Interpreter adds the targeted controller specific components such as REST API's etc. Pyretic is one member of the Frenetic family of SDN programming languages (also under development is Frenetic-OCaml: embedded and implemented in OCaml). Pyretic enables network programmers and operators to write succinct modular network applications by providing abstractions. Pyretic is both a programmer-friendly domain-specific language embedded in Python and the runtime system that implements programs written in the Pyretic language on network switches.

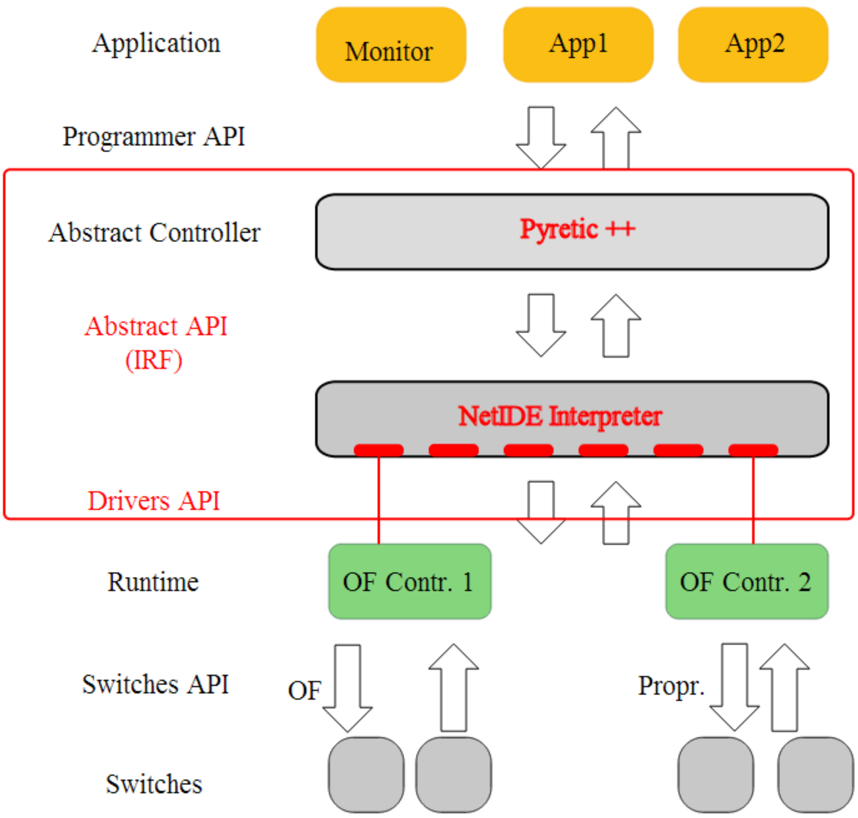

Figure 16. The high level architecture of NETIDE

They expected key outputs NETIDE are as follows:

- Define an Intermediate Representation Format (IRF) – and associated APIs to support the development of SDN solution-independent network applications
- Deliver a prototype IDE and associated tools that support the SDN development lifecycle based on IRF
- Develop a prototype of a run-time environment for IRF based applications including an Network App Engine that supports open & proprietary SDN controllers, and allows for emulator/simulator-in-the-loop

There is strong correlation with NETIDE and T-NOVA's task 4.3, which is focused on the development of an SDK for SDN. For this purpose, alignment between the two project will be sought, via synergies that have already started (e.g. workshops) and via partners that are members of both consortia.

2.11. Consolidation and Discussion

The previous sub-sections surveyed a number of currently running projects (i.e. MCN, CONTENT, UNIFY and NetIDE), industry frameworks and solutions (i.e. CloudBand, HP OpenNFV, and Qosmos/INTEL/TIETO) as well as efforts from Standardisation Bodies (i.e. ETSI NFV ISG). It is important to say that T-NOVA partners are participating or at least follow closely those activities.

For T-NOVA the activities in the frame of ETSI NFV ISG are of primary importance. T-NOVA architecture (to be discussed in the following section) intends to align to the one proposed by the ETSI NFV ISG. Moreover, the ETSI architecture is also a general guide for most of the solutions, platforms and projects discussed.

Each project/platform/standard surveyed provided valuable inputs, specifically:

- ETSI NFV ISG
  - NFV reference architecture
  - Component functionalities
  - Interface definition
- TMF
  - Definition of the new virtualized operations environment (TMF ZOOM)
  - ETSI NFV MANO Gap Analysis
  - ETSI NFV MANO Relevant Specifications
- CloudNFV, CloudBand, HP OpenNFV and Qosmos/Tieto/Intel
  - Mapping of reference architecture to available SW/HW components/platforms
  - Service Lifecycle
  - Distributed Cloud architecture to support NFV deployment
  - High value VNFs for telcos
  - Key architectural components and interfaces
- MCN, CONTENT, UNIFY, NetIDE EU Projects
  - Exploitation of Cloud for deployment of telco related functionalities
  - Heterogeneous networks virtualisation
  - End-to-end Cloud and Network Orchestration
  - Interconnection of Telco Cloud and Carrier Networks
  - Software abstraction and Programmability for SDN Controllers

The above background information has been discussed thoroughly in the respective Tasks of WP2 (2.3, 2.4, 2.5, 2.6) in order to provide guidelines and influence the design of the overall T-NOVA architecture produced within Task 2.2 and presented in the next chapter.

3. T-NOVA OVERALL ARCHITECTURE

3.1. Key Features of the T-NOVA System and Architecture Principles

Deliverable D2.1 (System Use Cases and Requirements) presented and analysed a set of high-level system requirements based on different use cases of the T-NOVA system. If a consolidation of all these high-level system requirements can be attempted, these can be distilled down to the following five key features/functionalities, which need to be implemented by the T-NOVA system:

F1. IT infrastructure virtualisation, supporting resource elasticity

IT assets are necessary for the hosting of VNFs. These assets need to be virtualised in order to be provided as-a-Service, and the associated resources must be scalable in order to optimise usage.

F2. Network infrastructure virtualisation, supporting resource elasticity

Network connectivity is required for both the interconnection of the VNF/VNFCs among the various NFVI PoPs as well as for the customer's access points. Network links need to be virtualised in order to be provided as-a-Service, and the associated resources must be scalable in order to optimise usage.

F3. Automated service provisioning, monitoring, scaling and optimisation

Although an NFV service can be manually deployed, a key feature of T-NOVA is service automation. Following deployment, the service will need to be monitored, scaled and optimised also in an automated manner.

F4. Service advertisement and brokerage

In order to increase the benefits from the automated service provisioning for the customer, a service advertisement as well as a Brokerage framework needs to be in place. The latter is required to facilitate the involvement of multiple actors in the T-NOVA ecosystem.

F5. SLA, billing and accounting support

SLA, billing and accounting support is expected to accelerate market uptake of the T-NOVA system since it enables to actually generate revenue from its deployment and operation.

In addition to the aforementioned key features, in order to maximise the impact of T-NOVA results and accelerate the market uptake of the project achievements to the highest possible extent, we identify the following five high-level architectural principles which need to be respected during the architecture design and implementation phase:

P1. High TRL / Short time-to-market

NFV is generally considered a close-to-market technology, and T-NOVA project outcomes target to TRLs equal to or higher than 6, meaning that at the end of the project the system should be stable and well operational in a lab environment under realistic traffic/load conditions, fulfilling all functional and non-functional requirements.

P2. Openness (architecture, interfaces)

Adopting an open architecture with open interfaces based on industry standards maximises cross-vendor interoperability and maximises project impact.

P3. Modularity and Flexibility

A modular design based on independent components allows multiple alternative configurations (including partial deployment), making the system flexible so as to fulfil diverse actors' needs.

P4. Resilience

In order to meet (or even approach) carrier-grade requirements, the system should quickly recover from faults, outages and malfunctions.

P5. Affordability

The procurement, application and deployment of the T-NOVA solution as a management framework should involve minimal CAPEX investment. For this purpose, it should adapt as much as possible to existing IT and network infrastructures.

3.2. T-NOVA Architecture Overview

Based on the required functionalities as well as architectural principles as mentioned in the previous section, it is possible to derive an overall, high-level view of the architecture of the T-NOVA system.

The first step should be to map the aforementioned key features to functional blocks.

F1. IT infrastructure virtualisation, supporting resource elasticity

State-of-the-art data-centre IT virtualisation technologies, including modern cloud platforms can fulfil most of the functionalities required for T-NOVA. The typical IT virtualisation structure can be followed, based on *compute*, *storage*, DC *network* and *hypervisor* domains. On top of these, a unified management framework conforming to contemporary cloud management paradigms can be foreseen, where the compute/storage/DC network and hypervisor domains are jointly managed achieving the automated provision, management and optimisation of IT IaaS services. In order to align with ETSI terminology, we call this management entity *Virtualised Infrastructure Management (VIM)*. A VIM manages and controls the infrastructure of a data centre in which VNFs are deployed. The latter is called *NFVI Point of Presence (NFVI-PoP)*

F2. Network infrastructure virtualisation, supporting resource elasticity

Intra-data-centre network assets are controlled by the *VIM*, as described in the previous paragraph. Contemporary Software Defined Network (SDN) technologies significantly facilitate software-driven and vendor-agnostic network management and are thus commonly employed in modern data centres, so it is realistic to assume that the intra-DC network is SDN-enabled. For the wide-area network (hereafter *Transport Network*), a separate management entity needs to exist, managing end-to-end connectivity between DCs/NFVI-PoPs and managing traffic steering to establish the end-to-end network service. We identify this management entity as *Transport Network Management (TNM)*.

F3. Automated service provisioning, monitoring, scaling and optimisation

The establishment of an end-to-end T-NOVA service normally requires the knowledge of the available resources, the planning of the service and the interaction with the VIMs and the TNMs managing the infrastructure assets which will be involved. All these operations need to be carried out by a higher-level management entity, one per Service Provider, which jointly orchestrates the underlying infrastructure. This is the role of the *T-NOVA Orchestrator*, which should incorporate functional modules dealing with *Resources Orchestration, Network Service Orchestration* as well as *VNF Management*. In order to fulfil its role, the Orchestrator should also maintain internal catalogues containing information about available and established Network Services, available VNFs and deployed instances of both, as well as infrastructure resources. At the same time, a Repository for hosting the VNF images and associated metadata should be foreseen (*NF Store)*, from where the VNF images are retrieved for deployment as instances into the infrastructure.

F4. Service advertisement and brokerage

A dedicated, customer-facing module should be foreseen, dedicated to *Brokerage* functionalities. Apart from pricing and trading policies, the Brokerage function should have access to Service and VNF catalogues as well as to the NF Store.

F5. SLA, billing and accounting support

In the same context, dedicated modules should also be foreseen for i) *Accounting* ii) *SLA Management* and iii) *Billing*. These components should collect resource usage data, monitor the status of the established SLAs and bill the customer accordingly. All service-related information can be presented through a unified *Dashboard*, which can also facilitate interactions with service brokerage procedures.

By assembling all the aforementioned components, which were identified as necessary, a high-level view of the T-NOVA system architecture can be derived, as shown in Figure 17.

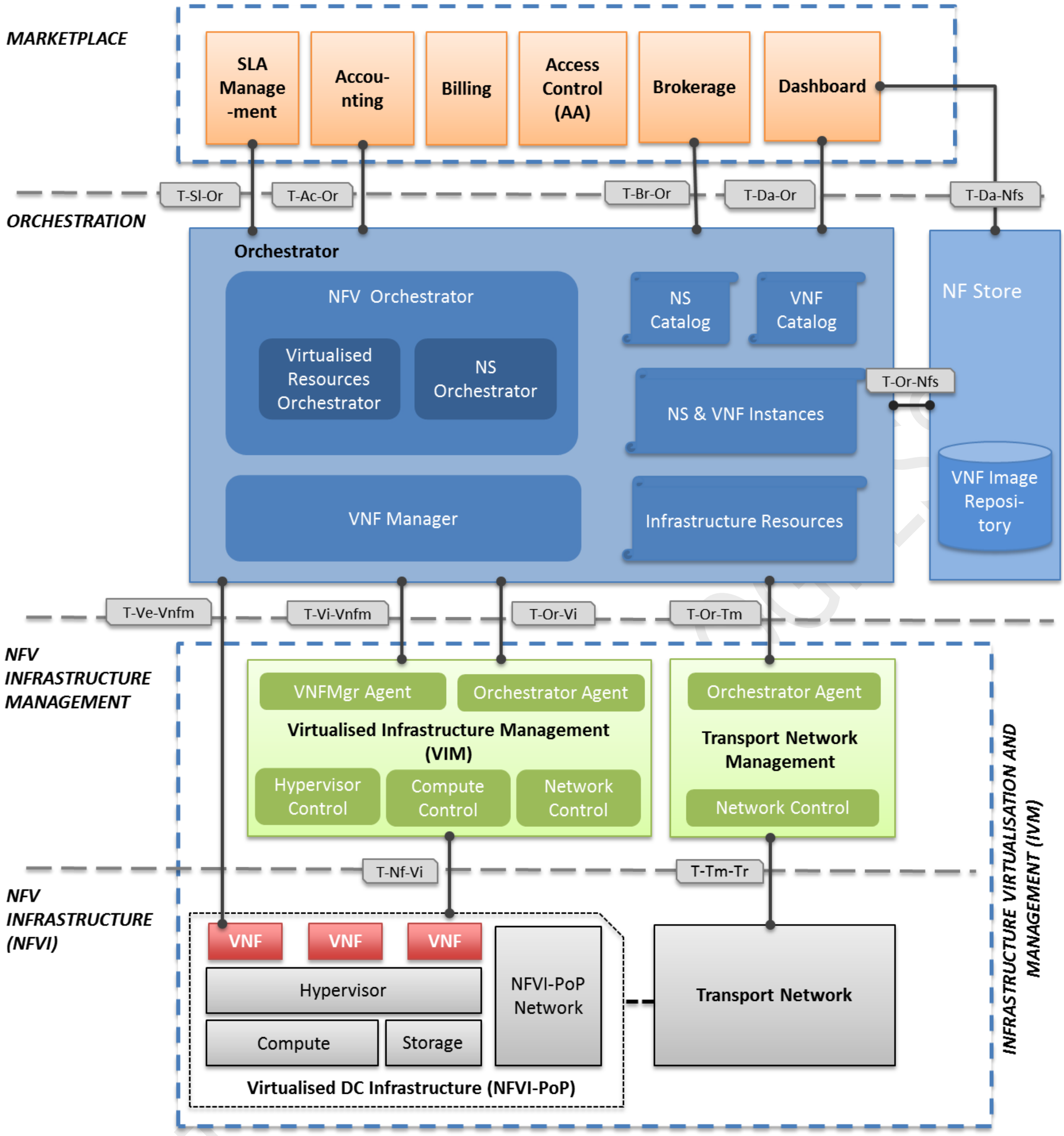

Figure 17. High-level view of overall T-NOVA System Architecture

The T-NOVA architecture can be hierarchically organised into four architectural layers:

- The *NFV Infrastructure (NFVI)* layer includes the physical and virtual nodes (commodity servers, VMs, storage systems, switches, routers etc.) on which the services are deployed.
- The *NFVI Infrastructure (NFVI) Management* layer comprises the infrastructure management entities (VIM, TNM). In the sections as well as the deliverables to follow, the NFVI and management layers are conceptually grouped under the name *Infrastructure Virtualisation and Management (IVM).*
- The *Orchestration* layer is based on the T-NOVA Orchestrator and also includes the NF Store

- Finally, the *Marketplace* layer contains all the customer-facing modules which facilitate multi-actor involvement and implement business-related functionalities.

The sections to follow present a first overview of the aforementioned layers and their components, while a more detailed description is to be found in Deliverables D2.31 and D2.41.

With regard to interfaces, it must be noted that in 7 mostly depicts reference points – which may contain one or more actual interfaces- between architectural layers. Each reference point label starts with "T-" to differentiate from interfaces defined in ETSI NFV ISG documents (and in specific Vi-Vnfm, Or-Vi, Ve-Vnfm, Nf-Vi) – although in many cases the functionality of the reference point will be almost aligned to the ETSI definitions.

Following this initial architectural overview, Table 1 suggests how the T-NOVA approach will effectively fulfil the architectural principles identified in the previous section.

