NETWORK FUNCTIONS AS-A-SERVICE OVER VIRTUALISED INFRASTRUCTURES

GRANT AGREEMENT NO.: 619520

Deliverable D2.41

Specification of the Network Function Framework and T-NOVA Marketplace

Editor Aurora Ramos (ATOS)

Contributors Felicia Lobillo (ATOS), Thomas Pliakas (CLDST), Simon Hohberg, Yacine Rebahi (Fraunhofer), Marco Di Girolamo (HP), Paolo Comi, Fausto Andreotti (Italtel), Ahmed Abujoda, Panagiotis Papadimitriou (LUH), Emmanuil Kafetzakis , George Xilouris (NCSRD), José Bonnet (PTInS), Dora Christofi, Georgios Dimosthenous (PTL), Athina Bourdena, Evangelos Markakis, Evangelos Pallis (TEIC) Nicolas Herbaut, Mamadou Sidibe (VIO)

Version 1.0

Date September 30th, 2014

Distribution PUBLIC (PU)

Executive Summary

This report outlines the outputs of the activities carried out in Tasks 2.5 and 2.6 of the EU-FP7 Project Network T-NOVA "Functions as-a-Service over Virtualised Infrastructures", which are focused respectively on the specification of the Virtual Network Function Framework and the Marketplace. One of the starting points has been the previous work in T-NOVA project in which use cases, and requirements at system level where collected in Deliverable D2.1, and the overall T-NOVA architecture was despicted in Deliverable 2.21.

The Network Function Framework is the architectural element of the T-NOVA system devoted to the definition of the structure and behaviour of the Virtual Network Functions (VNFs). Moreover, T-NOVA comprises an innovative Network Function Store, where the VNFs are stored and made available to T-NOVA as building blocks for creating network services.

A VNF is characterised by two attributes: the operational functionalities and the management behavior. The operational part explicitly defines the network functions that are supported, while the management part is responsible for the VNF lifecycle. Therefore, a VNF in T-NOVA shall support the APIs for interacting with the T-NOVA orchestration and virtualized infrastructure, and shall support the VNF lifecycle described in this report. The VNF metadata is a fundamental part of each VNF adressed also in this document. It provides the information for describing what the VNF functionalities are and how to manage it. Currently there are many approaches for implementing this concept which are focused on the technical requirements for making a virtual application running. In T-NOVA this information is extended with business aspects that allow the registration and trading of a VNF in the marketplace.

The concept of the marketplace has been introduced by T-NOVA as a novelty in the NFV scheme ir order to facilitate the interaction between the different stakeholders that are identified in the NFV business scenarios. On one hand the VNFs can be implemented by a wide range of developers providing software implementation, and on the other hand, network service providers may want to acquire VNFs to compose network services to be provided to its own customers. The T-NOVA Marketplace has been designed as a distributed platform placed on top of the overall architecture which, besides of including the users front-end, it comprises OSS/BSS components as billing and accounting, and innovative modules as the T-NOVA Brokerage to allow trading functionality

The virtualization of network functions is addressed by notable standardization bodies such as ETSI and IETF. In particular ETSI has developed a NFV reference architecture and has provided a common language in this area. Therefore, this report looks at ETSI to build on its work trying to go a bit further with the aim of contributing to the evolution of standards.

When working in the T-NOVA Marketplace specification also the ongoing work of TMForum standardization activities have been considered, as it is for instance the integration of a business service catalogue and SLA Management issues in virtualization.

WORK IN PROGRESS

© T-NOVA Consortium

The current report describes the interim version of requirements, features description and architectural details in a technology-agnostic manner for later implementation to be done in WP5/6 which are respectively dedicated to the implementation of the Network Function Framework and the T-NOVA Marketplace adressing the details of the suitable technologies and their operation. The final version of this report will be released by September 2015 with the feedback of one year time of working in the implementation tasks.

WORK IN PROGRESS

© T-NOVA Consortium

Table of Contents

WORK IN PROGRESS

WORK IN PROGRESS

© T-NOVA Consortium

Index of Figures

WORK IN PROGRESS

© T-NOVA Consortium

Index of Tables

WORK IN PROGRESS

© T-NOVA Consortium

1. INTRODUCTION

1.1. Motivation and scope

Network Functions Virtualization (NFV) constitutes a topic of immense interest to the networking community in the the research/academic domain but also in industry since it is candidate approach for short-term exploitation. Via the concept of infrastructure "softwarisation", NFV has the potential to entirely transform the networking market and open it to new entrants. In this context, T-NOVA introduces a complete open solution for NFV deployment, focusing on the Virtual Network Function (VNF) as a service perspective with a strong business orientation.

In order to provide this business orientation to the NFV scheme T-NOVA will develop a novel marketplace that will facilitate T-NOVA customers to select virtual appliances by means of a friendly front-end, "plug" them into their existing connectivity services, configure/adapt them according to their needs and, in the case of network service p roviders, also allow them to offer Network Services (NSs) composed by several VNFs to their own customers [1].

The service request will be carried out via a tailored customer front-end/brokerage platform that is part of the T-NOVA Marketplace. This marketplace will also provide all the T-NOVA stakeholders SLA and billing functionalities.

On the other hand, T-NOVA introduces an innovative Network Function Store (NF store) following the paradigm of already successful OS-specific "App Stores". This NF store contains VNFs by third-party developers, published as independent entities and accompanied with the necessary metadata for both technical and business description of the VNF.

Software developers willing to sell their VNFs through the T-NOVA marketplace shall extend their implementation of network functions supporting the APIs for interacting with the virtualized infrastructure and the T-NOVA orchestration for the service composition [2], and the VNF lifecycle described in this deliverable.

In this way, thanks to the NF store and the marketplace, it is expected that T-NOVA will contribute to expand market opportunities by attracting new entrants to the networking market. This capability will be particularly important for SMEs and academic institutions which can leverage the T-NOVA architecture by developing innovative cutting-edge Network Functions (NFs) as software modules that can be included in the NF store. This also will enable the rapid introduction of VNFs into the market.

1.2. Relation to T-NOVA overall architecture

T-NOVA overall architecture is described in previous deliverables [3]. The scope of the current document with respect to the T-NOVA architecture is the definition and

WORK IN PROGRESS

specification of the (i) marketplace; (ii) network function store (NF store); (iii) VNFs and (iv) assorted interfaces as graphically illustrated in Figure 1-1.

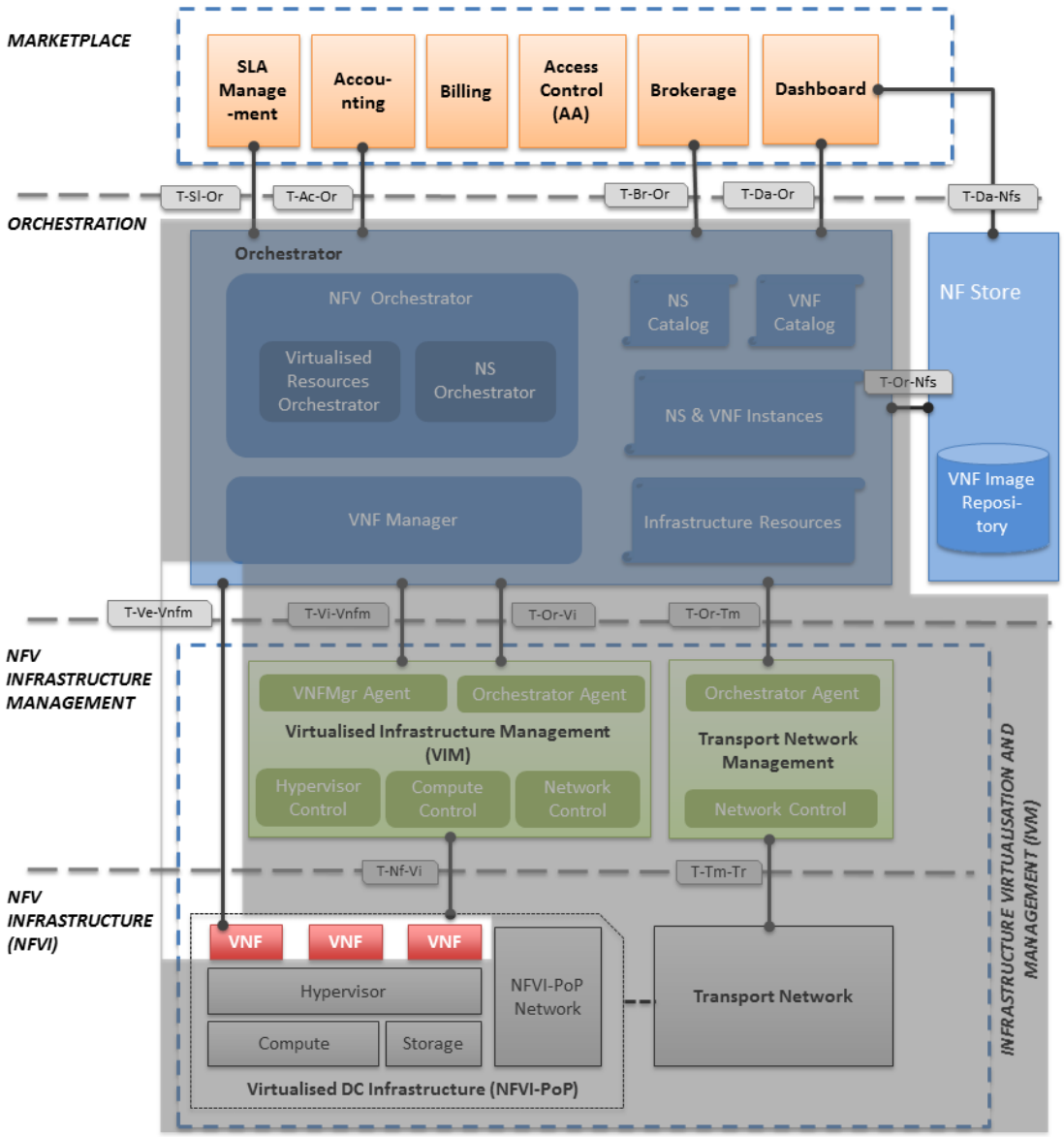


Figure 1-1 Relevance to the Overall Architecture

In summary, the T-NOVA Marketplace is a distributed platform placed on top of the overall architecture which, besides of including the users front-end, it comprises OSS/BSS components as billing and accounting, and innovative modules as the T-NOVA Brokerage, being in charge of managing all business relationships among the T-NOVA stakeholders (see section 2.3).