Table 1. Fulfilment of architectural principles

Architectural Principle	Proposed Approach
P1. High TRL / Short time-to-market	Rely on established and well-tested platforms as basis for developing some functionalities (e.g. VIM, TNM, account/billing etc.), to achieve rapid prototyping.
P2. Openness (architecture, interfaces)	Release architecture and interface specifications as public domain and rely on widely used communication protocols (e.g. HTTP REST) and data formats (e.g. XML, JSON)
P3. Modularity and Flexibility	Develop each entity as an independent, self-contained software module. Allow multiple architectural variants by defining several options in the information models.
P4. Resilience	Exploit technologies from the cloud domain for VM migration. Exploit SDN management for rapid traffic redirection. Establish a multi-level proactive monitoring framework.
P5. Affordability	Rely on and extend/build open-source software which can be installed on commodity servers. Ensure as wide as possible compatibility with existing infrastructure (e.g. support both SDN and non-SDN network elements) in order to minimise infrastructure upgrade costs.

3.3. High-level Description of T-NOVA Main Architectural Entities

3.3.1. Marketplace

T-NOVA introduces the concept of Marketplace in the Network Function Virtualization framework, with the aim of promoting the VNF service offerings and facilitating the commercial activity and fluent interaction among the different business stakeholders interacting with the T-NOVA system, which were previously identified when exploring NFV use cases in Deliverable D2.1.

The T-NOVA marketplace is a distributed platform placed on top of the orchestration level, which besides providing the Graphical User Interface, it supports the following general functions for -NOVA stakeholders:

- The Customers will be able to browse and select VNFs service offerings that best match their needs, as well as negotiate SLAs and exchange their billing information with the Service Provider, keeping track of all the services purchased.
- The Service Provider will be able to acquire VNFs, interacting with different Network Function suppliers by means of a brokerage/trading module in order to compose network services as bundles of VNFs not only for its own needs but also to offer the composed Network Services to its customers. Therefore, also SLA and billing information between SP and FPs will be managed.
- Several Network Function developers (Function Providers) will be able to publish their VNF to trade them by means of T-NOVA marketplace.

Based on the T-NOVA requirements gathered in Deliverable D2.1 we identify that the modules in 8 should be present at marketplace level:,

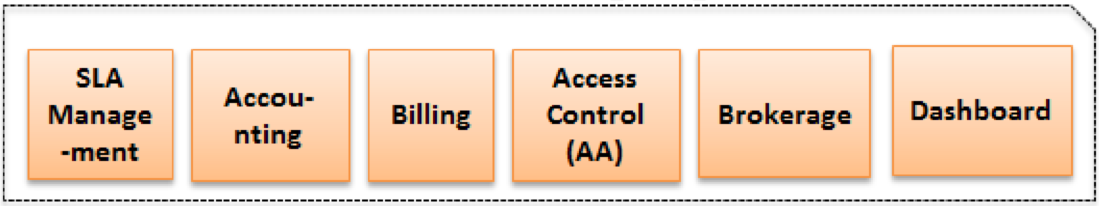

Figure 18. Marketplace components

- **Access control**: this component provides authentication and authorization functionalities to manage the access to T-NOVA by the different stakeholders. In other words, it regulates who is allowed to access the T-NOVA system and what is it allowed to do.
- **Accounting module**: this module is in charge of registering all business relationships (subscriptions, SLA evaluations and usage) and making the related information available for the billing system.
- **Dashboard**: it is the graphical front-end displaying monitoring/utilization/status information about the provisioned T-NOVA services. It will have one different view for each of the stakeholders accessing the T-NOVA marketplace.
- **Brokerage module**: it is the component that receives the customer requests, and will present at the customer the most suitable offerings that matches his/her requirements; depending on the applicable trading-policies the necessary actions to get the best price for each service+SLA when creating a new service starting from the VNFs will be carried out.

- the brokerage module will be connected to a service catalogue store at marketplace level that will store all the portfolio of offerings (service description, SLA, price) available in the T-NOVA marketplace.

- **SLA management module**: it is the component that will register all the SLA agreements among the involved parties, checks if the SLAs have been fulfilled or not, and informs the accounting system for the pertinent billable items.
- **Billing system**: It is the component that produces the bill for a customer on behalf of the Service Provider (SP). A bill might also be produced for the SP on behalf of its own suppliers (FPs).

Further information about the functionalities and internal architecture of each marketplace module, as well as the internal marketplace interfaces among these modules are defined under T2.6 and will be explained in Deliverable D2.41.

As shown in 9, the marketplace is connected on one hand to the orchestrator by means of 4 external interfaces (see section 3.5.1), and the dashboard is connected to the Function Store to allow Function Providers to directly interact with this module.

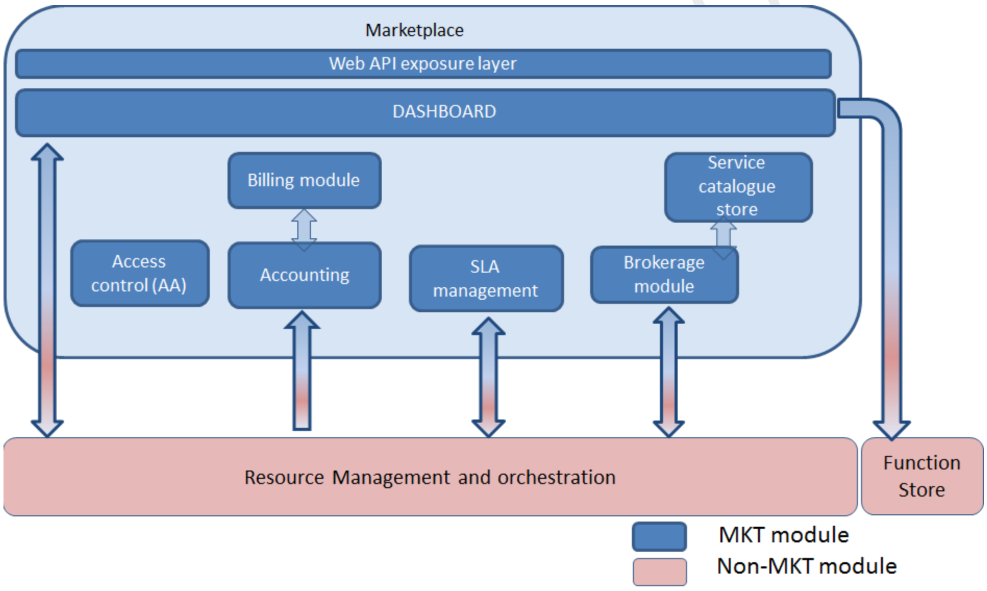

Figure 19. Marketplace external interfaces

3.3.2. Orchestrator

The T-NOVA Orchestrator is the core component of the T-NOVA architecture framework and it aims to address Network Services (NSs) and Virtual Network Functions (VNFs) lifecycle management operations over distributed and virtualized network/IT infrastructures.

3.3.2.1. Reference Architecture

The Orchestrator reference architecture, as well as the interfaces with the external Functional Entities (FEs) is depicted in 20. In detail, the orchestrator interacts with the Marketplace, which is the FE responsible for establishing the business and operational management of T-NOVA. Besides the Marketplace, the Orchestrator also interfaces with the Virtualized Infrastructure Management (VIM), for managing the data center network/IT infrastructure resources, as well as with the Transport Network Manager (TNM) for the WAN elements connectivity management. Finally, the Orchestrator interacts with the VNF itself, which is located in the T-NOVA Infrastructure Virtualized Management (IVM) layer, to ensure its lifecycle management.

Internally, the T-NOVA Orchestrator is composed by two main elements and a set of repositories. One of the core elements is the NFV Orchestrator (NFVO), acting as the front-end with the Marketplace and orchestrating all the incoming requests towards the other elements of the architecture. Further details about the NFVO and the associated incoming requests are given in section 3.3.2.2.1. To support the NFVO operation procedures, a set of repositories are defined to describe the available VNFs and NSs, the instantiated VNFs and NSs, as well as the available resources in the virtualized infrastructure. Further details about the orchestrator repositories are provided in section 3.4.2.3.3. Finally, the NFVO also interacts with the VNF Manager, responsible for the VNF-specific lifecycle management procedures, as described in section 3.4.2.3.2.

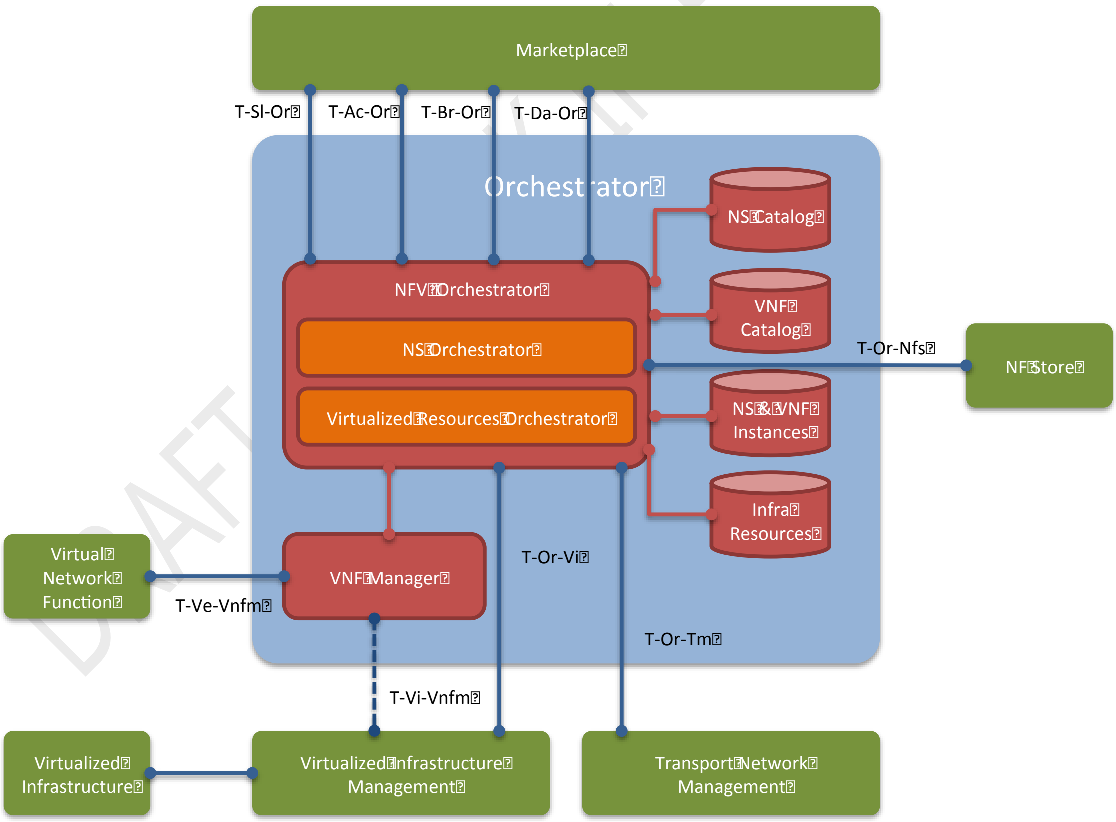

Figure 20. T-NOVA Orchestrator Reference Architecture and Interactions

3.3.2.2. Functional Entities High-level Description

<Should include brief introduction>

3.3.2.2.1 NFV Orchestrator

The main function of the NFV Orchestrator (NFVO) is to manage the virtualized Network Services (NSs) lifecycle procedures. Since the NSs are composed by Virtual Network Functions (VNFs) and Physical Network Functions (PNFs), the NFVO is able to decompose each NS into the constituent VNFs and PNFs. Nevertheless, although the NFVO has the knowledge of the VNFs that compose the NS, it delegates their lifecycle management to a dedicated FE designated by VNF Manager (VNFM).

Furthermore, besides orchestrating the virtualized service level operations, therefore abstracting the service specificities from the business/operational level – in this case the T-NOVA Marketplace – the NFVO also manages the virtualized infrastructure resource level operations. Hence, it coordinates the resource allocation to specific NSs and VNFs according to the availability of the virtualized infrastructures, also known as data centers.

To address the two main functionalities above mentioned, the NFVO is architecturally split in two modules, namely the Network Services Orchestrator (NSO) and the Virtualized Resources Orchestrator (VRO), further described below.

Network Services Orchestrator

The Network Service Orchestrator (NSO) is one of the components of the NFVO with the responsibility to manage the NS lifecycle management procedures (e.g. on-boarding, instantiation, supervision, assurance, scaling and usage accounting). Towards this objective, it interacts with the T-NOVA Marketplace (northbound interface) external entity and uses and with the following T-NOVA Orchestrator internal components:

- **Repositories**: collect information (e.g. deployment descriptors, dependencies, software images, SLAs, etc.) about the NSs/VNFs and storage of NSs/VNFs instances;
- **Virtualized Resources Orchestrator (VRO)**: request management actions related with the virtualized resources, either within the data center scope (e.g. compute, storage and network) and/or on the transport network segment;
- **Virtual Network Function Manager (VNFM)**: request of lifecycle management actions related with the VNFs.

Virtualized Resources Orchestrator

The Virtualized Resources Orchestrator (VRO) is the resource layer management entity of the NFVO main block. Its main responsibility is to orchestrate/manage the virtualized infrastructure resources distributed across multiple data centers, in particular, map the incoming NS requests to the virtualized infrastructure resources, as well as coordinating the resources allocation and placement for each Virtual Machine (VM) that composes the VNF (and the NS). Moreover, the VRO is also the entity interacting with the WAN elements for connectivity management purposes.

The following virtualized resources are managed by the VRO:

- **Compute**: virtual processing CPUs and virtual memory;

- **Storage**: virtual storage;
- **Network**: virtual links intra/interconnecting VNFs within the Data Center Network (DCN).

The VRO interacts with the following blocks:

- **Network Services Orchestrator (NSO)**: receive resources reservation/allocation requests for a specific NS, for all the constituent VNFs;
- **Repositories**: collect information about the VNF software components and resources, as well about the infrastructure resources availability;
- **Virtualized Infrastructure Manager (VIM)**: enforce resource reservation/allocations and collect monitoring information about the virtual links interconnecting the VNFs;
- **Transport Network Manager (TNM)**: enforce resource decisions allocations and collect monitoring information about the non-virtualized transport network elements;
- **Virtual Network Function Manager (VNFM)**: receive resources reservation/allocation requests.

3.3.2.2.2 Virtual Network Functions Manager (VNFM)

The VNF Manager (VNMF) is responsible for the lifecycle management of the VNF. This includes the following:

- Instantiate: create a VNF on the virtualized infrastructure using the VNF on-boarding descriptor;
- Configure: configure the instantiated VNF with the required information to start the VNF (can already include some customer-specific attributes/parameters);
- Scale: increase or decrease the VNF capacity by adding/removing VMs (out/in horizontal scaling) or adding/removing resources from the same VM (up/down vertical scaling);
- Update: modify configuration parameters;
- Upgrade: change software supporting the VNF;
- Terminate: release infrastructure resources allocated for the VNFs.

The VNFM interacts with the following components:

- **Network Services Orchestrator (NSO)**: receive VNF instantiation requests for a specific NS and provide VNF monitoring information;
- **Repositories:** collect information about the VNF internal components and resources;
- **Virtualized Resources Orchestrator (VRO)**: request reservation/allocation/release of virtualized infrastructure resources for the VNF;
- **Virtual Network Function (VNF)**: configure VNF specific information and receive VNF related monitoring information;

- **Virtual Infrastructure Management (VIM)**: collect monitoring information about the virtualized infrastructure resources allocated to the VNF.

3.3.2.2.3 Repositories and Catalogues

To support the T-NOVA Orchestrator lifecycle management operations, the following catalogues are defined:

- **NS Catalogue:** represents the repository of all the on-boarded NSs in order to support the NS lifecycle management:
  - NS Descriptor (NSD): contains the service description, including SLAs, deployment flavours, references to the virtual links (VLDs) and the constituent VNFs (VNFFG);
  - Virtual Link Descriptor (VLD): contains the description of the virtual network links that compose the service (interconnecting the VNFs);
  - VNF Forwarding Graph Descriptor (VNFFG): contains the NS constituent VNFs, as well as their deployment in terms of network connectivity;
- **VNF Catalogue:** represents the repository of all the on-boarded VNFs in order to support its lifecycle management;
  - VNF Descriptor (VNFD): contains the VNF description, including its internal decomposition in Virtual Network Function Components (VNFCs), deployment flavours and references to the virtual links (VLDs);
  - Software images of the VMs located in the IVM layer;
- **NS & VNF Instances:** represents the repository of all the instantiated NSs and VNFs, which can be updated/released during the lifecycle management operations;
- **Infrastructure Resources:** represents the repository of the available/reserved/allocated NFVI resources as abstracted by the VIM across operator's infrastructure domains. Furthermore, it also includes the resources available/reserved/allocated in the WAN segment.