The NF store is mainly a repository for the VNF images and their accompanying metadata. It plays an important role in the VNF lifecycle management as it is the component where the VNFs are published by different NF developers.

VNF lifecycle is also part of the scope for this deliverable. More specifically, information related to VNF deployment, management and termination have

© T-NOVA Consortium

interdependencies with various components of the overall architecture. Other stages of the lifecycle refer to interaction between the marketplace and the NF store.

Finally, the interfaces as defined in [3] are also specify in the current document.

1.3. Document structure

This document is structured as follows:

Firstly, Section 2 focuses on the specification of the T-NOVA Marketplace. After a overall description of the objective in section 2.1, section 2.2 contains the state of the art analysis over which the specification of the T-NOVA Marketplace has been performed. Section 2.3 gives an overview on how each T-NOVA stakeholder will iteract with the system and Section 2.4 summarizes the T-NOVA Marketplace lifecycle in order to provide the reader a whole picture of the marketplace functioning. The requirements gathering procedure is explanained in section 2.5, while section 2.6 contains the proper specification of each component in the marketplace according to the requirements specifications listed in Annex A.

Section 3 is devoted to the specification of Virtual Network Function (VNF) structure and behaviour and the design of the Network Function Store (NF store). Section 3.1 explains the scope of the section providing high level description of the subset of the addressed T-NOVA architecture. The description of the VNF structure is the topic of section 3.2. It describes also the information model of the VNF metadata descriptor. A first comparative analysis with ETSI NFV is provided. Section 3.3 completes the VNF description with the definition of the network function lifecycle. The description of VNF structure and behaviour is completed by section 3.4 providing the high level specification of the APIs supported by both the VNF and the NF store. Finally, chapter 3.5 outlines the high level design of the NF store.

The objective of section 4 is to make clearer and remark the relation between the marketplace and NF store, both from a functional and architectural point of view.

Section 5 contains the conclusions gathered from the contents of this document.

Finally, Annex A provides the list of all the requirements gathered for the different marketplace components and NF store, while Annex B contains a first version of the mock-up of the dashboard that has been specified in this deliverable but that it will be properly designed in future work in the project.

WORK IN PROGRESS

© T-NOVA Consortium

2. Specification of the T-NOVA Marketplace

2.1. Objective

The marketplace in the NFV scheme is an innovative concept that T-NOVA introduces with the aim of promoting the VNF service offerings and facilitating the commercial activity and fluent interaction among the different business stakeholders identified in [1]. Besides of providing the Graphical User Interface (GUI) for all of the stakeholders, the T-NOVA Marketplace will facilitate all the necessary features related to the market activity, such as trading, SLA (Service Level Agreement) negotiations and billing.

The components identified in the previous work in T-NOVA [3] that are part of the T-NOVA Marketplace and represented in Figure 1-1 and high level described in Table 2-1:

Name	Description
SLA Management Module	The marketplace functional entity which establishes and stores the SLAs among all the involved parties and checking if the SLAs have been fulfilled or not will inform the accounting system for the pertinent billable items (penalties or rewarding).
Accounting Module	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.
Billing Module	The markeplace functional entity that produces the bills based on the information stored in the accounting module.
Access Control Module	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.
Dashboard	The marketplace functional entity which provides the user front-end, exposing in a graphical manner all customer-facing services.
Brokerage Module	The marketplace functional entity which enables the interaction among actors for service advertisement, request and brokerage/trading.

Table 2-1 Main T-NOVA Marketplace components definitions

2.2. State of the Art

In order to design and later implement the marketplace context in T-NOVA, we have firstly looked at the most relevant ongoing standardization works when applying network services provision business processes to Network Function Virtualization

WORK IN PROGRESS

(NFV). Though still not very advanced, TMForum has provided some first inputs mapping their standardization document about business process to the ETSI NFV MANO architecture [4]. This is explained in 2.2.1.

In section 2.2.2 we have collected the state of art in relation to recent research projects that implementing a marketplace to provide network services. In section 2.2.3 we summarize two commercial solutions.

Finally, in section 2.2.4 we extract some conclusions gathered from this state of art in relation to T-NOVA Marketplace.

2.2.1. Standardization activities

2.2.1.1. ETSI ISG NFV

A network operator-led Industry Specification Group (ISG) was setup in the last quarter of 2012 under the umbrella of ETSI (European Telecommunication Standard Institute) to work through the technical challenges of NFV. ETSI ISG NFV in its document on global architecture [5] illustrates the high-level NFV framework, where three main working domains can be identified:

- 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.

Relation to T-NOVA Marketplace

As it has been explained in the previous work to define overall T-NOVA architecture [3], and according to

Figure 2-1, the marketplace is completely novel in regards to ETSI view [4]. T-NOVA introduces the marketplace concept aiming at opening the NFV market to third party developers for the provision of VNFs, a concept that currently falls outside the technical view of ETSI NFV.

On the other hand and also introduced in [3] where ETSI MANO has been deeply explained, ETSI does not provide any more insight on the OSS/BSS (Operating

WORK IN PROGRESS

Support System / Business Support System) of the operator besides the definition of an interface. Though OSS/BSS systems are not inside the scope of T-NOVA, the proposed marketplace contains partially some OSS/BSS functionalities (i.e. billing, accounting, SLA monitoring, Authentication, Authorisation, and Accounting (AAA)) that will be implemented/adapted.

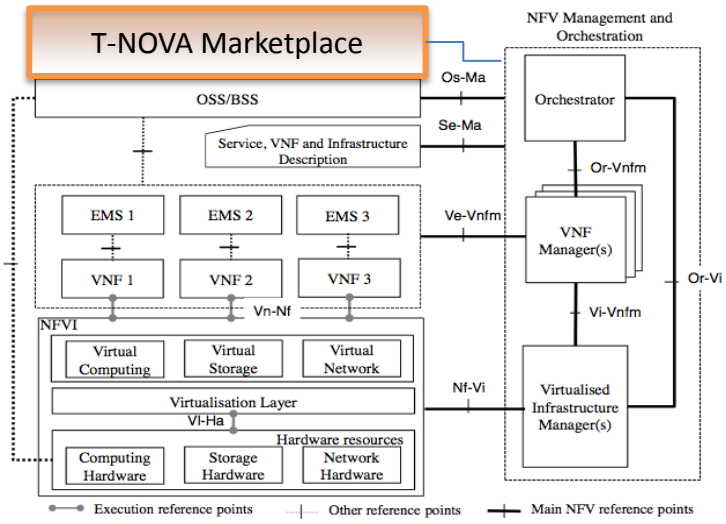


Figure 2-1 ETSI NFV architecture

2.2.1.2. TMForum

The general objective of Tele Management Forum [6], as a global trade association of service providers and suppliers, is the improvement on business agility and the growth of business through knowledge, tools, standard, training and best practices. The specific TM Forum's Agile Business and IT Program aims at optimize service providers' operations reducing costs, risks, and time to market by providing a set of integrated offerings that collects the experience and best practices gleaned from the major players within the industry. This vision is accompanied by a methodology for creating service oriented software solutions that adhere to the dynamic, loosely coupled and vendor-neutral principles demanded by today's service providers, enabling rapid service deployment and to drive down operational cost. The TMF's standards are collectively known as *Frameworx*, which is composed of four underlying components, each aimed at standardizing information models, interfaces, and lexicon:

- *Business Process Framework (eTOM, Telecom Operations Map):* the industry's common process architecture for both business and functional processes. This framework is meant to aid in the creation of a comprehensive, multi-layered view of all of the business processes necessary for a carrier's operation. It provides both guidelines and process flows, and aligns with standards from ITIL (*Information Technology Infrastructure Library*) and other external bodies.
- *Information Framework (SID, Shared Information/Data model):* provides a common reference model for enterprise information that service providers,

software providers and integrators use to describe management information. It is used to develop databases and provide a glossary of terms for business processes. The framework is intended to reduce integration costs and to reduce project management time and cost by minimizing the number of necessary changes to underlying architecture during the launch of a new product or service offering.

- *Application Framework (TAM, Telecom Application Map)*: it provides a common language between service providers and their suppliers to describe systems and their functions, as well as a common way of grouping them. It attempts to group the information and processes defined by the eTOM and the SID into recognizable applications.
- *Integration Framework (TIP, TM Forum Integration Program (TIP):* it shows how the business process, information and application frameworks interact to:
    - Create a catalog of business services that define functional and non-functional aspects of a service based on service oriented principles;
    - Develop a platform or domain-based enterprise architecture that provides the business agility required to compete in today's market;
    - Define critical standard interfaces that speed integration.

Relation to T-NOVA Marketplace

From the service provider point of view, the marketplace has been included into the T-NOVA system taking into account the approach proposed by TM Forum, as it is for instance the provision of business interaction agility for all the stakeholders, the creation of a business service catalogue to make the offerings easier for the customer, or the adoption a common language for service description that will be studied in future work in T-NOVA.

2.2.1.3. Applicability of TMForum standards to ETSI NFV

The TR227 TM Forum Specifications document [7] contains a description of the set of TM Forum documents that are relevant for ETSI NFV MANO work. It identifies areas where each TM Forum document may contribute to standarize the information presented and interfaces of the MANO reference points. Those that may be applicable to the specification of the marketplace are the following:

eTOM:

- Enables MANO to design interfaces and APIs that better reflect how an organization performs configuration, monitoring, and other processes.
- Enables MANO to achieve a better alignment between business processes of an organization and the process flows that are defined by MANO.
- Enables MANO to ensure that the reference points that it designs are appropriate for the business processes of an organization.