3.3.3. Infrastructure Virtualisation and Management

The concept of virtualisation has been a cornerstone in the rapid evolution and adoption of cloud computing. This model of abstracting the physical compute resources into virtual resources has been broadly adopted in the IT domain and is now also being embraced in the networking domain, by introducing Virtual Network Functions (VNFs). Application workloads run on top of hardware thanks to a hypervisor layer that decouples software and hardware through partitioning hardware and share it among different applications. For network applications such as the deployment of VNFs careful consideration must be given to all elements of the infrastructure environment to ensure the required performance particularly for intensive packets processing workloads or applications that support large numbers of simultaneous connections. In this context the availability of certain features within the

CPU, driver performance, efficient virtualisation of resources, physical distribution of virtual nodes etc. can have a significant influence on performance.

For this reason, the T-NOVA Infrastructure Virtualisation Management (IVM) domain (consisting of the NFVI and NFVI Management layers) will comprise a mixture of physical and virtual nodes and will be used to develop, implement and showcase T-NOVA services. The IVM will be fully integrated with the T-NOVA Orchestrator to ensure that requirements for the deployment and lifecycle management of T-NOVA VNFs can be carried out in an appropriate and effective manner. The IVM will be flexible enough to support a variety of use cases beyond those explicitly identified in T-NOVA (see Deliverable D2.1). As mentioned previously, infrastructure virtualization plays a key role in achieving this vision in T-NOVA. Virtualisation and management of the virtualised resources extends beyond the compute and storage to include network infrastructure in order to fully exploit the capabilities of the T-NOVA architecture.

3.3.3.1. IVM Architecture

Using both the NFVI and MANO architectures as references the proposed architecture for the T-NOVA IVM is presented in 1.

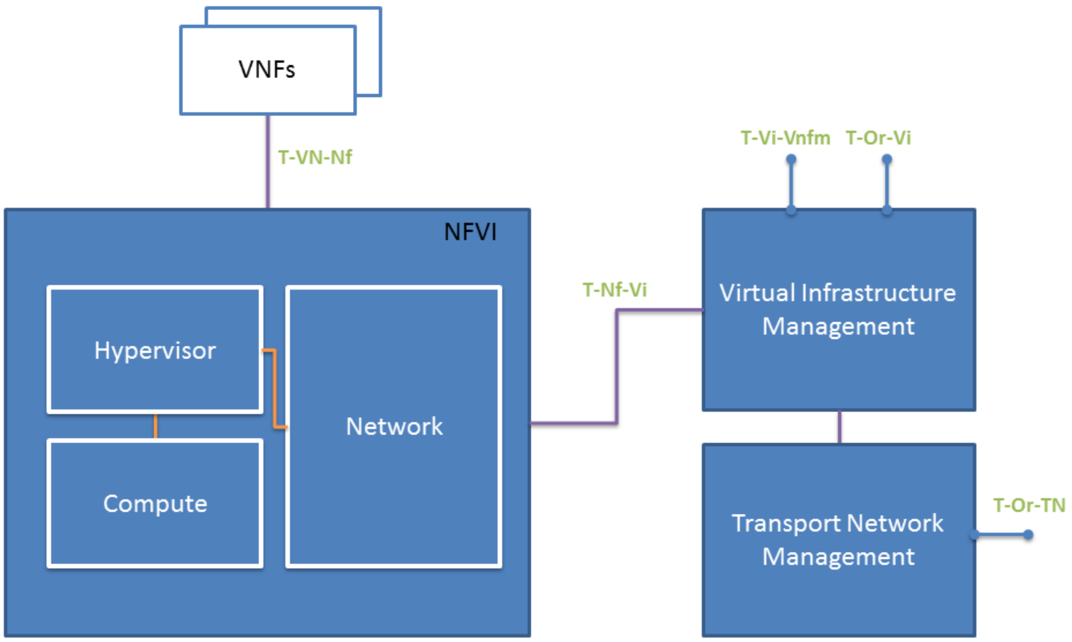

Figure 21. High-level architecture of T-NOVA IVM.

For the T-NOVA architecture the decision was taken to separate the *VIM* function from the Orchestration layer, whereas ETSI considers management and orchestration (MANO) as a single domain. The reason is twofold: first, there already exist several quite mature technological frameworks which can implement most of the VIM functionalities – which is not the case for the Orchestrator. Second, it is a matter of cardinality; the T-NOVA architecture considers a single Orchestrator but multiple VIM instances, one per NFVI-PoP.

The VIM acts in unison with the NFVI under the control of the Orchestrator through two specific interfaces:

- Virtual Network Function Management – VIM Interface (T-Vi-Vnfm)
- Orchestrator –VIM Interface (T-Or-Vi)

These interfaces enable the Orchestrator and its VNF Manager to manage the VIM and subsequently the resources under the control of the VIM.

The T-NOVA architecture, following the ETSI WG INF (focused on the specification of the NFVI), identifies three domains in order to disaggregate the complexity of the required capabilities into logical functional domains. The domains are as follows:

- *Compute domain*, which operates at the lowest level, also in the computing and storage slices. It comprises the generic high volume servers and storage. The underlying physical elements are abstracted by the hypervisor allowing aggregation of these resources across many discrete servers and assignment of them to VNF's.
- *Hypervisor domain*, which operates at a virtual level, encompassing the computing and storage slices. It provides abstraction of the hardware to the VNF. This supports capabilities such as portability and scalability of the VNF. The hypervisor also is responsible for the allocation of the compute domain resources to the VMs and also provides a management interface to the orchestration and management system which supports the loading and monitoring of VMs and VNF. The hypervisor is also responsible for network connectivity between VM's hosted on the same physical server as well as those hosted on different physical servers.
- *Network domain*, which operates both at virtual level and the hardware level, of the Data Centre's (NFVI-PoP) network slice. It comprises all the generic high volume switches interconnected into a network which can be configured to supply infrastructure network services.

In addition to the VIM, the analysis of the system requirements of T-NOVA, including the needs of interconnecting distributed NFVI-PoPs and establishing end-to-end services, highlight the necessity for a *Transport Network Management* (TNM) capability for the set-up, teardown and management of network traffic between points of presence (PoP) e.g. inter-data centre connectivity supporting VLAN tunnelling over WAN. Released specifications to date by ETSI have not identified this as an explicit capability as their focus was on virtualised network capabilities. However they do include an interface (i.e. Ex-Nf) which acts as an interface between the infrastructure network domain and any existing non-virtualised network. The T-NOVA TNM domain implements and interface to the Orchestrator in order to obtain appropriate management of the TNM.

A detailed description of the function, purpose and requirements for each domain will be presented in Deliverable D2.31.

3.4. High-Level Description of Interfaces

This section presents an overview of the main reference points of the T-NOVA architecture, their purpose and the information which is exchanged. The relevant reference points of the ETSI NFV architecture are also identified, although the correlation/overlap between T-NOVA and ETSI may vary.

3.4.1. Marketplace – Orchestrator Interfaces

As aforementioned, there are four interfaces between the Marketplace and the Orchestrator. A first approach to the information to be exchanged by means of these interfaces is provided below. Further details and requirements for these interfaces are carried out in tasks T2.3, T2.5 and T2.6, and will be included in deliverables D2.31 and D2.41.

These interfaces are summarized in Table 2 and they all have some partial overlap with the ETSI Os-Ma interface, which connects the MANO and OSS/BSS domains.

Table 2. Interfaces between the T-NOVA Orchestrator and the Marketplace

Interfacing entities	T-NOVA Reference Point name	ETSI NFV Relevant Reference Point	Description and Comments
Dashboard-Orchestrator	T-Da-Or	Os-Ma	This Dashboard sends the Orchestrator all the changes in service parameters. The Orchestrator sends the Dashboard all the service monitoring information.
SLA management module-Orchestrator	T-Sl-Or	Os-Ma	The SLA management module sends the Orchestrator the SLA agreement. The Orchestrator communicates SLA-related metrics to the management module so that the latter can check SLA compliance
Accounting module-Orchestrator	T-Ac-Or	Os-Ma	The Orchestrator sends the Accounting module all the resource usage information.
Brokerage module-Orchestrator	T-Br-Or	Os-Ma	The Brokerage module sends the Orchestrator all the service compositions information. The Orchestrator sends the Brokerage module information on all the available VNFs.

3.4.1.1. Dashboard-Orchestrator (T-Da-Or)

This interface will convey the information that a given SLA is ready to be applied, together with its parameters (from the SLA management module to the Orchestrator), or has been breached (from the Orchestrator to the SLA management module). We emphasize that, since the Orchestrator has to follow the SLA related metrics, in order to try to escalate the service and keep the SLA, the SLA management module will only be notified in the case of a SLA breach.

3.4.1.2. SLA management module-Orchestrator (T-Sl-Or)

This interface will convey the information that a given SLA is ready to be applied or has been breached (from the Orchestrator to the SLA management module), as well as that SLA agreement (its parameters -- from the SLA management module to the Orchestrator). We emphasize that, since the Orchestrator has to follow the SLA related metrics, in order to try to escalate the service and keep the SLA within agreed levels of service, the SLA management module will only be notified in the case of a SLA breech.

3.4.1.3. Accounting module-Orchestrator (T-Ac-Or)

This interface will convey all the resource usage information (from the Orchestrator to the Accounting module), so that the Customer can later be billed accordingly. No information is expected to flow from the Accounting module to the Orchestrator.

3.4.1.4. Brokerage module-Orchestrator (T-Br-Or)

This interface will convey all the service composition information (from the Brokerage module to the Orchestrator), as well as the information about the VNFs available for composing a new Network Service (from the Orchestrator to the Brokerage module).

3.4.2. Marketplace – Function Store Interface

T-Da-NfS: this interface will convey the information provided in the Dashboard in order to publish a new VNF in the NF Store as well as the modification and withdrawal of the VNFs.

3.4.3. Orchestrator – Network Function Store Interfaces

The interface between the Orchestrator and FS (T-Or-NFs), supports a set of key functions within T-NOVA. The interface has no mapping to the ETSI MANO architecture, as the FS is a T-NOVA specific entity not considered in ETSI MANO. This interface allows the orchestrator to be notified about VNFs that have been recently added, removed, or modified in the FS. Furthermore, it also allows the orchestrator to retrieve VNF related information, such as virtual machine images, VNF metadata, including name, id, pricing information, requirements and VNF capabilities.

The Orchestrator interacts with the Network Function Store for gathering information about the VNF and gets the images and metadata description of the VNF that are made available to the system.

The interface is further described by T2.3 for the Orchestrator side and by T2.5 for the NF Store side, in deliverables D2.31 and D2.41, respectively.

Hereunder a summary of the primitives is presented.

Interface T-Or-Nfs:

Orchestrator side:

- Notification of modification in the NF Store
- Delivery of VNF image and VNF metadata
- Removal/Deletion of the VNF from the NF Store

Function Store side:

- Request for delivering the new or updated VNF image and VNF metadata.

Table 3. Interfaces between the T-NOVA Orchestrator and the FS

Interfacing entities	T-NOVA Reference Point name	ETSI NFV Relevant Reference Point	Description and Comments
NFV Orchestrator –Function Store Interface	T-Or-Nfs	None	This interface allows the NFV Orchestrator to receive VNF advertisements (new VNF added to FS, VNF removed from FS, VNF modified) and to also receive VNF related information (e.g. virtual machine images, VNF metadata, including name, id, pricing information, requirements and VNF capabilities)

3.4.4. Orchestrator – VNF Interfaces

The interface between the Orchestrator and VNF supports a set of key functions within T-NOVA. The Ve-Vnfm interface directly maps to the interface identified in the ETSI NFV architecture, as outlined in Table 4. On the one hand, this interface allows the VNFM to configure the VNF and possibly perform a set of control operations (e.g. start, stop) as well as other VNF lifecycle related operations. On the other hand it allows the VNFM to deliver detailed information on the status (e.g. running state, error state) and performance of the VNF (function/VM level information) so that it can act accordingly to the SLA in place.

The Orchestrator directly interacts with the VNFs during their running phase. This interaction happens in the "Management" and "Termination" status of the VNF lifecycle.

The services provided with this interface are summarized hereunder:

Management status

- set-up: initialization of the VNF, e.g. configuration of the VNF network interfaces
- start and stop: request to the VNF to start or stop providing the service.
- Monitoring: the VNF can provide to the orchestrator monitoring information useful to the orchestrator for deciding about scaling the VNF or the service.

Termination status

- Terminate: request to release the resources allocated to the VNF and shut-down of the VNF itself

Table 4. Interfaces between the T-NOVA Orchestrator and VNFs

Interfacing entities	T-NOVA Reference Point name	ETSI NFV Relevant Reference Point	Description and Comments
Virtual Network Function Manager–Virtual Network Function Interface	T-Ve-Vnfm	Ve-Vnfm	This interface allows the VNF Manager to configure the VNF, collect monitoring/performance data, and be notified about faults within the VNF.

3.4.5. Orchestration Layer – VIM Interface

The interfaces between the Orchestration Layer and VIM support a number of key functions within the T-NOVA. As shown in the IVM high level architecture section 3.3.3, two specific interfaces have been identified and map to the interfaces identified in the ETSI NFV Architectural Framework as outlined in Table 5. The first one is the Virtual Network Function Management – VIM Interface (T-Vi-Vnfm) and is responsible for the exchange of infrastructure monitoring information allocated to the VNF either through explicit request by the VNF Manager or through periodic reporting initiated by the VIM. The types of data exchanged over this interface include detail information on the status, performance and utilisation of infrastructural resources such CPU, storage, memory etc. Data will also encompass networking information relating to a specific a VNF such as NIC level network traffic from the hosting VM or inter VM network traffic if a VNF is deployed across more than one VM. Finally VNF performance data will also be exchanged over this interface (In the case of performance data its builds on monitoring data with the addition of context such as the expected operational target for a service, or involves utilising data from various levels within VNF software to understand what is happening with a particular service e.g. performance may relate to end-to-end actions of a VNF service such as (secure) VM-to-VM communications, QoS etc. VNF performance may also be related to effectiveness of configuration changes or optimisations relative to the baseline performance of a VNF service. Collectively the data will be used by the VNF Manager within the T-NOVA Orchestrator to track VNF service performance and to ensure SLA compliance.

The second interface identified is the Orchestration – VIM interface (T-Or-Vi). This interface is responsible for handling requests from the NFV Orchestrator with respect to the full life cycle of a NS service. Typical examples of requests sent over this interface would include instantiation (preceded by on-boarding), scaling, termination etc. This interface will also be used by the NFV Orchestrator to reserve/allocate resources as well as to pass resource configuration to VIM such as definitions of VMs and network requirements such as the specification of the interconnections between VNF instances (i.e. network topology), e.g. HEAT templates in an OpenStack Cloud environment. Specific types of monitoring information will also be exchanged over this interface such as data related to the network connections between VNF instances either within a data centre or intra data centre connections which are physically dispersed. This interface is also used by the VIM to report back to the Orchestrator the outcome of all requests received.

Table 5. Northbound Interfaces of the T-NOVA IVM

Interfacing entities	T-NOVA Reference Point name	ETSI NFV Relevant Reference Point	Description and Comments
Virtual Network Function Management – VIM Interface	T-Vi-Vnfm	Vi-Vnfm	This interface allows the VNF Manager to request operations related to the NS lifecycle; and/or for the VIM to report the characteristics, availability, and status of VNF related infrastructure resources.
Orchestrator – VIM Interface	T-Or-Vi	Or-Vi	This interface allows the Orchestrator to request reservation/allocation of resources and NS related lifecycles operations; and for the VIM to report the characteristics, availability, and status of infrastructure resources.

3.4.6. Orchestrator – Transport Network Management Interfaces

The Transport Network Manager provides management capabilities of the transport network segment. The T-NOVA Orchestrator – Transport Network Interface (T-Or-Tm) interface supports requests from the Orchestrator to provide connectivity services typically for inter data centre (over MAN and/or WAN) or Internet connectivity. Furthermore, the interface supports retrieving monitoring information with respect to the provisioned connectivity services. This interface is also used by the TNM to report back to the Orchestrator the outcome of all requests received.

Table 6. Interfaces between the T-NOVA TNM and the Orchestrator

Interfacing entities	T-NOVA Reference Point name	ETSI NFV Relevant Reference Point	Description and Comments
Orchestrator – Transport Network Interface	T-Or-Tm	Or-Vi	This interface between TNM and the Orchestrator is used to manage the set-up, tear down and monitoring of networks

3.4.7. VIM – NFVI Interfaces

The interface between the VIM and the NFVI is used to a) communicate management decision to the NFVI layer and b) to provide to the VIM awareness about the status of the NFVI, including resource usage and availability. The T-Nf-Vi reference point can be broken down into three separate interfaces, allowing the control of the Compute, Hypervisor and NFVI-PoP network, as shown in Table 7.

Table 7. Interfaces between the T-NOVA VIM and NFVI

Interfacing entities	T-NOVA Reference Point name	ETSI NFV Relevant Reference Point	Description and Comments
VIM – Hypervisor Interface	T-Nf-Vi/H	[Nf-Vi]/H	This interface dispatches management decisions from the VIM to the Hypervisor and communicates back the Hypervisor status and monitoring metrics.
VIM – Compute Interface	T-Nf-Vi/C	[Nf-Vi]/C	This interface dispatches management decisions from the VIM to the Compute domain and communicates back the Compute domain status and monitoring metrics.
VIM – Network Interface	T-Nf-Vi/N	[Nf-Vi]/N	This interface dispatches management decisions from the VIM to the NFVI-PoP Network domain and communicates back Network status and monitoring metrics.

3.5. ETSI NFV ISG Compliance and Mapping

ETSI NFV ISG is undoubtedly the most relevant standardization initiative to the T-NOVA scope. The ISG has produced a number of high-level specification documents, which have been considered as input during the Requirement and Specification activity (Task 2.1). From those documents, the most relevant to the architecture definition process is the NFV Architectural Framework – ETSI GS NFV 002 V1.1.1 (2013-10) [23]. Besides this document, the ISG has recently released a series of draft documents that provide a more detailed view regarding the Architecture components and their interactions. These documents are in draft status but they still serve their purpose as work-in-progress documents giving a view on how the discussion on the NFV architecture evolves. These documents are:

- GS NFV INF 001 V0.3.6 (2014-01) - NFV Infrastructure Overview [6]
- GS NFV INF 001 V0.3.6 (2014-01) - NFV Infrastructure Architecture; Architecture of Compute Domain [24]
- GS NFV INF 004 V0.3.1 (2014-05) - NFV Infrastructure Architecture; Architecture of the Hypervisor Domain [25]
- GS NFV INF 005 V0.3.1 (2014-05) - NFV Infrastructure Architecture; Architecture of the Network Domain [26]
- GS NFV INF 007 V0.3.1 (2014-05) - NFV Infrastructure Architecture; Interfaces and Abstractions [27]
- GS NFV SWA 001 v0.2.0 (2014-05) – Virtual Network Functions Architecture, [28]
- GS NFV MAN 001 v0.6.1 (2014-07) - NFV Management and Orchestration, [11]

The initial analyses of those documents along with valuable inputs from the partners that currently participate in the standardisation body's activities have heavily influenced the design of T-NOVA overall architecture. T-NOVA will follow closely the ISG activities, with the ultimate aim to provide contributions based on the findings and developments during its lifecycle. It is very important to state that T-NOVA architecture although aligned with the reference architecture of NFV ISG, is designed with the requirement that it needs to be implemented in the course of the project - and in this sense its scope needs to be focused on specific use cases.

The subsection below provides a view on how the T-NOVA workplan maps to the ETSI NFV reference architecture.

3.5.1. ETSI ISG NFV impact in T-NOVA

The T-NOVA architectural vision is in general alignment with the current set of both finalised and draft ETSI documents and with the current technical approach of the three main WGs, INF, SWA and MANO. In specific:

- For WP2 which is related to Use Cases, Requirements and Architecture definition and specification:
    - Task 2.3 orchestrator key requirements and architecture clearly point to the NFVO Functional Entities currently being worked in ETSI NFV MANO WG,
    - Task 2.4 infrastructure key requirements and architecture clearly point to the NFVI and Functional Entities currently being worked in ETSI NFV INF WG,
    - Task 2.5 virtual network functions key requirements and architecture clearly point to the VNF Functional Entities currently being worked in ETSI NFV SWA WGs.

Similarly:

- WP3 Implementation of the T-NOVA Orchestrator and associated interfaces, strongly correlates with MANO WG
- WP4 implements the Infrastructure Virtualisation and Management layer comprising the NFVI and the VIM. T-NOVA assigns the Infrastructure Management (i.e. VIM/TNM) to a separate WP, due to the multiplicity of the cardinality that this architectural component withholds in the proposed infrastructure. The work here corresponds to both ETSI NFV MANO WG (partially) and to ETSI NFV INF WG.
- WP5 Implementation of the VNF framework as well as the Function Store and the actual VNFs, is aligned with ETSI NFV SWA WG.

Task 2.6 and WP6, which introduce the Marketplace concept which along with Function Store in T2.5/WP5 are the novelties proposed by T-NOVA aiming at opening the NFV market to third party developers for the provision of VNFs. Currently this concept falls outside the technical view of ETSI NFV.

The next figure illustrates the above mentioned mapping between ETSI ISG NFV Functional Entities and the Work Packages / Tasks of the T-NOVA workplan.

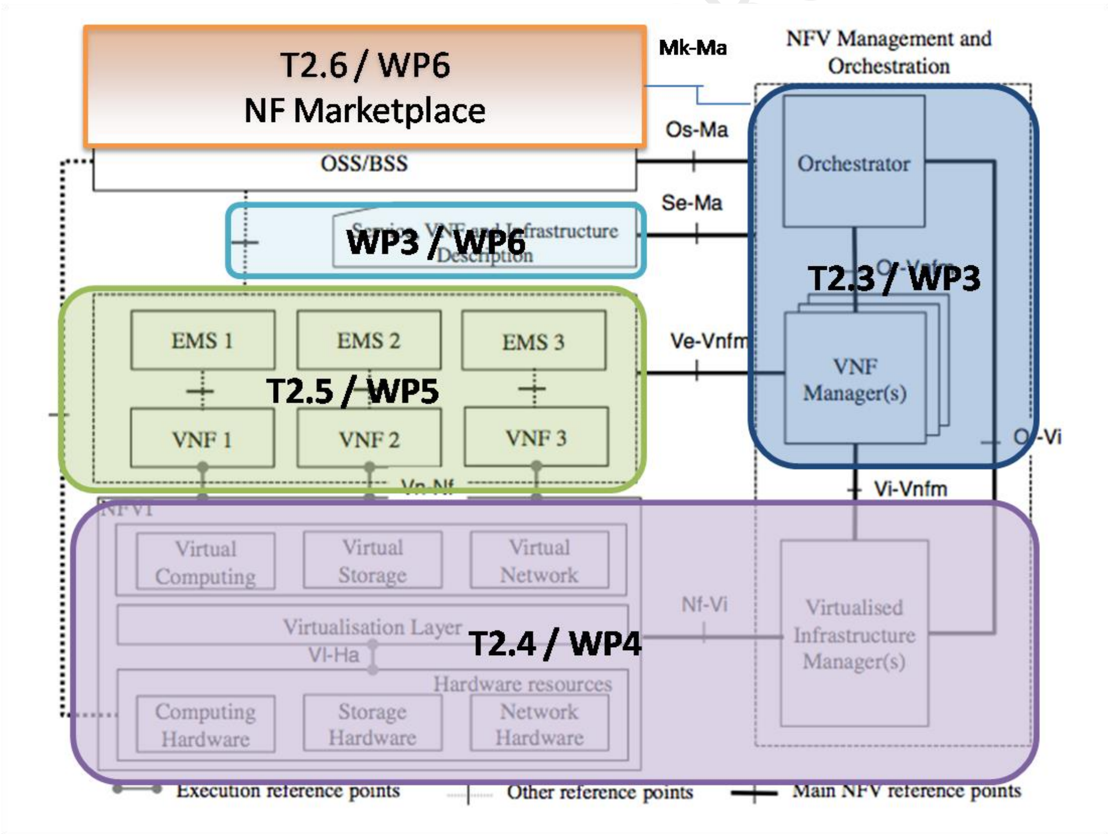

Figure 22. T-NOVA scope mapping into ETSI NFV reference architecture

The following table (

Table 8) summarises on the mapping of T-NOVA to ETSI view related to the most prominent architecture components.

Table 8. Mapping of T-NOVA to ETSI architectural component

Architectural Component	ETSI View	T-NOVA View
Virtualised Network Function (VNF)	A VNF is a virtualization of a network function in a legacy non-virtualized network.[6]	T-NOVA and ETSI views are completely aligned. In T-NOVA the VNF are considered as single or multiple VMs deployed in Compute Nodes.
NFV Infrastructure	The NFV Infrastructure is the totality of all hardware and software components, which build up the environment in which VNFs are deployed, managed and executed. It can span across several locations, The network providing connectivity between these locations is regarded to be part of the NFV Infrastructure.[7]	T-NOVA also aligns its definition to that of ETSI. The proposed overall architecture, as well as the detailed w.r.t the NFVI (subject of Task2.4/WP4) supports the existence of multiple NFVI-PoPs at several locations as part of the NFVI, interconnected by transport network links.
Hardware Resources	Includes the physical IT (i.e. computing and storage) and network resources available at the NFVI. IT resources are considered to be provided by COTS equipment. Network resources include network elements such as switches and routers along with interconnecting links. Two types of networks are considered for the NFVI: NFVI-PoP Network: providing the networking required to interconnect the IT resources along with external network connectivity Transport Network: providing interconnection between NFVI-PoPs	T-NOVA is also aligned w.r.t the hardware resources supported. Besides IT resources that will be based on COTS hardware, networking resources will be provided by hardware network elements and physical links.
Virtualisation Layer and Virtualised	The virtualization layer abstracts the hardware	Our solution is aligned to ETSI view, with the latter being more generic

[6] ETSI GS NFV 002 V1.1.1, Section 7.2.2, pp14

[7] ETSI GS NFV 002 V1.1.1, Section 7.2.4.1, pp15.

resources	resources and decouples the VNF software from the underlying hardware, thus ensuring a hardware independent lifecycle for the VNFs. The NFV architectural framework does not restrict itself to using any specific virtualization layer solution. To ensure operational transparency, the operation of the VNF should be independent of its deployment scenario.	with respect to the actual implementation choices. The virtualization layer will be based on currently available virtualization enabler frameworks (e.g. OpenStack) using open source hypervisors for the abstraction of the IT and network resources within the compute nodes. With respect to the network domain, the network hardware will be virtualized by the virtualisation layer in order to be able to virtualize network paths that provide connectivity to between VMs running VNF instances.
Virtualised Infrastructure Manager(s)	Virtualized Infrastructure Management comprises the functionalities that are used to control and manage the interaction of a VNF with computing, storage and network resources under its authority, as well as their virtualization. Multiple Virtualized Infrastructure Manager instances may be deployed	T-NOVA is completely aligned to the specification of VIM by ETSI. For T-NOVA multiple VIM instances are considered, one for each NFVI-PoP.
Orchestrator	Orchestration and management of NFV infrastructure and software resources, and realizing network services on NFVI	Orchestrator will be the component that will allow the orchestration and management of the NFVI, by orchestrating the software and hardware resources for the provision of NFaaS
VNF Manager(s)	Manages the VNF lifecycle.	Total alignment to ETSI view.
Operation and Business Support Systems (OSS/BSS) – Marketplace	ETSI view does not provide any more insight on the OSS/BSS of the operator apart from the definition of an interface.	Although the OSS/BSS system is not in the scope of T-NOVA, the proposed Marketplace component of the overall architecture contains partially some OSS/BSS modules (i.e. Billing, Accounting, SLA monitoring, AAA) that will be implemented/adapted. Additionally the notion of the Marketplace is considered an added value, compared to the ETSI view.
Network Function	In ETSI view this component	The NF Store will be the

Store	does not exist. Most of the VNF related metadata information is included in Service VNF and Infrastructure Description component. There is no component in the TSI NFV proposed overall architecture that acts as a repository of the VNF images and templates.	architectural component, proposed by T-NOVA that will withhold metadata information and description related to the VNFs, along with the actual images of the VNFs submitted by the Function Providers (NF repository).

4. SERVICE LIFECYCLE AND SEQUENCE OF INTERACTIONS

This section presents the mapping of the Use Cases (UCs) described in Deliverable D2.1 (System Use Cases and Requirements), Section 5.2 to sequence diagrams. These sequence diagrams illustrate how the involved components of the T-NOVA overall architecture are involved in each use case and how they interact with each other. The interactions are only described in a high-level manner; forthcoming deliverables will provide a more detailed insight on the workflow for achieving each system Use Case.

4.1. UC1: Compose NFV Services

UC1 specifies the interactions that take place during the VNF composition phase. The involved stakeholders (i.e. SP, FP and NIP) are participating via the Dashboard component. It is important to state that the composition can have two different triggering modes: i) offline (asynchronously triggered by the SP) ii) on line (dynamically triggered by the Customer request).

For the sake of clarity, the Dashboard, although considered as part of the Marketplace, is illustrated separately. Additionally, UC1 is the overall UC that also contains generic execution parts of UC1.1 – UC1.3. UC1.1 and UC1.3 are related to the interaction of the Customer with T-NOVA with respect to the provision of a NS. Lastly UC1.2 is related to the VNF advertisement by the FPs, and is independent of the other UCs.

The sequence diagram for this UC is illustrated in 3.

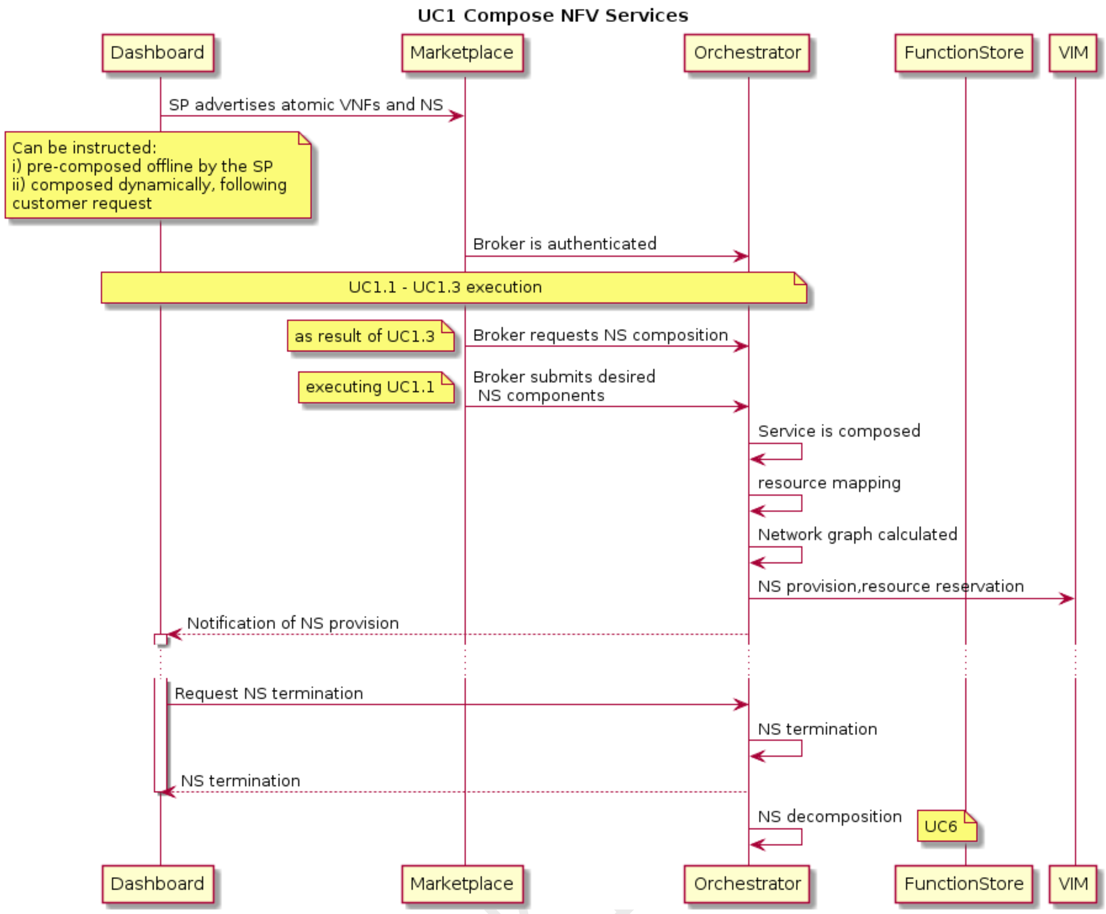

Figure 23. UC1 Sequence Diagram

4.1.1. UC1.1 Browse / select offerings: service + SLA + pricing

UC1.1 defines how the T-NOVA Customer selects the service among the provided offerings (service, SLA, pricing) and how the SLA agreement is established among the involved parties. The involved stakeholders are the Customer, the SP, the FP and the IP.