SID:

WORK IN PROGRESS

- Provides detailed models in an object-oriented form that can be used to further define MANO service and resource concepts.
- Provides a detailed model of how services and resources are managed, including definition of metrics to represent key characteristics and behaviour of services and resources as well as SLAs.
- Provides a framework to design interfaces and APIs for various business and operational processes.

TIP:

- Creates a common shared integration environment.
- Provides detailed interactions in the form of a set of messages exchanged as a protocol.
- Links business processes to information elements (e.g., the SID model elements).
- Defines standardized templates for the development of new interfaces.
- Increases interface reusability through addressing a broad set of business process scenarios.

Relation to T-NOVA Marketplace

Analysing the document TR228 TM Forum Gap Analysis related to MANO Work [8] we can gather that one of the topics that TM Forum points out as missing from NFV 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.

Though being out of the T-NOVA scope the interface between the MANO architecture and the existing OSS/BSS system of operators, T-NOVA aims to 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 external operator, and what could be a first input for latest studies in the interoperability with OSS/BSS existing systems, that TM Forum ZOOM intends to address in the future [6]. Other future work of TMForum that T-NOVA Marketplace is aligned to is the impact of the SLA management in virtualization.

2.2.2. Other projects

In this subsection we reference research projects implementing marketplaces for network services provision:

- *XIFI* [9]. This is a project of the European Public-Private-Partnership on Future Internet (FI-PPP) programme with the objective of facilitate the uptake, deployment and federation of several instances of a common market platform to pave the way for a unified European marketplace that is crucial for enabling commercial exploitation of FI resources. Inputs from this project in relation to service description languages and catalogues could be considered in T-NOVA.

WORK IN PROGRESS

- *Bandwidth Exchange* [10]. This is a TM Forum catalyst project that demonstrates how operators can control and monetize their network bandwidth capabililities by securitising their bandwidth as an exchange tradable commodity. They created a commercially viable bandwidth exchange marketplace, which involves trading options and futures on bandwidth like other commodities on a typical commodity exchange (such as the Chicago Merchantile Exchange) and a spot market (such as FX market) where the transaction involves a physical settlement.

- *Service Bundling in a B2B2X Marketplace* [11]. This is a TM Forum catalyst project that will aim to show how a buyer can bundle a collection of services sourced from different suppliers and deliver them seamlessly to a customer. These components could include traditional network access products, as well as NFV and IaaS products. Concrete business roles and process touchpoints enable a well-defined relationship among players in the value chain to ensure seamless delivery.

2.2.3. Commercial products

We have identified two commercial products that may be related partially to the T-NOVA Marketplace:

- *CENX Ethernet Lifecycle Manager (ELM)* [12] is a software based solution that automates lifecycle management specific to carrier Ethernet and IP services across data network infrastructures and evolving the concept of SDN/NFV. It provides an accurate and actionable visual representation of all inter-carrier Ethernet services, in order to allow Service Providers, Access Providers and Cloud Exchange Providers to deliver quality connectivity services. Some features are ordering automation, big data analytics, service orchestration and delivery across multiple operator networks and visualisation of transport services end-to-end.

- *Equinix Platform - Marketplace* [13] is a platform where users can buy and sell connectivity among other services. Provides service offering capabilities for sellers and buyers have access to wide area of services across multiple operators.

2.2.4. Conclusions

Analysing the state of art, we have found some solutions from which we can build on in order to develop the T-NOVA Marketplace; however at this stage it does not exist a proper marketplace to deliver VNF as a Service.

The T-NOVA Marketplace will be designed considering the above solutions, the current NFV ETSI architecture, the on-going TMForum Best Practices for business

WORK IN PROGRESS

services delivery, and we will be aware of its future project "Service Bundling in a B2B2X Marketplace" when taking implementation decisions.

2.3. T-NOVA stakeholders interacting with the Marketplace

Starting from the business roles analysis performed in Deliverable D2.1 [1], we concluded that all of the T–NOVA roles identified can be grouped to be played by three different stakeholders which we have called the basic stakeholders in the T-NOVA landscape. For simplicity it has been decided from the implementation point of view that only the basic stakeholders will be considered when specifying the T-NOVA system. Nevertheless, interesting future work for T-NOVA system in general, and for T-NOVA Marketplace in particular, will be its extension for multiple non-basic T-NOVA stakeholders.

Therefore, three different stakeholders will exploit the T-NOVA Marketplace: Customer, Service Provider (SP), and Function Providers (FPs). This situation is reflected in Figure 2-2.

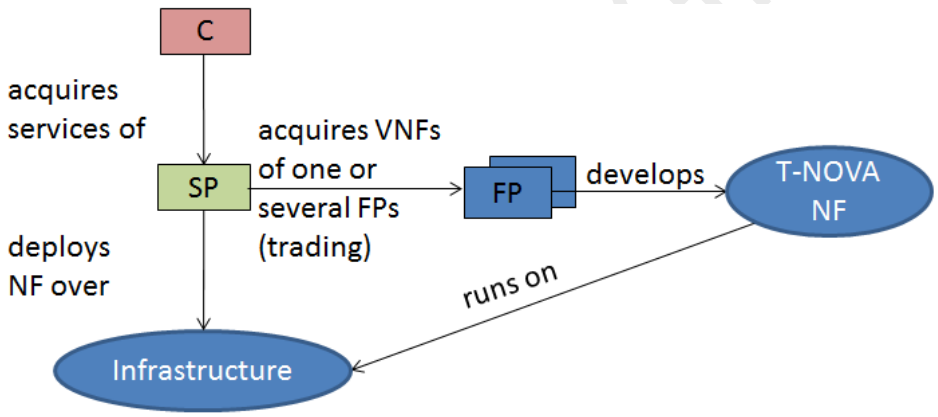


Figure 2-2. Stakeholders interacting in T-NOVA Marketplace

The T-NOVA Marketplace will facilitate the following functionalities for the T-NOVA stakeholders:

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

© T-NOVA Consortium

Besides these three basic stakeholders, T-NOVA will have its own T-NOVA operator that will be in charge of the system, and the access control for the rest of the stakeholders.

2.3.1. Subscription management

As the T-NOVA Marketplace will be accessed by three previous main categories of users or stakeholders, and each of these categories has a specific task, it is crucial that the marketplace implements a Role Based Access Control (RBAC) framework that:

- restricts account access only to authorized users,
- links a user to an account and assigns the user to some specific roles,
- links each role to specific permissions or profile. For instance, a FP will be allowed to upload VNFs in the function store or to upgrade them. If this provider wants to purchase a service, it will also be assigned the role of a customer. In other words, this framework will allow users to perform actions according to the assigned roles.

As a first step, the stakeholders need to register on the T-NOVA system. Here typical information such as provider name, user name, address, wished role, email address, etc., will be requested in order to create an account on T-NOVA for this user. If this user wishes to use an existing account (on Google, Yahoo, etc) for authenticating himself, an account on the T-NOVA system will be automatically created for him in (see details in section 2.6.2.2.).

2.3.2. Trading mechanisms

One of the added values of the T-NOVA marketplace is the possibility of auctioning among several FPs. In this way, the SP, and T-NOVA customers in general, will benefit from this functionality receiving the best price option possible, according to their requirements of service description and SLA level.

The principal objective of the T-NOVA auction is to trade the VNFs between the SP and several FPs to achieve best price option for T-NOVA customer on a competitive basis, while maximising the revenue that the auctioneer raises and efficiently exploiting the VNFs. Given those objectives, the design of T-NOVA auction system must also address some challenges an auctioneer faces. Briefly, these challenges include winner's curse (this situation occurs when one or more users pay too much for the auctioned item(s)), collusion (most of the times caused in small size markets) and signaling (less signaling simplify the auction process and attract bidders).

Considering trading models, two main approaches exist, the reserve price estimation (or fix-price) and the auctions. The former case refers to the lowest price the seller is willing to accept for a given item (i.e. service), without performing bids. The latter case performs bids based on single, multiunit or combinatorial auctions. There are plenty of auction protocols, including the sealed first-price auction, the sealed second-price auction (called Vickerey auction), the open ascending-price (English)

WORK IN PROGRESS

auction and the open descending-price (Dutch) auction. The main ones are collected in Table 2-2. More information regarding the auction theory and the types of auctions can be found in [14]. In T-NOVA marketplace the auctioning will be performed by beans of the brokerage module (see section 2.6.2.3.).

Trading Mechanisms	Usage
Fix-price	In cases that the offer of NFs is lower to the demand
Auctions	In cases that the offer of NFs is higher to the demand
Vickerey/Sealed bid auction	• Optimize social welfare (the price at which charged the winner depends on the bids of competitors and none of the individual offer) • Avoid signaling • Truethfull in case of second price
English auction	• Quite profitable type of auction for the seller/ Reach to large amounts • Winner's curse is not possible to be prevented (bidders overestimate the value of the item) • Requires signaling • Not truethfull
Dutch auction	• Very profitable type of auction for the seller. The purchase price was not allowed to drop too much, due to the fear that another bidder will first acquire the item • Requires signalling • Not truethfull
Combination of English/Dutch and Vickery auctions	• Decrease the collusion
Combinatorial auction - Vickerey-Clark-Grooves	• Sell package of items – winner claim all or nothing • Truethfull • Very Complex

Table 2-2 Overview of Trading mechanisms

2.4. T-NOVA marketplace use cases - lifecycle

Based on the T-NOVA use cases in collected in [1], we describe in this section the whole marketplace lifecyle for a customer accessing the system to purchase VNFaaS, or Network Services (NSs) based on VNFs. This general procedure is represented in Figure 2-3.