The sequence diagram for this UC is illustrated in 4.

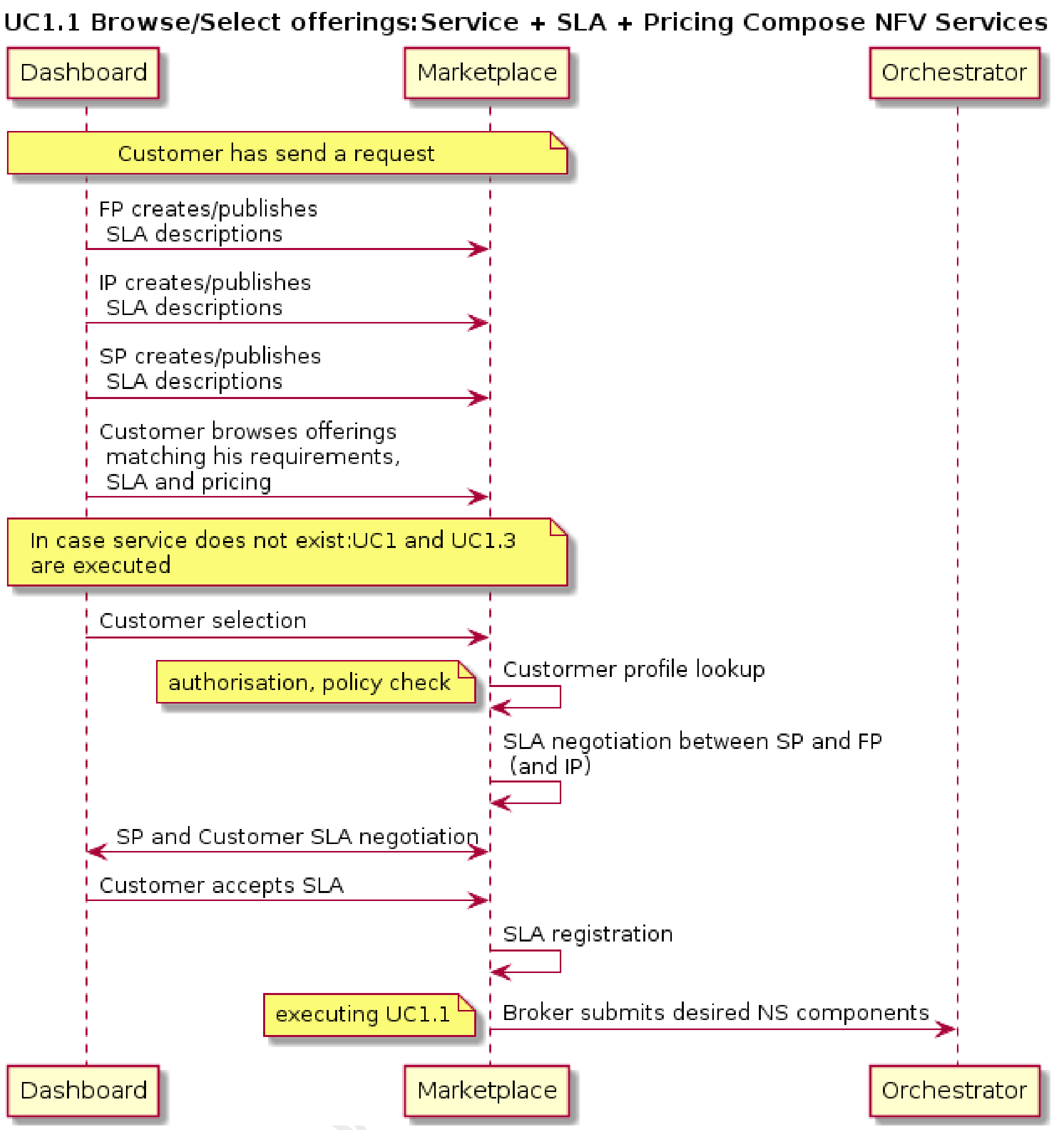

Figure 24. UC1.1 Sequence Diagram

4.1.2. UC1.2 Advertise NFs

This UC presents the procedure for supporting the advertisement of NFs by the Function Providers.

The sequence diagram for this UC is illustrated in the figure below.

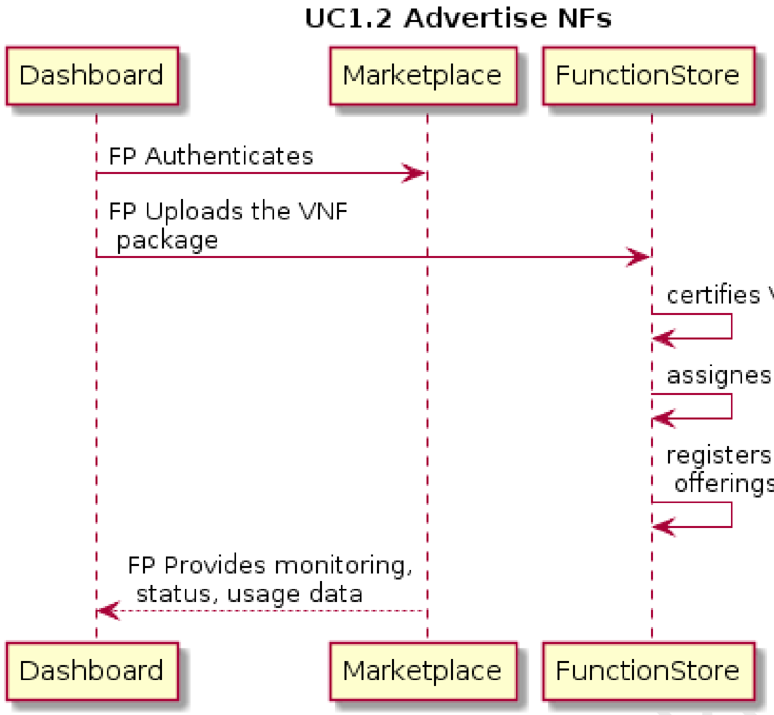

Figure 25. UC1.2 Sequence Diagram

4.1.3. UC1.3 Bid /trade

The UC describes the workflow for the trading between the SP and the FP through the T-NOVA Marketplace.

The sequence diagram for this UC is illustrated in the figure below.

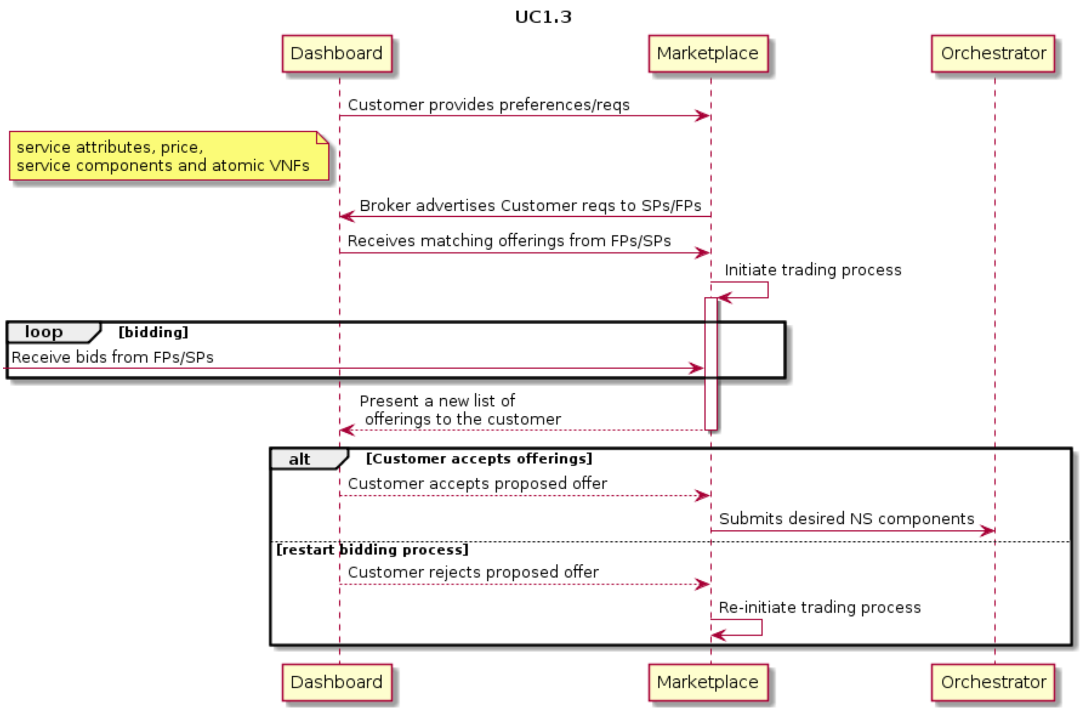

Figure 26. UC1.3 Sequence Diagram

4.2. UC2 Provision NFV Services / UC2.1 Map and deploy service

UC2 describes the workflow for provisioning T-NOVA Network Services. The main stakeholder participating within this UC is the SP. This role is the one that controls the whole NFV service provisioning process. The Customer is notified at the last step of the UC, in order to successfully complete the whole NS provisioning workflow.

UC2.1 is summoned by UC2 in order to map the required network and computing resources and fulfil the deployment of the VNFs to the appropriate locations in the Infrastructure.

The sequence diagram for this UCs is illustrated in Figure 27.

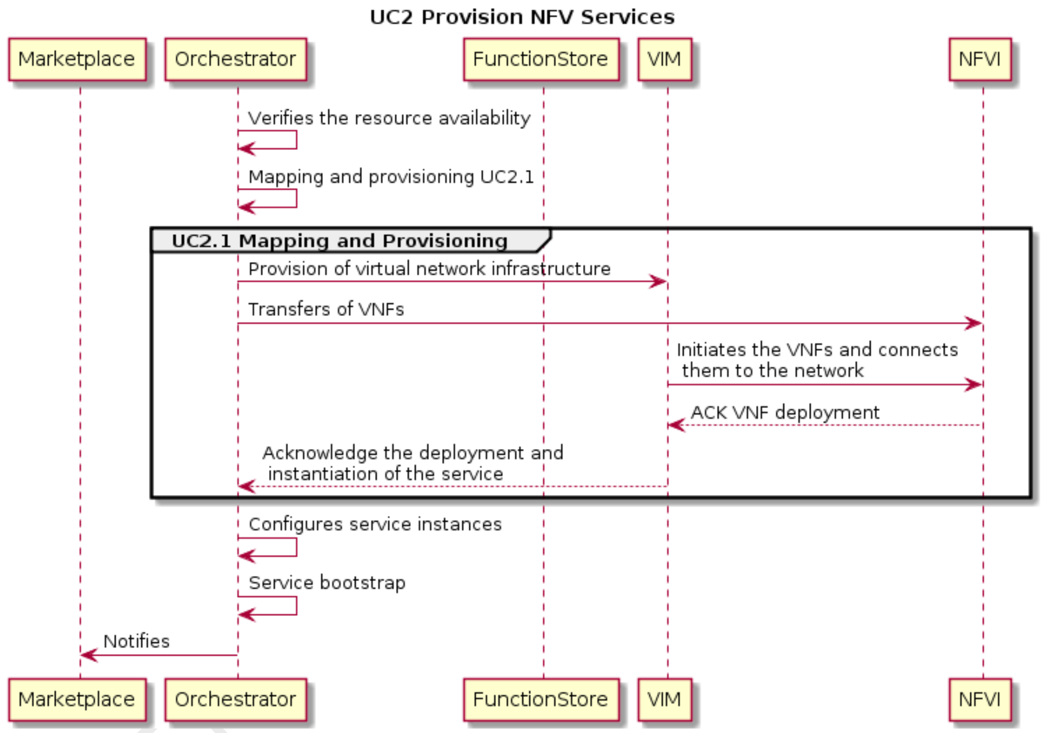

Figure 27. UC2 / UC2.1 Sequence Diagram

4.3. UC3 Reconfigure/Rescale NFV Services

UC3 (and UC3.x) is focused on the adaptation of the resources allocated to a specific service, towards SLA fulfilment and resource usage optimisation. For all UC3.x Use Cases, the precondition is the execution of UC2 and auto-scaling support of the involved VNF Service.

4.3.1. UC3.1 Scale-Out / In VNF Service

This UC comprehends the modification (scale-out or scale-in) of the number of instantiated VNF images used by a given NS.

The sequence diagram for this UCs is illustrated in Figure 28.

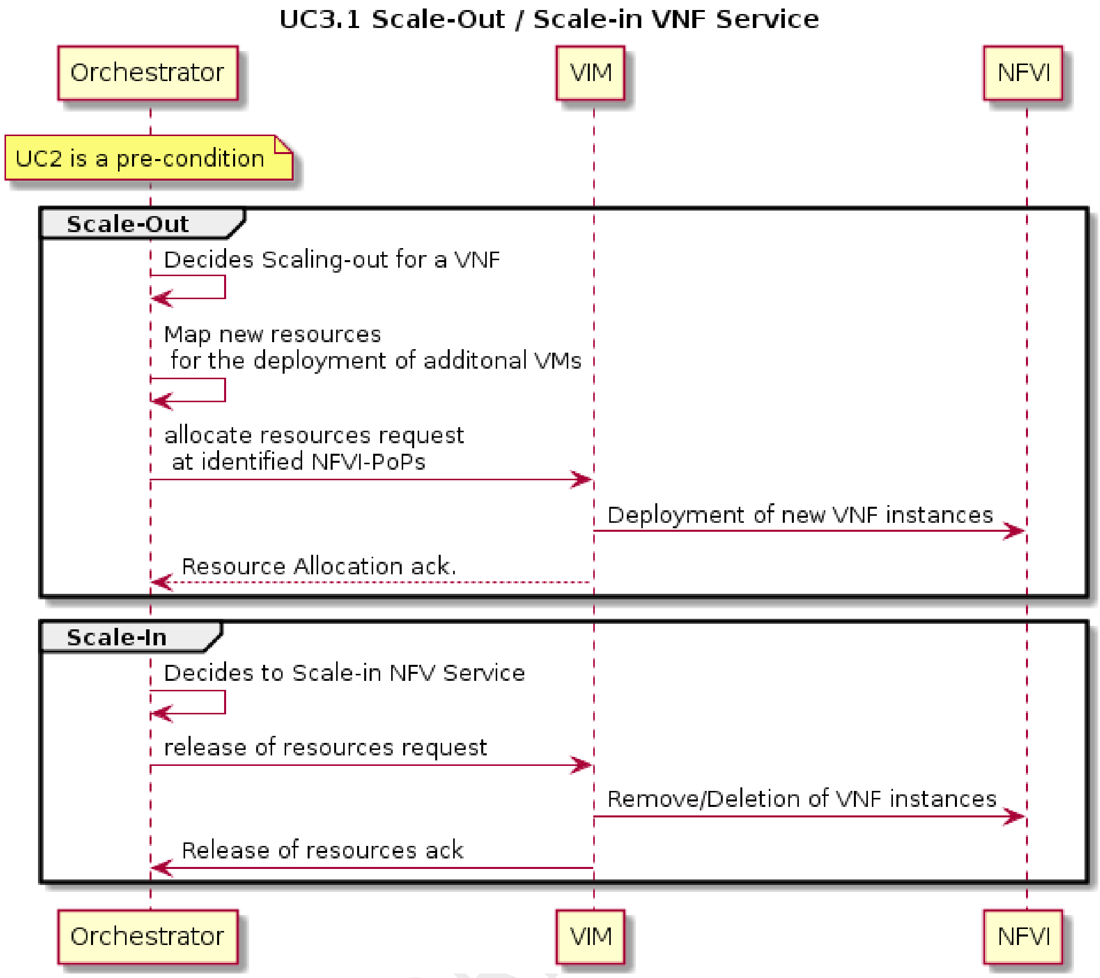

Figure 28. UC3.1 Sequence Diagram

4.3.2. UC3.2 Scale-Up /Down VNF Service

UC3.2 involves the increase or decrease of the computing/network resources allocated to a single VNF instance.

The sequence diagram for this UCs is illustrated in Figure 29.

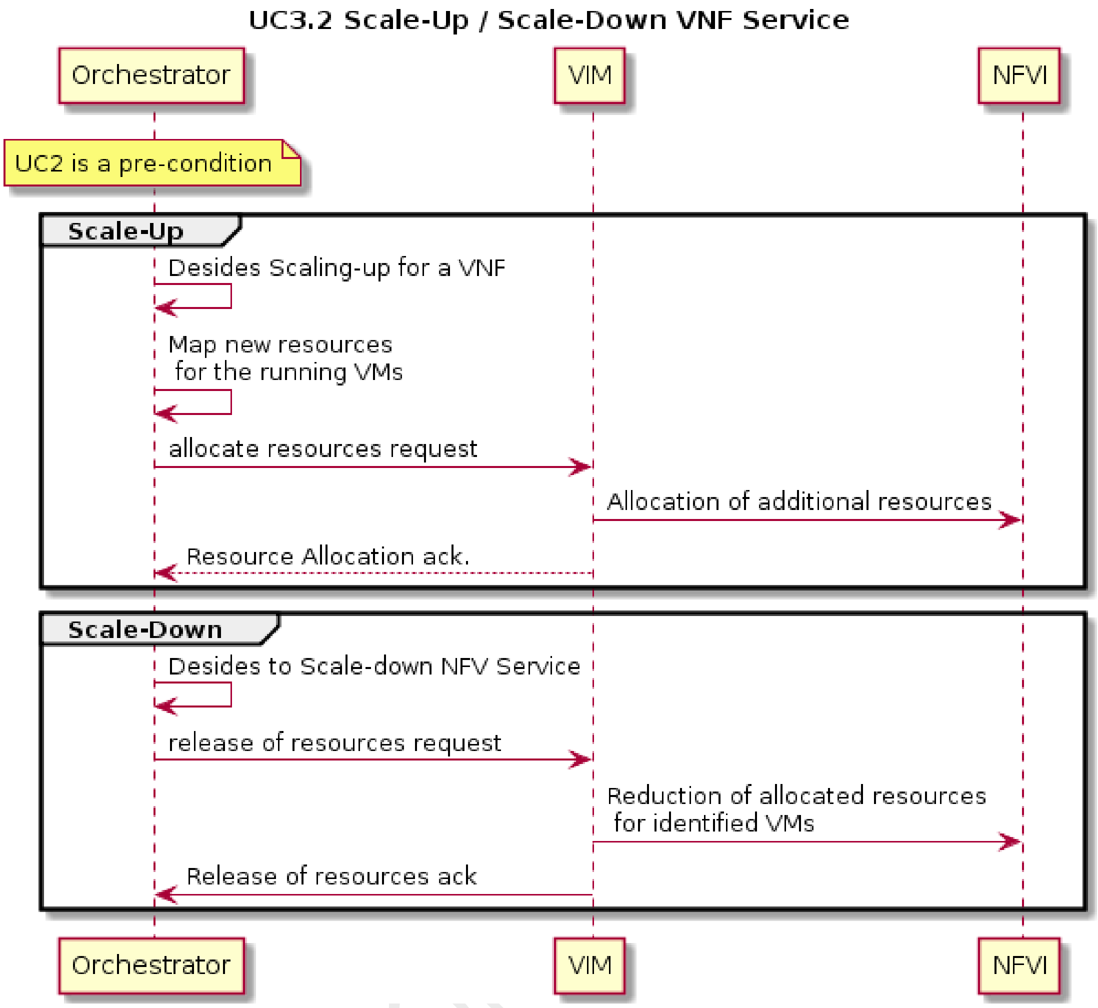

Figure 29. UC3.2 Sequence Diagram

4.3.3. UC3.3. Reconfigure VNF Service

This UC realises the reconfiguration of VNF Service parameters. It can be automatically triggered, scheduled at a given time or manually launched by the SP and/or the Customer.

The sequence diagram for this UC is illustrated in Figure 30

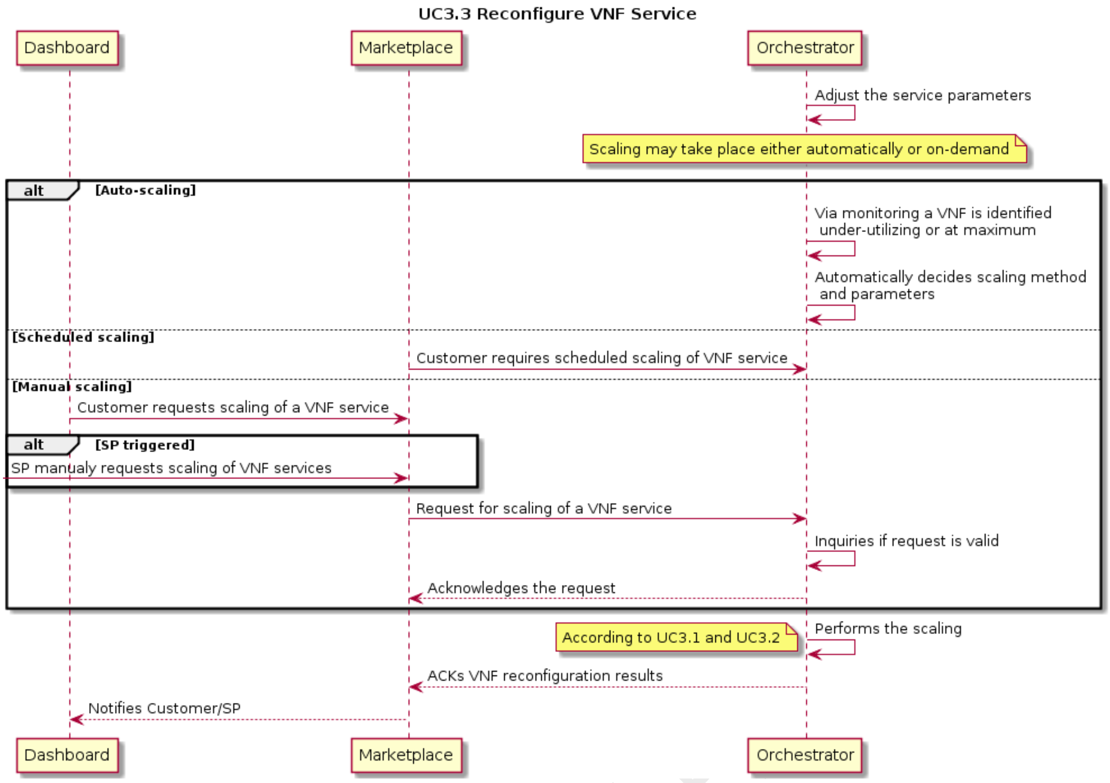

Figure 30. UC3.3 Sequence Diagram

4.4. UC4. Monitor NFV Services

This UC refers to the monitoring of the consumed resources by a T-NOVA service as well as its overall status. The monitoring service constantly operates and provides monitoring metrics to the Customer and the SP.

The sequence diagram for this UC is illustrated in Figure 31.

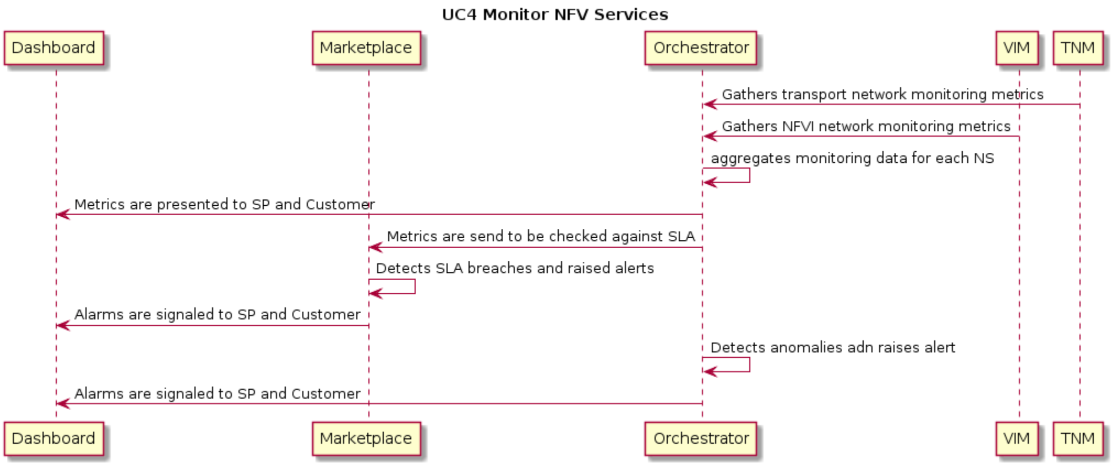

Figure 31. UC4 Sequence Diagram

4.4.1. UC 4.1 Monitor SLA

This UC defines the workflow for the evaluation of the agreed SLA among the stakeholders. The UC also considers the reaction to be taken according to the results.

The sequence diagram for this UCs is illustrated in Figure 32

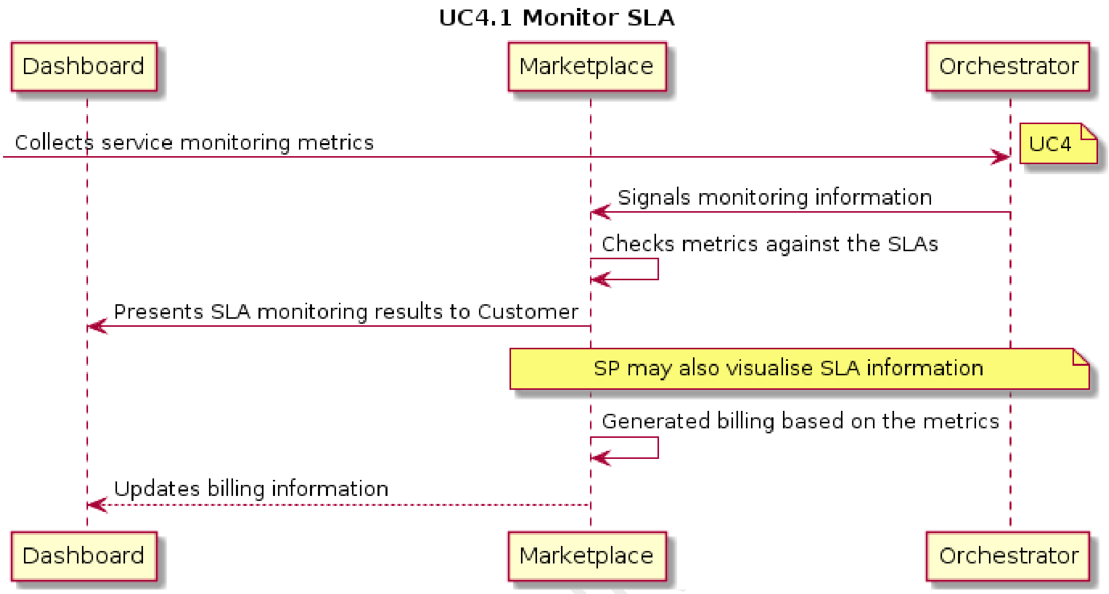

Figure 32. UC4.1 Sequence Diagram

4.5. UC5 Bill NFV Service

UC5 defines the billing procedure for a T-NOVA Customer, and in extent the billing for SP by the FP (and IP) based on accounting and processing of SLA conformance.

The sequence diagram for this UCs is illustrated in Figure 33.

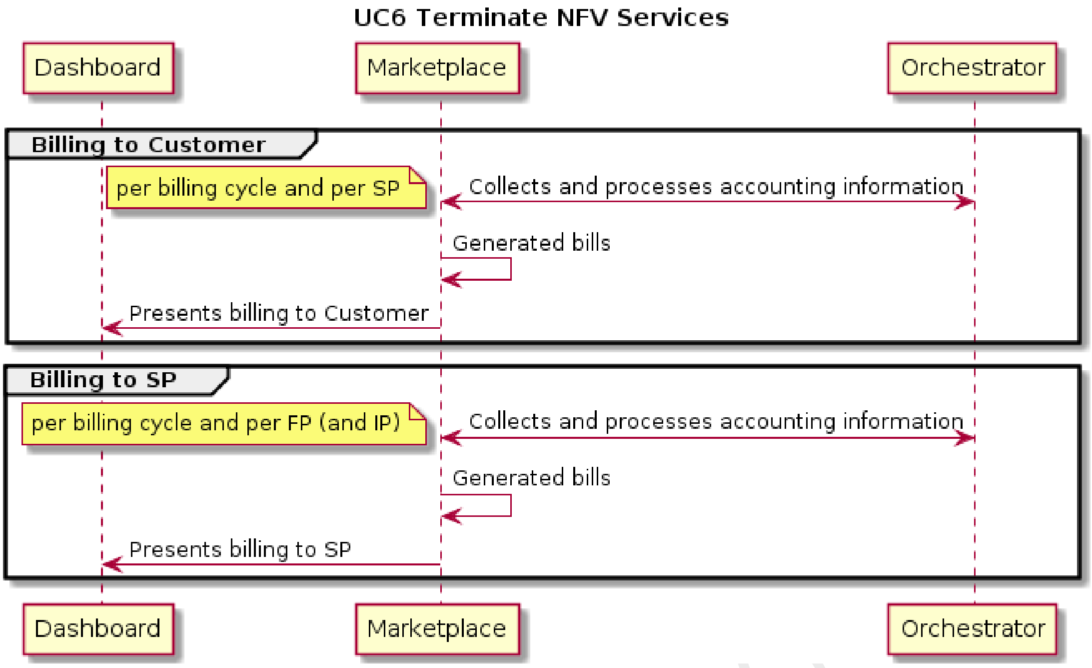

Figure 33. UC5 Sequence Diagram

4.6. UC6 Terminate NFV services

This UC examines the termination workflow covering three distinctive cases grouped by the entity that triggers the termination of a service. The T-NOVA Customer triggers the first case, the SP triggers the second, while the third case is when the SP not only terminates a service but also removes the NF from the Catalogue.