WORK IN PROGRESS

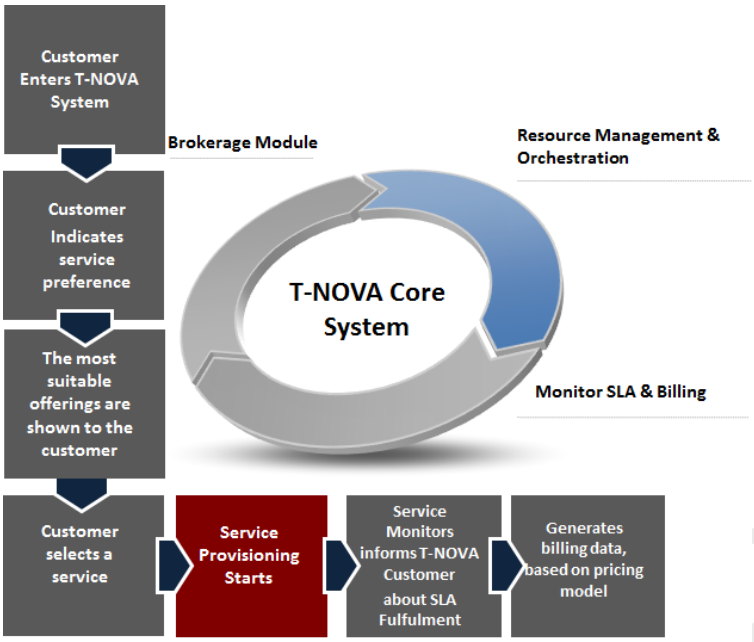


Figure 2-3 Marketplace lifecycle

Background:

- The Function Providers (FPs) that want to sell their VNFs through T-NOVA system enter the system providing their VNFs required information (according to sections 3.3.2 and 3.4.2).
- The Service Provider (SP) that want to purchase VNFs in order to later sell NSs based on VNFs through T-NOVA enter the system.
- A business service catalogue is filled offline in the marketplace when a new service composition has taken place triggered by the SP. This business service catalogue will contain all the information of the commercial offerings available (more information in 2.6.2.4. Business Service Catalogue).

1. The T-NOVA customer enters T-NOVA system through the dashboard identifying (or registering if it is its first time).

2. The T-NOVA customer indicates the service (VNF, several VNFs or Network Service) he/she would like to purchase (perform search).

3. The most suitable offerings available in the marketplace business service catalogue will be shown to the customer in order for him to select one, including price and SLA options.

In the event that there is not any available service offering in the business service catalogue matching the customer request, a new service composition should have to take place and trading mechanisms will be performed among FPs if several FPs offer similar VNFs dynamically.

5. The customer selects one of the offered services, together with the configuration of specific technical configuration parameters needed for the service provision, and the SLA agreement procedure will be initiated.

WORK IN PROGRESS

6. All the related information is stored in the different marketplace modules (customer profile, SLA, accounting, etc.).

7. Service provisioning starts.

8. Service monitoring information, SLA fulfilment information and billing information will be made available through the dashboard each time it is required by the customer when accessing the T-NOVA marketplace.

8. Billing will take place when finishing the services in pay-as-you-go services, or each time the assigned bill cycle finishes for the rest of services (between customer and SP, but also between SP and FPs).

WORK IN PROGRESS

© T-NOVA Consortium

2.5. Requirements for T-NOVA Marketplace

The requirements capture process has focused on identifying the desired behaviour for the T-NOVA marketplace and its components, which were most of them identified based on the previous requirements analysis performed at T-NOVA system level [1].

None of the marketplace components requirements in this deliverable specifies how they will be implemented; implementation details will be specified in the future work in technical T-NOVA WorkPackages (WPs) as the implementation-specific descriptions are not considered to be requirements. The goal of these requirements is to develop an understanding of what the marketplace components need, how they interact between each other, and their relationship to the overall T-NOVA architecture [3]. Additionally, the T-NOVA use cases [1] were also considered and cross-referenced with marketplace components requirements.

Requirements were primarily anchored to the existing T-NOVA use cases and the interactions with the whole system both in terms of the actions and requests that would be expected. Additionally the high-level data/information that is be required by the marketplace to successfully deploy its functionalities was also identified. Identified requirements were primarily functional since they are related to the behaviour that is expected from the marketplace.

Using a systems' engineering approach the high level architecture for the marketplace was described in Deliverable 2.21 [3], each component of the overall system was specified in terms of high-level functional blocks. This approach identified the following functional blocks:

- Dashboard
- Access Control
- Brokerage module
- SLA management module
- Accounting module
- Billing module

Also a Business Service Catalogue has been identified to be part of the Marketplace matching TMForum proposal por business agility.

The T-NOVA Marketplace components requirements can be found in Annex A - Requirements Specification as well as the requirements describing how they should interface one with one another. These requirements were used as a foundational input into the specification of the overall marketplace architecture and its constituent components, which are presented in section 2.6. The coverage of these requirements by the T-NOVA system will be explanained after implementation work.

WORK IN PROGRESS

© T-NOVA Consortium

2.6. Specification of the T-NOVA Marketplace architecture: components and interfaces

Based on the requirements performed at system level [1], the initial approach for the marketplace architecture included in Deliverable 2.2 [3], and the requirements gathered for each component in the marketplace, the overall diagram for the T-NOVA marketplace architecture with both the external and internal interfaces, is depicted in Figure 2-4:

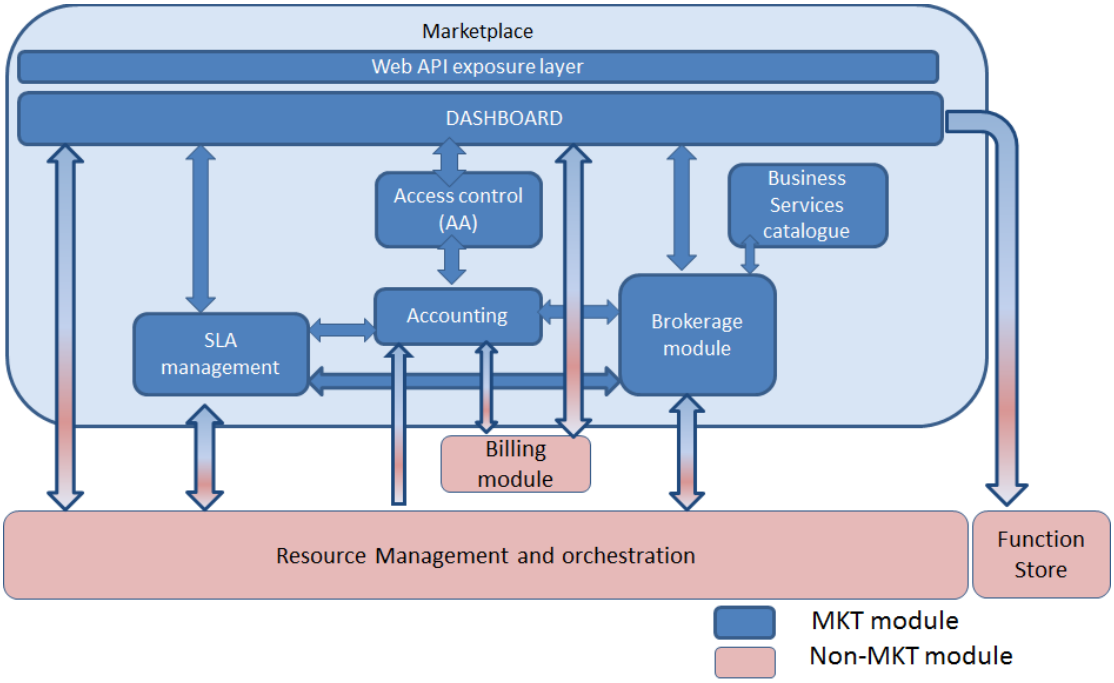


Figure 2-4 Marketplace architecture

The specification of each module's functionality, internal architecture and their interfaces are addressed in sections 2.6.3 to 2.6.8. Section 2.6.2 refers to the external interfaces of the marketplace.

2.6.1. External Interfaces to the T-NOVA Marketplace

The marketplace modules will communicate with other three T-NOVA components: the orchestrator, billing module and the function store.

2.6.1.1. Orchestrator

The T-NOVA Orchestrator, which specification can be found in Deliverable 2.31 [2], deals with the optimal deployment of network services instances, as requested by the customer or the Service Provider (SP) on the marketplace, according to a yet to be designed algorithm, the required SLA and the current status of the available infrastructure.

© T-NOVA Consortium

While all network services instances have been instantiated and are running, it is also the orchestrator's responsibility to follow the available metrics, both from the infrastructure and from the service metrics. In order to meet the agreed SLAs, the orchestrator may scale out or up the supporting insfrastructure, communicating such changes to the marketplace, so that a change in accounting is registered and later billed to the customer. Later, if the scaled (out or up) infrastructure is perceived as being more than enough to fulfill the SLA, it can be scaled in or down. Throught all this process, the orchestrator must provide the marketplace with meaningfull metrics showing how Network Service (NS) instances are working. Details of the interfaces between the orchestrator and marketplace modules are explained in the following sections: Dashboard, Brokerage module, Accounting module and SLA management module.

2.6.1.2. Billing system

Though the billing system is part of the features implemented by the marketplace, we are considering it as an external element, since for simplicity it has been decided to use an existing opensource billing module for the T-NOVA marketplace. Therefore the novelty of the billing procedure in T-NOVA does not lie in the invoicing itself but in the whole architecture between brokerage pricing mechanisms, SLA management module billable items, and the accounting system that will feed the billing existing module.

In this way, the accounting module (2.6.2.6.) will be in charge of sending to the billing application all the information that it may need. In future work in T-NOVA (task 6.4), the different existing billing opensource options will be studied in order to choose the more suitable one.

2.6.1.3. Network Function Store (NF store)

As it will be explanained in section 3.5, this T-NOVA component will store the VNFs images and metadata that the marketplace, more concretely the brokerage module, will use to perform trading mechanisms among Function Providers (FPs), to later include those VNFs in the service composition process performed by the orchestrator.

For a VNF to be part of a service composition process, it is necessary that the orchestrator makes it available, according to T-NOVA Orchestration specification in D2.31 [2]. Whenever a VNF is uploaded, updated or removed from the NF Store, the orchestrator is informed in order to update its internal registers. This process makes the VNF available for the brokerage module and therefore available to the T-NOVA system.

WORK IN PROGRESS

2.6.2. Marketplace modules specification

2.6.2.1. Dashboard

With the aim of creating a single entry to the T-NOVA system that provides simplicity for the different T-NOVA users or stakeholders, a unified T-NOVA Dashboard will be designed, taking into account the different roles of the T-NOVA Marketplace. This common dashboard for the whole T-NOVA environment will host three views for the three basic stakeholders that will access the T-NOVA Marketplace: the Service Provider (SP), the Function Provider (FP) and the Customer.

Starting from the dashboard requirements in Annex A, in this section we include the general description of the information that the dashboard will have to show, first ideas for its design and the general information that will have to be collected by the different APIs of the dashboard coming from the rest of the T-NOVA components. The complete design information and later implementation of T-NOVA user dashboard will be done in T6.3, and its outcomes will be gathered in the Deliverables D6.01 and D6.3.

Functionality

The main features of the dashboard are presented in Figure 2-5 . Each view is allocated with specific functionalities stemming from the requirements gathered from Annex 6.1.1.

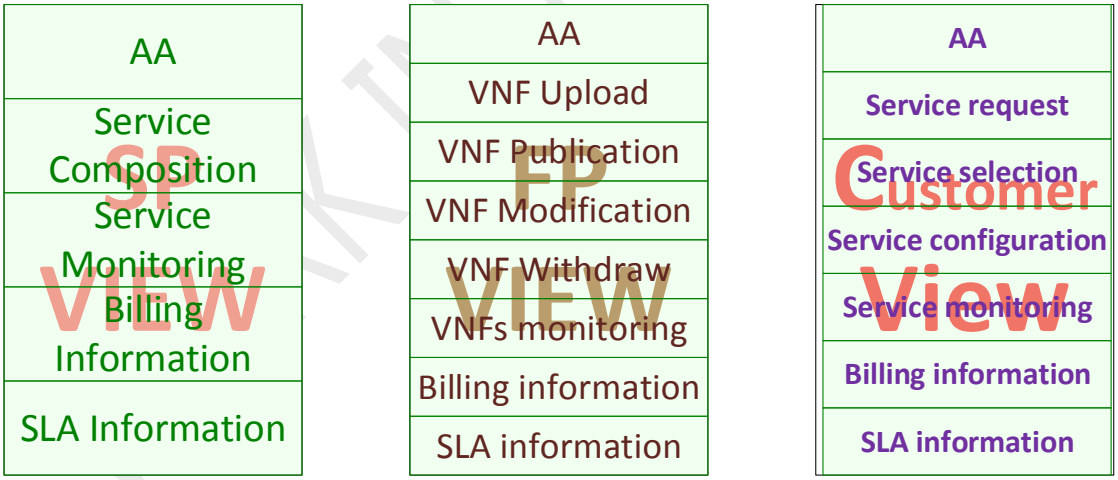


Figure 2-5 Dashboard views

The SP view of the dashboard will allow the SP to perform the functionalities shown in Table 2-3.

Functionality	Short Explanation
AA	Authorization and Authentication of the respective role into the T-NOVA Dashboard.

© T-NOVA Consortium

Service composition	Graphical wizard that will help the SP to compose a new Network Service (NS) starting from the brokerage among the FPs owing the available VNFs.
Service monitoring	Graphical representation of all monitoring data for a selected or "consumed" Service.
Billing information	Graphical representation of the billing outcomes of selected or "consumed" service. There will be two types of billing information for the SP: - Charges for the SP's customers (BSS functionality). - Invoices on behalf of its own suppliers, the FPs.
SLA information	Details of the selected or "consumed" service based on how they respect the agreed SLA. The SP will have accessed to two different kinds of SLA contract and SLA monitoring information: - SLA between SP and its customers (BSS) - SLA agreed with his its suppliers, the FPs

Table 2-3 SP dashboard view

The FP view of the dashboard will allow the FPs to perform the functionalities shown in Table 2-4.

Functionality	Short Explanation
AA	Authorization and Authentication of the respective role into the T-NOVA dashboard.
VNF Upload	Graphical wizard that will help the FP to upload his VNF with the necessary parameters.
VNF Publication	Graphical representation for the FP to provide the last check in order to publish the uploaded VNF
VNF Modification	Small graphical wizard that provides the ability to the FP to modify the uploaded VNF.
VNF Withdraw	Graphical representation that gives to the FP the ability to remove an already published or uploaded VNF
VNFs monitoring	Graphical representation of all monitoring data for a selected or "consumed" NF.
Billing information	Graphical representation of the Billing outcomes for a selected or "consumed" NF.
SLA information	Information of the selected or "consumed" NFs based on the agreed SLA and its fulfilment.

Table 2-4 FP dashboard view

WORK IN PROGRESS

The customer view of the dashboard will allow the customer to perform the functionalities shown in Table 2-5.

Functionality	Short Explanation
AA	Authorization and Authentication of the respective role into the T-NOVA Dashboard.
Service request	Graphical representation of the Services/Functions returned by the T-NOVA business service catalogue.
Service Selection	Graphical representation assisted by a check box providing the ability to the customer to select a service for consumption.
Service configuration	Small Graphical wizard providing to the customer predefined parameters for defining the selected service.
Service monitoring	Graphical representation of the data gathered from the monitoring modules.
Billing information	Graphical representation of the billing outcomes of selected or "consumed" service.
SLA information	Details of the selected or "consumed" Service based on how they respect the agreed SLA.

Table 2-5 Customer dahsboard view

Design

The dashboard constitutes the T-NOVA system front-end, as offered to the Customer, the SP and the FPs for service consumption, discovery, interaction, publication, etc. In order for the dashboard to be as up-to-date as possible and terminal-agnostic, a web-based design has been selected.

Furthermore, the dashboard shall be able to meet and, if necessary, to adapt to the specific stakeholder's needs/requirements as much as possible providing the best experience to a specific stakeholder. This implies that the implementation shall achieve a flexible service presentation by means of an appropriate choice of technologies and tools. T-NOVA will allow every role to personalise some settings such as interface, appearance and content according to its profile.

More specifically, in the authentication stage, all stakeholders share a common layout that displays the generic graphical interface composed by the basic controls that enable stakeholder specific authentication. Once authenticated, every stakeholder will be able to customize the overall experience according to a set of preferences and his profile.

The main design decision gathered from the requirements in [1] has been to have a common dashboard with different customized views based on different roles. Furthermore and by gathering all the requeriments we have designed a first version of a mock-up for the dashboard that will be used as a Pilot for the upcoming work in Task 6.1 – User Dasboard. The mock-up is available in Annex B 6.2.

WORK IN PROGRESS

Interfaces

The information that will be collected from the rest of T-NOVA components to be used by dashboard will be provided through the following APIs:

AA: the Authentication and Authorization access control system will provide an API to the dashboard to provide and collect all the information necessary to authenticate the T-NOVA users or stakeholders.

SLA management: the goal of this API is to show the users the following information coming from the SLA management module:

- SLA template specification to be filled by the SP and FPs.
- SLA offering to the customer and associated to each service.
- SLA fulfilments by all the stakeholders.

Note: the SLA selection performed by the customer to manage the SLA negotiation process and the SLA contract information will come from the brokerage module that performs the trading.

The SLA front-end tool that will be integrated in the dashboard will be also responsible to make the correct request to the SLA API and then gather and show the results.

Brokerage

This interface is exploited for trading issues, among the T-NOVA users (i.e. C, SP, FP) and the brokerage module. The information that will go through this API will be related to:

- Service request and selection: by means of this API customer requests and selections will be sent to the brokerage module.
- Service composition/VNF request: this functionality enables the SP to request network functions or compose a new service, considering customer's requests.
- Advertise VNF: this functionality is exploited for the communication between FP and the brokerage module, as the latter perform the intermediate communication, this is trading.

Orchestrator

The interface between the dashboard and the orchestrator will be used to manage the following information:

- Notify about a new/update/deleted Network Service (NS) after a successful new NS composition, update or need to remove the NS, the SP uses the dashboard to notify the orchestrator about that NS.
- Notify about a new/updated/deleted NS subscription: after a successful subscription, update on configuration or need to remove a running NS, the customer uses the dashboard to notify the orchestrator about the need to instantiate that NS.
- Receive service usage data: through this interface the SP and customer will be able to get the monitoring information of the service.

WORK IN PROGRESS

© T-NOVA Consortium

Billing

The billing API for the dashboard will have to manage the following information between dashboard and billing module:

- Bills charged per user and per service (SP and customer).
- Charges done to SP's customers (BSS functionality to the SP).
- Charges done to FP's customers, which is the SP.

Network Function Store

This interface allows the FPs to publish and manage their VNFs into the NF Store. The publication consists in uploading the VNF image, registering the VNF and its metadata into the function store. The VNFs are versioned allowing the FPs to provide further upgrades. Finally, the FPs can remove their VNFs. In summary, the information managed with this interface is:

- VNF image and VNF metadata descriptor.
- VNF version.
- Upload, upgrade and delete the VNF package.

2.6.2.2. Access control (AA)

In T-NOVA, different stakeholders are foreseen. Each of these stakeholders will have a specific role and accordingly some associated permissions (see below in this section *Policy Enforcement Service*). For instance, a Function Provider (FP) will be able to upload a VNF and upgrade it if needed. A T-NOVA customer will be able to select the VNFs that he is willing to deploy and use, and should not be allowed to upload/remove a given VNF from the NF Store. One of the main challenges in T-NOVA is how to administer security in a multi-user environment. To address this issue, T-NOVA will specify and develop a lightweight Role Based Access Control (RBAC) system where decisions are based on the functions a given stakeholder is allowed to perform within T-NOVA.