The sequence diagram for this UCs is illustrated in Figure 34

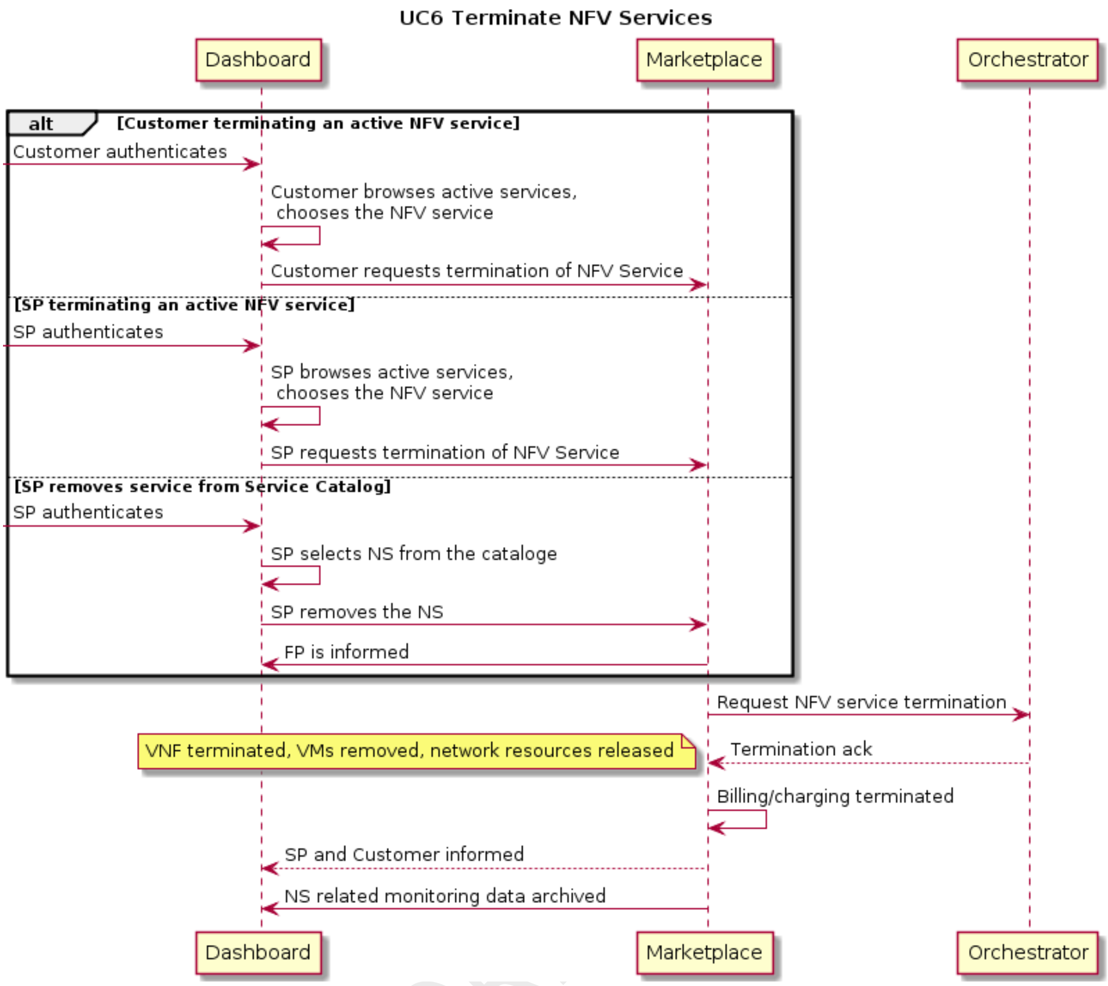

Figure 34. UC6 Sequence Diagram

5. ARCHITECTURE EXTENSIONS

This section contains a description of how the T-NOVA architecture can be extended in order to adapt to different scenarios, according to diverse value chain configurations and stakeholders' needs. It must be noted that these architectures are out-of-scope of the project implementation. However, the aim of this section is to demonstrate the elasticity, and adaptability of the T-NOVA framework, as a future-proof architecture for VNF deployment and management.

5.1. BSS/OSS Interfaces

This scenario depicts how the T-NOVA reference architecture can be easily integrated with the current BSS/OSS systems that network operators and/or service providers may have already deployed in their facilities. The aim of this integrated scenario is to depict how the current reference architecture can be deployed in actual environments and being integrated in the different BSS/OSS systems of the stakeholder.

Integration of novel solutions with already existing systems becomes a key item to be considered when moving towards actual deployments.

- **Business value**: Integration of the novel T-NOVA framework with existing deployed solutions. Facilitates the migration from current systems to T-NOVA-like framework
- **T-NOVA roles**: T-NOVA Broker, T-NOVA SP, T-NOVA Infr. Provider

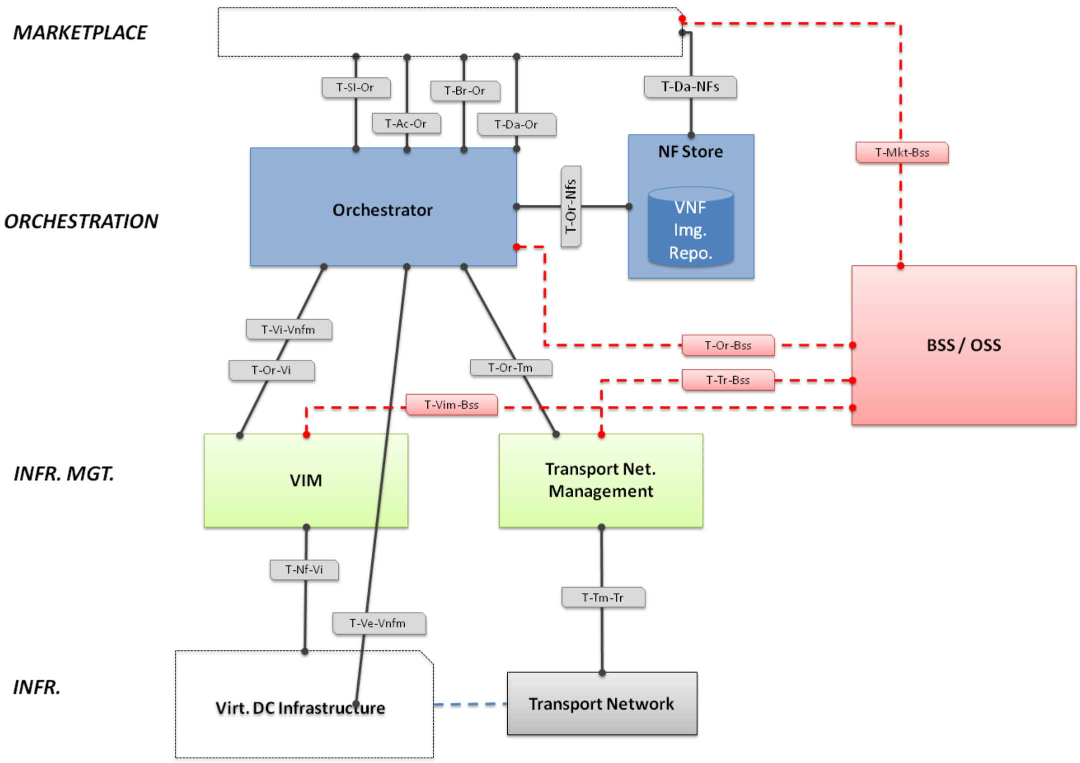

Figure 35. Architecture extension for interfacing with BSS/OSS

5.2. Multiple NFVI-PoP providers

This alternative case comprehends a simple extension of the reference architecture. In this case, some of the VIMs and the corresponding infrastructure are owned by a distinct infrastructure provider.

This case considers that SLAs at the infrastructure level between the distinct providers are in place. However, in order to ensure the correct deployment of VNFs and the correct service provisioning workflows execution, it is required that the BSS/OSS systems of both providers communicate.

Furthermore, since the third-party infrastructure provider is completely independent, the VIM' may not be compliant with the T-NOVA VIM functionalities. Thus, it may be the case that the corresponding T-Or-* interfaces must be adapted in order to integrate both infrastructure segments.

- **Business value**: Include different infrastructure providers (with differentiated VIMs) into a T-NOVA-like scenario for service provisioning and VNF management. Integration of T-NOVA with existing different virtualisation solutions.
- **T-NOVA roles involved**: T-NOVA Broker, T-NOVA SP, T-NOVA Infr. Provider

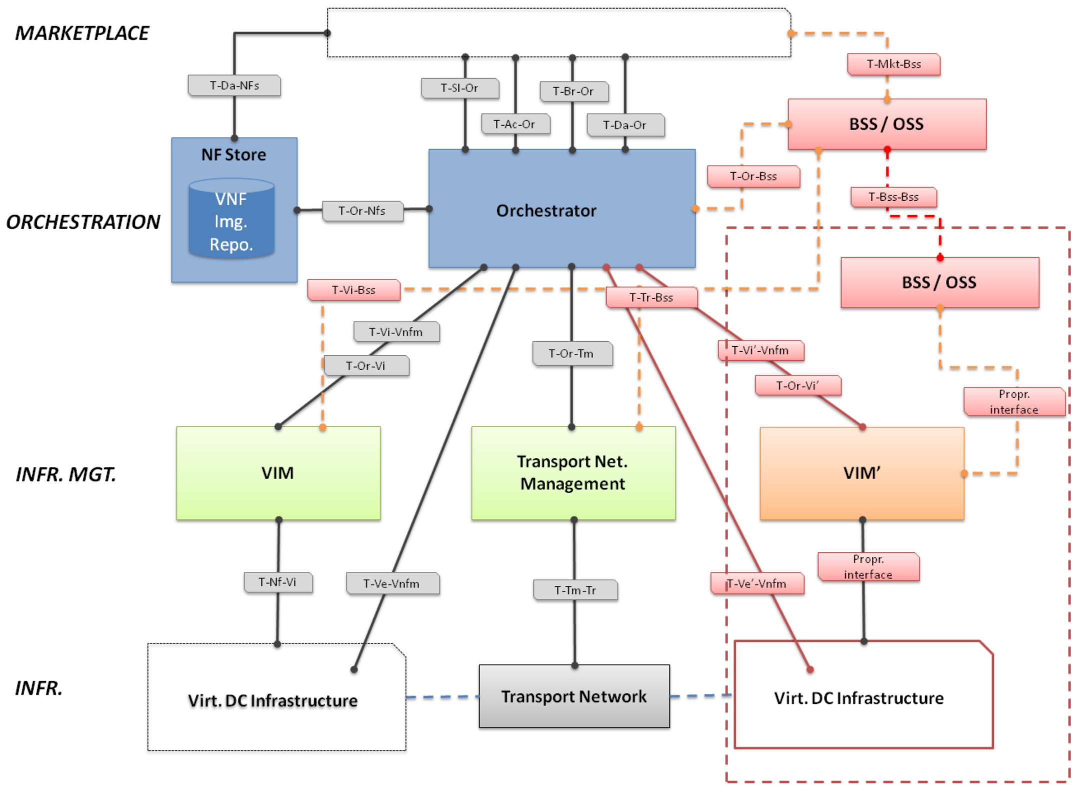

Figure 36. Architecture extension to support multiple NFVI-PoP providers

5.3. Multiple T-NOVA SPs, Single Broker

This basic case comprehends the scenario where there are several service providers, not federated among them (i.e. horizontally disjoint), and there is one single broker which interacts with all of them. Considering that the broker is an intermediary business entity that facilitates trading between the customer and the service providers, this scenario only depicts a typical case where a single broker is capable of providing services using several SPs below them.

- **Business value:** Increase the service offering from the T-NOVA broker towards the T-NOVA consumer, since services can now be provisioned over the infrastructure offered by two (or more) SPs.
- **T-NOVA roles:** T-NOVA Customer, T-NOVA broker, T-NOVA SP

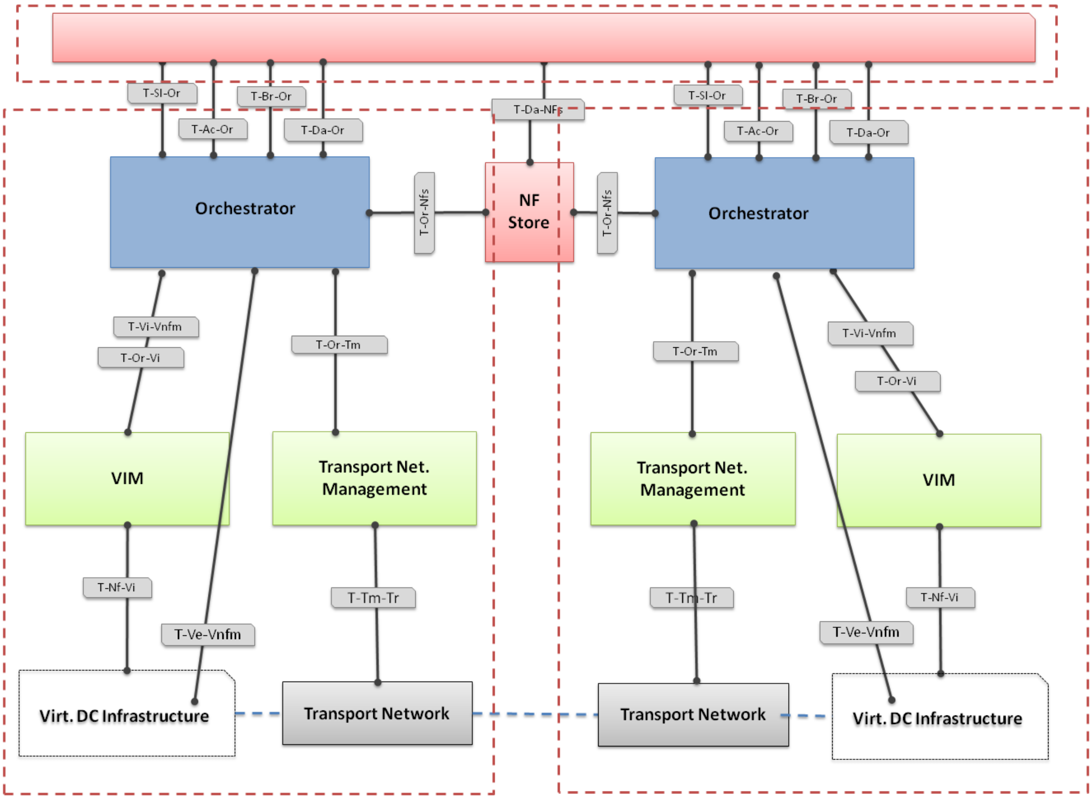

Figure 37. Architecture extension to support multiple SPs under a common broker

5.4. Multiple T-NOVA SPs (Orchestrator Federation)

This scenario comprehends the case where there are several vertically integrated service providers (i.e. two differentiated service providers, that are also capable of playing the broker role and thus they are capable of combining multiple applications into one sellable service). Typically, there would be no communication between the different service providers, since they are direct competitors.

However, in actual production environments, there are several business agreements between service providers. Those agreements respond to different business needs (e.g. increase service portfolio). In order 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 *T-Or-Or*.

This new interface shall include all the information exchange procedures, as well as authentication and authorization between the different service providers.

Through this federation of service providers, 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.

- **Business value**: Increase the possibility of the T-NOVA to sell its services to the T-NOVA broker by means of utilizing resources from the federated SP
- **T-NOVA roles**: T-NOVA SP

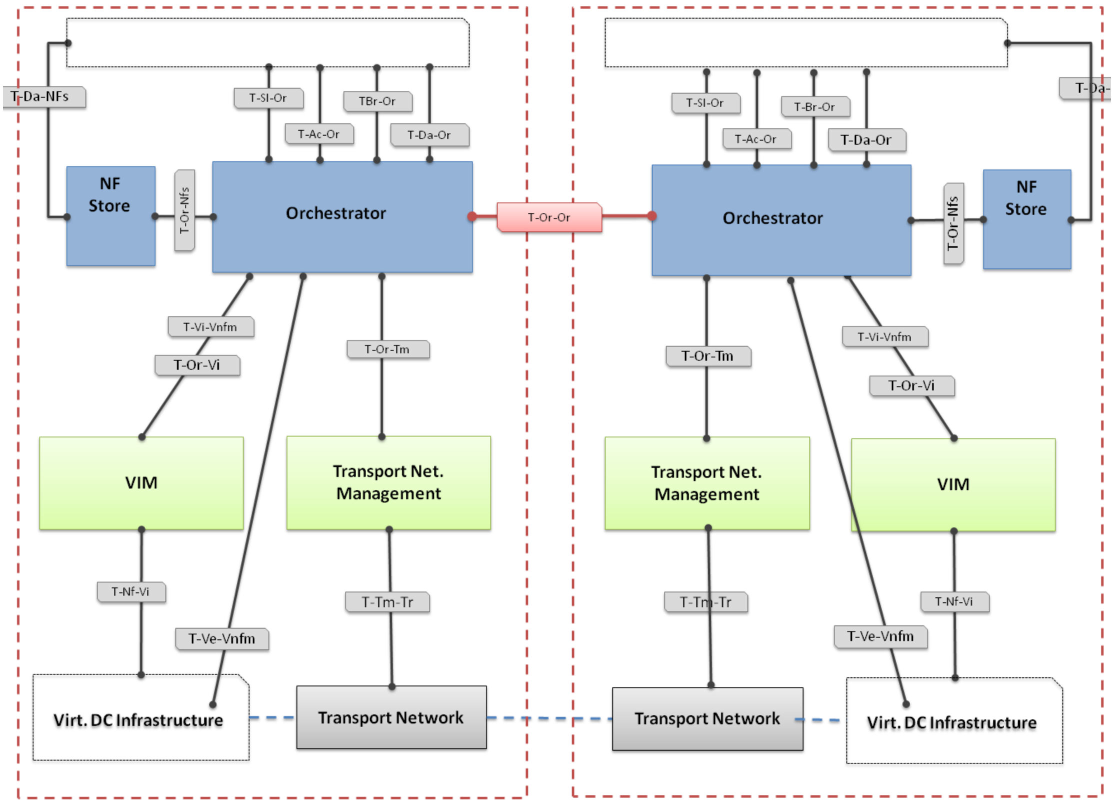

Figure 38. Architecture extension for SP federation

5.5. Multiple brokers over the same SP

This scenario corresponds to a situation where two or more independent brokers utilize the same SPs in order to offer the different services to the T-NOVA customers.