The requirements gathered for this module are collected in Annex A. The main conclusions are summarized in the following bullets:

- The different stakeholders should be authenticated before any operation on the T-NOVA system.
- The different stakeholders should be authorized to perform tasks that are associated to their roles and permissions.
- Roles are created according to their functions in T-NOVA, and stakeholders are assigned roles based on their responsibilities and qualifications.
- Roles can be reassigned or granted new permissions if needed.
- Roles and permissions should be updatable and revocable.

Functionality

The RBAC system will be offering two main functionalities:

WORK IN PROGRESS

- *Authentication:* authentication is the process by which the system will verify that a user of T-NOVA is exactly who he is claiming to be.
- *Authorization:* authorization is the process by which a user is allowed to perform the tasks he wants to.

Architecture

The general diagram of the access control system is depicted by Figure 2-6.

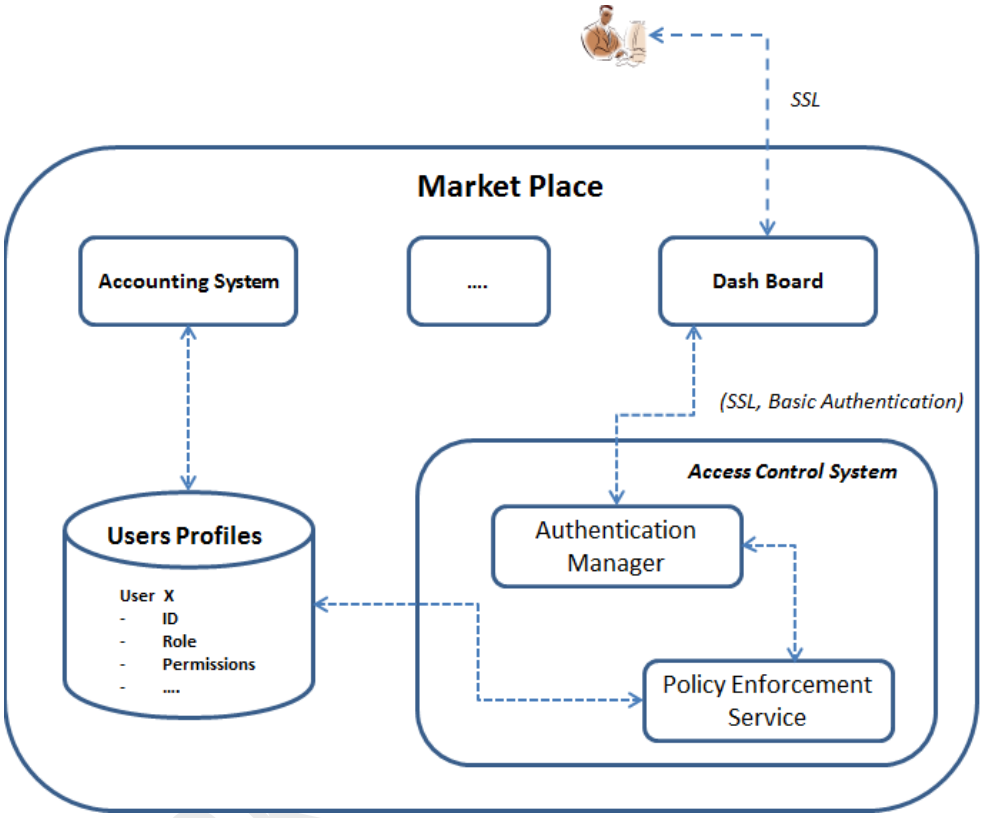


Figure 2-6. RBAC high level architecture

Authentication Manager

To enable the T-NOVA system to provide different functionalities to the stakeholders, a mechanism for authenticating a stakeholder is required. In T-NOVA, this is performed by the *Authentication Manager,* allowing a user to register and login with username and password or Open ID. For convenience and user friendliness purposes, the T-NOVA system will support a technology like OpenID (which is a technology that enables a person to use his account on Yahoo or Facebook for instance to access other web sites) in addition to classical accounts that will be created on the T-NOVA system. To be more concrete, when a user registers, he has to decide whether to use Open ID or to create an account on T-NOVA. When selecting Open ID the required user information is automatically retrieved. In the classic registration case, the user has to provide this information manually by filling out a registration form. Also, using Open ID does not require setting a password, but just the ID of the Open ID account. This information is then sent to the *Authentication Manager* to register the new user.

© T-NOVA Consortium

Finally the *Authentication Manager* returns a user object for the new user that contains an authentication token reflecting that the user is logged in.

Policy Enforcement Service

The *Policy Enforcement* web service implements a Role Based Access Control (RBAC) mechanism that allows assigning users different roles resulting in different rights. Such an access control mechanism allows the T-NOVA system to implement functionality such as uploading a VNF or purchase a service. When a new user registers a user profile will be created containing the name and email address of the user. Furthermore the profile will also contain the current role of the user.

The roles foreseen at this stage of the project are:

- T-NOVA operator: in charge of the T-NOVA system
- Service Provider: it purchases several VNFs to compose a service to be sold to its final customers.
- Function Providers: the entities that are allowed to upload and upgrade a given VNF on the T-NOVA system.
- Customer: the entity interested in purchasing a T-NOVA service.

Interfaces

Several interfaces are foreseen to ease the communication with the other parts of the T-NOVA system. This includes:

- *Interface to the dashboard:* the T-NOVA Access Control module will provide an API to the dashboard allowing it to authenticate the T-NOVA users.
- *Interfaces to the accounting system:* the T-NOVA Access Control System will provide an API to access the "User profiles" database in the accounting system as some features are needed to handle the accounting, such as:
  - User ID.
  - Token reflecting the authorization session (when it starts and when it finishes).
  - A signature for preventing that the user modifies the token.

2.6.2.3. Brokerage module

Towards facilitating trading between diverse actors in the NFV scene the T-NOVA marketplace includes an innovative brokerage module, in which VNFs by several Function Providers (FPs) can be brokered/traded.

Functionality

Via the brokerage module API in the dashboard, the customers place their requests for T-NOVA services and declare their requirements for the corresponding VNFs, receive offerings and make the appropriate selections, taking into account the price and the offered SLAs. Trading policies such as long-term lease, scheduled lease,

WORK IN PROGRESS

short-term lease or spot markets (these leasing types refere to the duration of VNFs exploitation) can be based either on fixed-price or action-based strategies (see section 2.3.2 Trading mechanisms).

In T-NOVA there are several objectives in order to select an auction mechanism, which should be taken into account. The first objective is to avoid too much signalling overhead. This objective may be satisfied with the *sealed-bid auction*. In this auction scheme, bidders simultaneously submit sealed bids so that no bidder knows the bid of any other participant. Hence, bidders cannot change their bids after the announcement of the other bids. In the case of sealed-bid auction the first price auction model should be implemented. Sealed-bid auction may not be truthful (truthfulness prevents market manipulation, since the bidding is performed considering the true value of the item), however the VNFs auctions are often organized to maximize the payoff, and not to be truthful.

To be precise, the implementation of the second price auction model is also possible since there is no problem in switching the payment method in an auction engine (this may be an optional feature implemented in the T-NOVA Brokerage Module), thus, making this auction truthful. The T-NOVA Brokerage Module may change the pricing rule in a flexible manner. It is a matter of implementing an extra policy in the brokerage module operation mechanism. Additionally, the call price may be used to provide rational item valuation. The brokerage module will determine the proper call price for each VNF based on marketing factors. It is also possible that the bidders use their own valuation tools along with both the brokerage module, so that the former (i.e. bidders) to be able to learn the optimum call price. Future work for auctioning implementation will be done in Task 6.3 – Brokerage module.

In summary, the brokerage module will provide the following functionalities:

- VNF discovery: this process is required in order the brokerage module to seek for the requested VNF.
- Service composition: this process is required in case that there is not any ready available service and the brokerage module request for new service composition.
- Service matching: with this process the brokerage module tries to find the required services in the business service catalogue.
- Trading: this process enable the brokerage module to trade the VNFs, especially through auctions, in case that one VNF is offered by more than one FP. Figure 2-7 depicts the sequence diagram of general auction trading.
    - Note: the trading is performed typically in T-NOVA between the SP and FPs, if the SP is an external stakeholder issuing the T-NOVA system as a particular case of T-NOVA customer. In case the T-NOVA operator is acting as T-NOVA Service Provider, the trading would be performed in the same way between the T-NOVA customer and Function Providers.

WORK IN PROGRESS

© T-NOVA Consortium

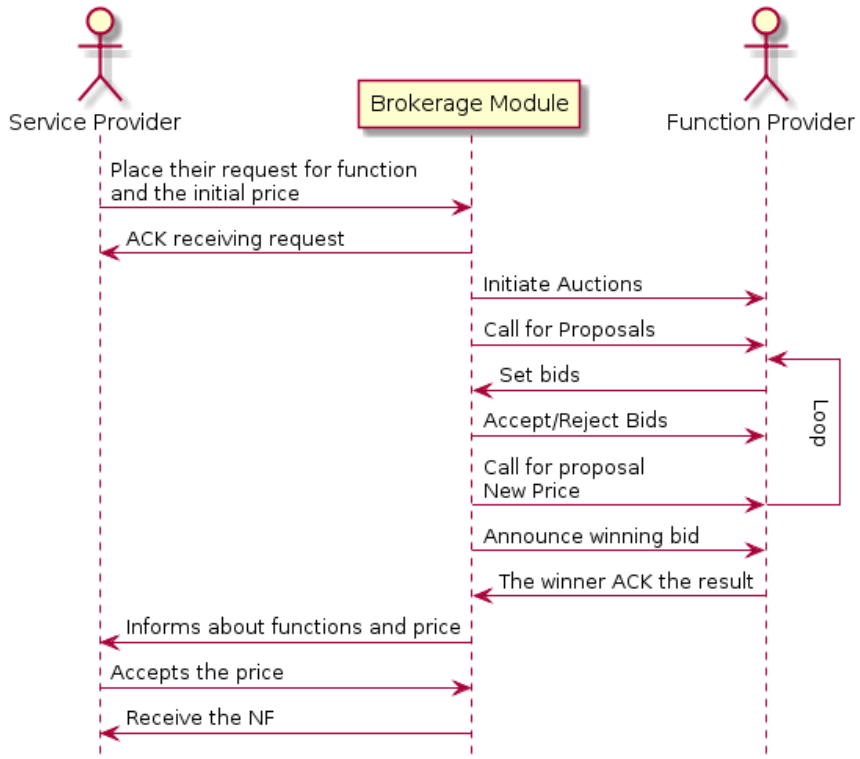


Figure 2-7 Trading process

1. The SP provides to the brokerage module the VNF request and the initial price.
2. The brokerage module sends an ACK that initiate auctions.
3. The brokerage module informs the FPs regarding the request and the initial price
4. FP sends their bids for the functions (Price + SLA specification)
5. The brokerage module solves an auction to maximize its revenue.
6. The brokerage module informs the bid results.
7. Depending on the type of auction, an iteration (3-6) continues until the bid winner is found.
8. The brokerage module announces the final results.
9. The winner acknowledges the results.
10. The brokerage module indicates the the VNF's price, which is provided by the FP that won the bidding, to the SP.
11. The SP accepts the price and SLA.
12. The SP receives the VNF.
13. (Price will be stored in the accounting module, and SLA agreement in the SLA management module).

Architecture

The overall architecture of the brokerage module and its interfaces is depicted in Figure 2-8.

© T-NOVA Consortium

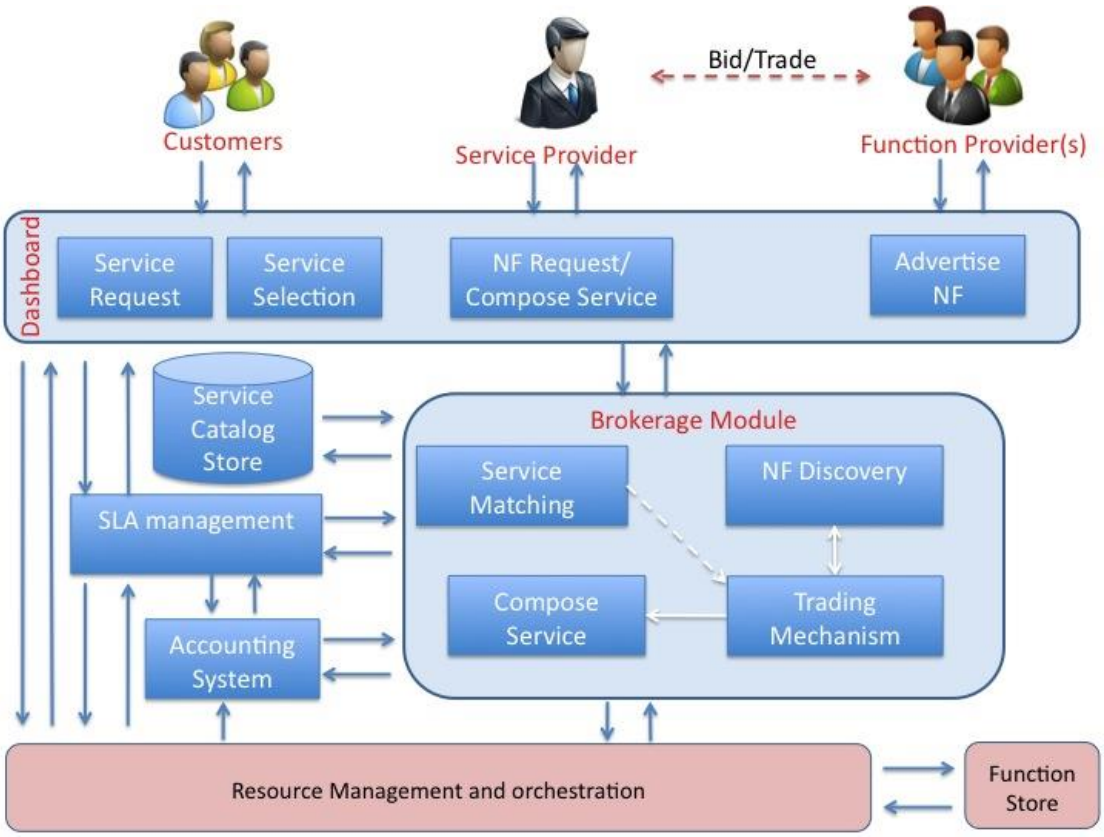


Figure 2-8 Brokerage module internal architecture

Through the dashboard the customers initially request the service. The details of the request are used by the brokerage module to query the business service catalogue for the most suitable offerings. The matching result is presented to the customer. In case that this service is not readily available in the business service catalogue, the customer is guided through the new service creation/composition procedure. Then the communication is initiated between the customer and the SP, in order the latter to request for the required VNFs. The process of trading between SP and FPs is then initiated according to the sequence diagram depicted in Figure 2-7.

Interfaces

The required interfaces of the brokerage module for the proper communication with the other parts of the T-NOVA system are:

- *Interface to the dashboard*: this interface is required in order for the users of T-NOVA system (i.e. Customers, SP, FPs) to be allowed to trade. For this purpose, functionalities such as service request, service selection by the customer, service composition/VNF request by the SP and advertise VNF/trading by the FPs are exploited.
- *Interface to the business service catalogue*: this interface is required in order for the brokerage module to have access to the catalogue that stores all the services. Considering customer's preferences the brokerage module seeks in the business service catalogue to find the appropriate service (in case that the service is already stored).

- *Interface to the accounting module*: with this interface the brokerage module provide to the accounting system the appropriate information related to service selections and price in order this information to be forwarded to the billing system.
- *Interface to the SLA management module*: this interface is exploited in order for the brokerage module to provide information to the SLA management module regarding the SLA agreed between SP and FPs as a result of the trading process. (The SLA management module requires such information in order to create and store the SLA contract and for SLA monitoring issues).
- *Interface to the orchestrator*: this interface is required in order for the brokerage module to retrieve information about the available VNFs for a service composition.

2.6.2.4. Business Service Catalogue

According to the system requirements gathered in Deliverable D2.1 [1], in order to a T-NOVA user (typically the customer) to be able to easily know the services already available in the T-NOVA system, and access the description of those services, the T-NOVA Marketplace will store all this information in what we have called the "business service catalogue", matching also with the approach suggested by TM Forum in its "integration framework", in which functional and non-functional aspects of a service based on service oriented principles are defined. This work will be continued in T-NOVA under Task 6.1 – service description and T6.2 – brokerage module.

Starting from the requirements in Annex A - Requirements Specification for this component, which are mainly related to the need of the catalogue to be browsable, including the service description, SLAs offered by the Service Providers and price for each services&SLA, we explain next its functionality and the way the information will be stored.

Functionality

The business service catalogue will be used by the Service Provider (SP), who by means of the brokerage module, will be able to search in the catalogue the most suitable service offering to be shown to the customer according to its requests.

The business service catalogue will be filled with the service offering information manually and offline after a service composition has taken place by the SP through the orchestrator and the pricing through the brokerage module.

Design

In order to structure the information in a simple way, the business service catalogue will have entries containing what we have named "offering", each of them will composed by: service description + SLA offer + price, according what

Figure 2-9 shows.

WORK IN PROGRESS

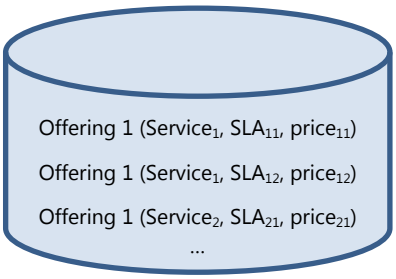


Figure 2-9 Business Service Catalogue

Interfaces

- Brokerage module: the business service catalogue will be accessed directly and only by the brokerage module in both read and write mode.

2.6.2.5. SLA management module

Service Level Agreements (SLAs) represents a contractual relationship between a service consumer and a service provider in order to provide a mechanism to increase trust in providers by encoding dependability commitments and ensuring the level of Quality of Service is maintained to an acceptable level.

In T-NOVA there will be a SLA agreed between Function Provider (FP) and Service Provider (FP) and between the Service Provider and its customers, and per each service, since the same service could have different SLA levels associated. One VNF can be offered by a FP with different SLA levels, according to different prices that will be negotiated through the trading mechanisms implemented by the brokerage module (see section2.6.2.3.). On the other hand, one Network Service (NS) could have also different SLA levels with different prices, being part of different offerings as it is represented in Table 2-6:

SLA per service
service1, SLA11, price11
service1, SLA12, price12
service2, SLA21, price21
service2, SLA22, price22

Table 2-6 SLA per service

SLAs describe the service that is delivered, its properties and the obligations of each party involved. Moreover, SLAs establish that in case the guarantee is fulfilled or violated, rewards or penalties, monetary or not , can be applied, respectively. T-NOVA SLA management module will provide information for later accounting, depending on the terms and conditions gathered in the SLA and on whether this SLA has been met by all parties or not.

WORK IN PROGRESS

The requirements for the T-NOVA SLA management module, listed in Annex A, are mainly related to the need to provide mechanisms to get an agreement presented and agreed, store all the SLA agreements, to inform the orchestrator, and to know all the SLA fulfilment to inform the billing system for possible penalties.