From the broker perspective, there is no change in their business operation. From the service provider perspective, it aims at maximizing theirrevenues by means of serving two or more (the maximum number of) brokers.

- **Business value**: T-NOVA SP revenues maximized by means of increasing the number of SP customers (i.e. T-NOVA broker)
- **T-NOVA roles**: T-NOVA broker, T-NOVA SP

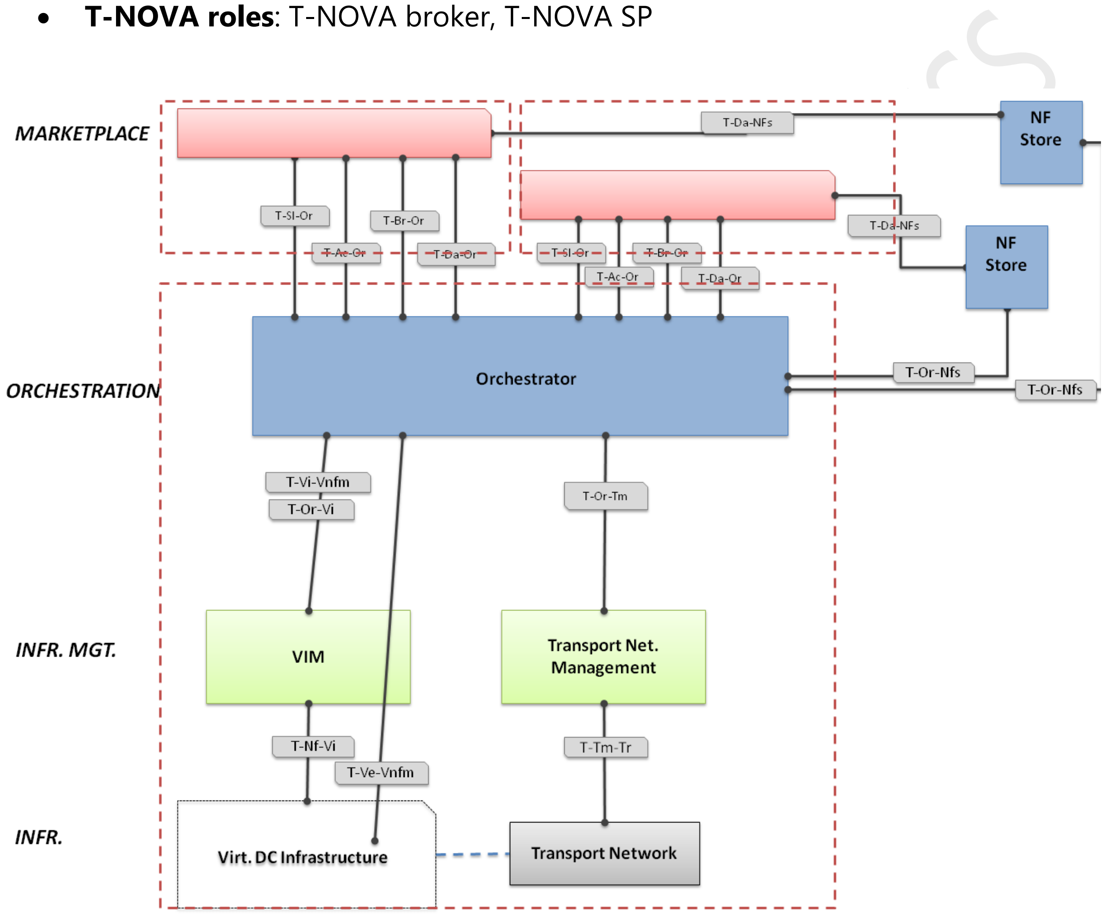

Figure 39. Architecture extension for multiple brokers under the same SP

6. CONCLUSIONS

This document presents a first approach to the high-level overall architecture of the T-NOVA system, the entities and the main interfaces/reference points. All T-NOVA partners contributed to this endeavour, achieving consensus among the consortium members on the initial architectural vision.

Several integrated NFV architectures proposed by industrial consortia, R&D projects and standardisation bodies were taken into account. T-NOVA inherited and adapted several concepts and architectural principles from the surveyed proposals in order to come up with a layered system architecture which fulfils all user requirements set out in D2.1, is reasonably complex and feasible to implement, being compatible with existing state-of-the-art IT and network infrastructures. In addition, the proposed architecture is compliant with the current technical approach as well as the terminology of ETSI ISG NFV, introducing at the same time several extensions to it.

Furthermore, a technical analysis of the use cases defined in D2.1, using sequence diagrams involving the high-level architectural entities, initially proves that the proposed architecture can effectively accommodate all system use cases.

Using the overall architecture as reference, the project can proceed to the next tasks, which are the detailed definition of the T-NOVA layers and subsystems (to be contained in deliverables D2.31 and D2.41) as well as the initiation of the implementation phase. Using an iterative approach, the feedback received from the detailed subsystems' design and specification as well as from the early phases of implementation will help to refine and amend the overall architecture as well. The outcomes of this refinement will be reflected in the second release of this deliverable (D2.22: Overall System Architecture and Interfaces – Final).

7. REFERENCES

[1] TM Forum. TM Forum WebSite. [Online]. http://www.tmforum.org

[2] TMF ZOOM. TMF ZOOM. [Online]. http://www.tmforum.org/ZOOM/16042/home.html

[3] TMF. (2014) TR229 ZOOM/NFV User Stories V1.3.1. [Online]. http://www.tmforum.org/TechnicalReports/TR229ZOOMNFVUser/54793/article.html

[4] TMF. TMF NFV Catalyst Program. [Online]. http://www.tmforumlive.org/catalysts/

[5] ETSI NFV ISG, "GS NFV 001 Network Function Virtualisation (NFV); Use Cases," 2013.

[6] ETSI NFV ISG. (2014, March) GS NFV INF 001 NFV Infrastructure Overview.

[7] ETSI NFV ISG, "NFV White Paper Update," 2013.

[8] ATIS, "ATIS-I-0000044 Operation Opportunities and CHallenges of SDN/NFV Programmable Infrastructure," 2013.

[9] TMF. TR227 Specifications relevant for MANO Work V.1.0.1. [Online]. http://www.tmforum.org/TechnicalReports/TR227TMForumSpecifications/54805/article.html

[10] TMF. TR228 Gap Analysis related to MANO Work V1.0.1. [Online]. http://www.tmforum.org/TechnicalReports/TR228TMForumGAP/54770/article.html

[11] ETSI NFV ISG. (2014, July) NFV-MAN 001 NFV Management and Orchestration. [Online]. http://docbox.etsi.org/ISG/NFV/Open/Latest_Drafts/NFV-MAN001v061%20management%20and%20orchestration.pdf

[12] CloudNFV. (2013) CloudNFV. [Online]. http://www.cloudnfv.com

[13] TMF. GB922: Information Framework. [Online]. http://www.tmforum.org/browse.aspx?type=4&search=GB922#TRCSearch/LIST

[14] HP. (2014) OpenNFV. [Online]. http://h71028.www7.hp.com/enterprise/downloads/services/4AA5-1044ENW.pdf

[15] INTEL. (2014) Intel Network Builders. [Online]. http://networkbuilders.intel.com

[16] Qosmos. (2014) White Paper: Realising the Benefits of Network Functions Virtualisation in Telecoms Networks. [Online]. http://embedded.communities.intel/com/docs/DOC-7981

[17] Alcatel-Lucent. (2014) CloudBand. [Online]. http://www.alcatel-lucent.com/solutions/cloudband

[18] MCN Consortium. (2013) http://www.mobile-cloud-networking.eu. [Online]. http://www.mobile-cloud-networking.eu

[19] MCN Consortium. (2013) Deliverable D2.2 - Overall Architecture Definition. [Online]. http://www.mobile-cloud-networking.eu/site/index.php?process=download&id=124&code=93b79f8f5b99f67a6cdc28369c05b65f624cfee7

[20] Project CONTENT FP7. Official Web Site. [Online]. http://content-fp7.eu

[21] Project UNIFY. (2013) Project UNIFY. [Online]. https://www.fp7-unify.eu/

[22] Project NetIDE. (2013) Project NetIDE. [Online]. http://www.netide.eu

[23] ETSI NFV ISG. (2013, October) GS NFV 002 NFV Architectural Framework.

[24] ETSI NFV ISG. (2014, January) GS NFV INF 001 NFV Infrastructure Architecture; Architecture of Compute Domain.

[25] ETSI NFV ISG. (2014, May) INF WG.

[26] ETSI NFV ISG. (2014, May) INF WG.

[27] ETSI NFV ISG. (2014, May) GS NFV INF 007 NFV Infrastructure Architecture Interfaces and Abstractions.

[28] ETSI NFV ISG. (2014, May) SWA WG.

[29] Project MCN FP7. (2013) Mobile Cloud Networking. [Online]. http://www.mobile-cloud-networking.eu

[30] John Rath. (2014) Alcatel-Lucent Expands CLoud Partnership With Intel. [Online]. http://www.datacenterknowledge.com/archives/2014/02/26/alcatel-lucent-expands-cloud-partnership-intel

[31] Citi Research, "Cloud Computing - A Two Part Series Part1: Overview, Drivers, Forms and Service Types," 2012.

LIST OF ACRONYMS

Acronym	Explanation
AAA	Authentication, Authorisation, and Accounting
API	Application Programming Interface
CAPEX	Capital Expenditure
CIP	Cloud Infrastructure Provider
CSP	Communication Service Provider
DASH	Dynamic Adaptive Streaming over HTTP
DDNS	Dynamic DNS
DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Denial of Service
DoW	Description of Work
DPI	Deep Packet Inspection
DPDK	Data Plane Development Kit
E2E	End-to-End
EU	End User
FP	Function Provider
GW	Gateway
HG	Home Gateway
HTTP	Hypertext Transfer Protocol
IP	Internet Protocol
IP	Infrastructure Provider
ISG	Industry Specification Group
ISP	Internet Service Provider
IT	Information Technology
KPI	Key Performance Indicator
LAN	Local Area Network
MANO	MANagement and Orchestration
MVNO	Mobile Virtual Network Operator

NAT	Network Address Translation
NFaaS	Network Functions-as-a-Service
NFV	Network Functions Virtualisation
NFVI	Network Functions Virtualisation Infrastructure
NFVIaaS	Network Function Virtualisation Infrastructure as-a-Service
NIP	Network Infrastructure Provider
NS	Network Service
OPEX	Operational Expenditure
OSS / BSS	Operational Support System / Business Support System
PaaS	Platform-as-a-Service
PoC	Proof of Concept
QoS	Quality of Service
RTP	Real Time Protocol
SA	Security Appliance
SaaS	Software-as-a-Service
SBC	Session Border Controller
SDN	Software-Defined Networking
SDO	Standards Development Organisation
SI	Service Integrator
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SME	Small Medium Enterprise
SP	Service Provider
TEM	Telecommunication Equipment Manufacturers
TRL	Technology Readiness Level
TSON	Time Shared Optical Network
UC	Use Case
UML	Unified Modelling Language
vDPI	Virtual Deep Packet Inspection
vHG	Virtual Home Gateway
VM	Virtual Machine
VNF	Virtual Network Function
VNFaaS	Virtual Network Function as a Service

VNPaaS	Virtual Network Platform as a Service
vSA	Virtual Security Appliance
vSBC	Virtual Session Border Controller
WAN	Wide Area Network
WP	Work Package

ANNEX 1 TERMINOLOGY

This annex contains a reference of all main T-NOVA architectural entities as well as some general terms used throughout the deliverable.

The terms marked with an asterisk (*) have been aligned with ETSI NFV ISG terminology.

A.1.1 General Terms

Name	Virtualised Network Function (VNF)*
Description	A virtualised (pure software-based) version of a network function

Name	Virtualised Network Function Component (VNFC)*
Description	An independently manageable and virtualised component (e.g. a separate VM) of the VNF

Name	T-NOVA Network Service (NS)
Description	A network connectivity service enriched with in-network VNFs, as provided by the T-NOVA architecture.

Name	Network Function Virtualisation Point of Presence (NFVI-PoP)*
Description	A physical location where VNFs can be deployed. It can refer to large, centralised data centres or light-DCs distributed in the network.

Name	NFV Infrastructure (NFVI)*
Description	The totality of all hardware and software components which build up the environment in which VNFs are deployed

A.1.2 MarketPlace Domain

Name	Marketplace
Description	The set of all tools and modules which facilitate the interactions among the T-NOVA actors, including service request, offering and provision, trading, service status presentation and configuration, SLA management and billing

Name	SLA Management Module
Description	The Marketplace functional entity which establishes and stores the SLAs among all the involved parties (Customer , SPs, and FPs, and checking if the SLAs have been fulfilled or not will inform the accounting system for the pertinent billable items (penalties or rewarding).

Name	Accounting Module
Description	The Marketplace functional entity which stores all the information needed for later billing for each user: usage resources for the different services, SLAs evaluations, etc.

Name	Billing Module
Description	The Markeplace functional entity that produces the bills based on the information store in the accounting module.

Name	Access Control Module
Description	The Marketplace functional entity which administers security in a multi-user environment, managing and enabling access authorization/control for the different T-NOVA stakeholders considering their roles and permissions.

Name	Dashboard
Description	The Marketplace functional entity which provides the web-based user front-end, exposing in a graphical manner all customer-facing services of th

Name	Brokerage Module
Description	The Marketplace functional entity which enables the interaction among actors for service advertisement, request and brokerage/trading.

A.1.3 Orchestration Domain

Name	Orchestrator*
Description	The highest-level infrastructure management entity which orchestrates network and IT management entities in order to compose and provision an end-to-end T-NOVA service.

Name	Resources Orchestrator*
Description	The Orchestrator functional entity which interacts with the infrastructure management plane in order to manage and monitor the IT and Network resources assigned to a T-NOVA service.

Name	NS Orchestrator*
Description	The Orchestrator functional entity in charge of the NS lifecycle management (i.e. on-boarding, instantiation, scaling, update, termination) which coordinates all other entities in order to establish and manage a T-NOVA service.

Name	VNF Manager*
Description	The Orchestrator functional entity in charge of VNF lifecycle management (i.e. installation, instantiation, allocation and relocation of resources, scaling, termination).

Name	NS Catalog*
Description	The Orchestrator entity which provides a repository of all the descriptors related to available T-NOVA services

Name	VNF Catalog*
Description	The Orchestrator entity which provides a repository with the descriptors of all available VNF Packages.

Name	NS & VNF Instances Record*
Description	The Orchestrator entity which provides a repository with information on all established T-NOVA services in terms of VNF instances (i.e. VNF records) and NS instances (i.e. NS records)

Name	NF Store
Description	The T-NOVA repository holding the images and the metadata of all available VNFs/VNFCs

A.1.4 Infrastructure Virtualisation and Management Domain (IVM)

Name	Virtualised Infrastructure Management (VIM)*
Description	The management entity which manages the virtualised (intra-NFVI-PoP) infrastructure based on instructions received from the Orchestrator

Name	Transport Network Management (TNM)
Description	The management entity which manages the transport network for interconnecting service endpoints and NFVI-PoPs e.g. geographically dispersed data centres

Name	VNF Manager Agent*
Description	The VIM functional entity which interfaces with the Orchestrator to expose VNF management capabilities

Name	Orchestrator Agent *
Description	The VIM/TNM functional entity which interfaces with the Orchestrator to expose resource management capabilities

Name	Hypervisor Controller*
Description	The VIM functional entity which controls the VIM Hypervisors for VM instantiation and management

Name	Compute Controller*
Description	The VIM functional entity which manages both physical resources and virtualised compute nodes

Name	Network Controller @ VIM*
Description	The VIM functional entity which instantiates and manages the virtual networks within the NFVIPoP, as well as traffic steering

Name	Network Controller @ TNM
Description	The TNM functional entity which instantiates and manages the virtual networks within the wide-area transport network, as well as traffic steering