Functionality

The SLA management module is in charge of providing mechanisms to get an agreement presented and agreed, informing the involved parties (Customer, SP, and FPs) and storing the SLAs, it will later receive and will process all measurements related to the SLA from the monitoring system (in the orchestrator) and, checking if the SLAs have been fulfilled or not, will inform the accounting system for the pertinent billable items (penalties or rewardings).

A SLA basically consists of two main steps:

1. Paper-signed contract, in this case, between the customer and the SP, and between the SP and FPs, including the description of the quality of service and the penalties to be applied (could also be on a web site by agreeing terms and conditions).
2. eContract: It is automatically negotiated between parties for each customer, depending on the demand. Always based on a paper-signed framework contract (step 1).

The SLA management module needs to be able to provide the following functionalities: publication, discovery and negotiation of SLAs requirements, in order to manage the SLA lifecycle that can be split in the following phases:

1. *SLA Template Specification*: for the SP (and FPs), a clear step-by-step procedure describing how to write an SLA template to provide a correct service description.
2. *Publication and Discovery*: publish the provider offer and possibility for the customer to browse/ compare offers.
3. *Negotiation*: agreement on SLA conditions between the customer and the SP and between the SP and the FPs. This could be a bargain-like transaction or simply a combo list selection of predefined choices when the customer selects a specific offering.
4. *Resource Selection*: depending on the chosen SLA for every service, the SP selects the resources that need to be assigned to the service in order to meet this SLA. (As an option the resources could be explicitly selected by the customer as well, depending on the policy).
5. *Monitoring and Evaluation of the SLA*: comparing all the terms of the signed SLA with the metrics provided by the monitoring system (from the orchestrator), in order to internally prevent upcoming violations.
6. *Accounting*: invoking the charging/billing system according to the result to inform about billable items as penalties or rewards.

WORK IN PROGRESS

© T-NOVA Consortium

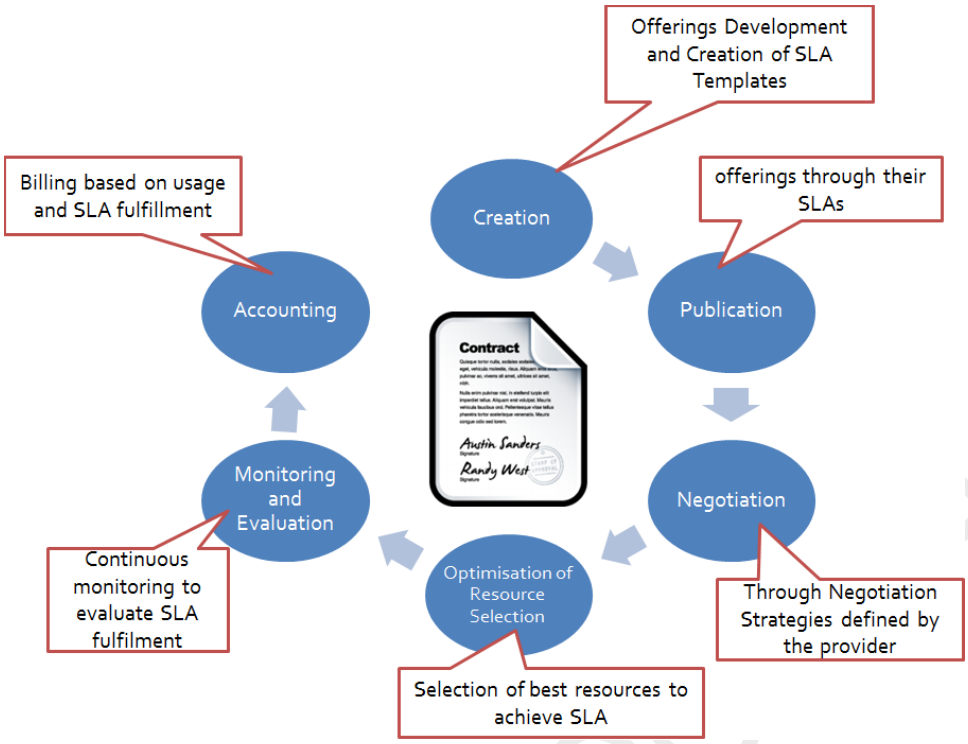


Figure 2-10 SLA lifecycle

Design

The information that shall be stored in the SLA management module is high level described in Table 2-7.

Item	**SLA template specification**	**SLA contract** *Parties involved* *Parameters* *Penalties*	**SLA fulfilment**	**Billable items**
	Input from the SP and FP dashboard	Input from the brokerage module as the output of the SLA negotiation	Input from the the monitoring system in the orchestrator	Output of the SLA management module (to be sent to the accounting module)
VNF ID				
Service ID				

Table 2-7 SLA management module information

© T-NOVA Consortium

Interfaces

So far, several interfaces are foreseen to ease the communication with the other parts of the T-NOVA system. These are:

- *Interface to the dashboard:* the SLA management module will provide an API to the dashboard to show the pertinent SLA information (template specification, contract, SLA fulfilment, etc.).
- *Interfaces to the accounting system:* the SLA management module will inform the accounting module about billable items as penalties or rewards when the SLA has not been achieved.
- *Interface to the brokerage module:* the SLA management module will be notified by the brokerage module about the SLA selected by the SP when purchasing a VNF and by the customer when selecting the Network Service (NS).
- *Interface to the orchestrator:* the SLA management module will inform the orchestrator about the terms agreed on the SLA, and will receive from the monitoring system of the orchestrator the monitoring information required to determine the level of fulfilment of the SLA for each service.

2.6.2.6. Accounting module

Functionality

The accounting module in T-NOVA will be in charge of registering all the business relationships (subscriptions, SLA evaluations and usage) that will be needed for billing. The way of storing and providing this information as well as the study of different options for billing opensource applications will be refined in later work (Task 6.4). Once that a billing opensource application is chosen for T-NOVA system the accounting module will be adapted to it.

Design

The high-level information that shall be stored in the accounting module is described in Table 2-8.

User ID	User Role-permissions	Service ID/VNF ID	Service Usage	Pricing information	SLA billable items
	Input from AA module	Identification in the business service catalogue (service) or in the Function Store (VNF)	Input from monitoring system in the orchestrator	Input from the brokerage module	Input from the SLA management module

Table 2-8 Accounting module information

WORK IN PROGRESS

Interfaces

- *Interfaces to the SLA management module:* the SLA management module will provide the accounting module about billable items as penalties or rewards when the SLA has not been achieved.
- *Interface to the brokerage module*: the brokerage module will provide the accounting system with the appropriate information related to prices for the service selections in order for this information to be forwarded to the billing system.
- *Interfaces to the access control:* the access control system will provide an API to access the "user profiles" database as some features are needed to handle the accounting.
- *Interface to the orchestrator*: the monitoring system in the orchestrator will use this interface to register in the accounting system the usage performed by the different services.
- *Interface to the billing module*: by means of this external interface the billing module will get all the information it will need to issue a bill.

WORK IN PROGRESS

© T-NOVA Consortium

3. Network Function Framework

This section is devoted to the definition and specification of the Network Function Store (NFS). This architectural element in T-NOVA contains the VNFs' (Virtual Network Functions) software images and their metadata description. This section provides also the high level description of VNF design. Finally, it describes the lifecycle of a VNF.

In summary the content of this section is:

- Design of VNF common components.
- Study of Network Function Lifecycle.
- Specification of NF APIs.
- Design of Network Function Store.

We recognise ETSI NFV as a notable body working on VNF specification. In this document we will make several references to ETSI work. We will also compare T-NOVA with ETSI NFV with the aim of identifying synergies and gaps.

3.1. NFS High level description

The Network Function Store (NF Store) contains all the information of the VNFs provided by different software developers, Function Providers (FPs) that want to offer them through the T-NOVA marketplace. The NF Store is maintained by the T-NOVA Operator.

The NF Store contains VNFs:

- provided by several (third-party) developers,
- published as independent entities,
- accompanied with the necessary metadata.

The diagram in figure 3-1 provides a very high level architectural description of the relationships of NF Store and VNFs with the other elements of T-NOVA architecture.

WORK IN PROGRESS

© T-NOVA Consortium

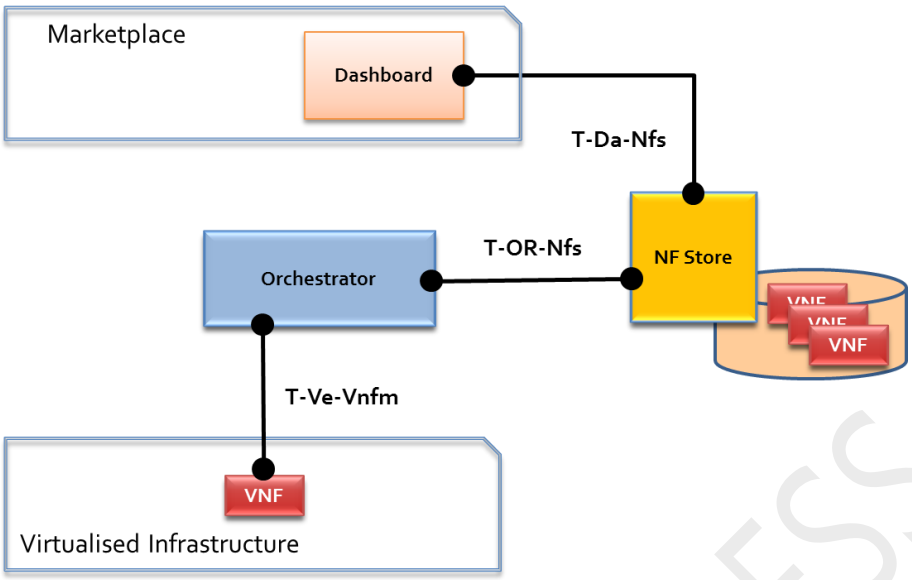


Figure 3-1. VNF framework high level architecture

The VNFs reside in the NF store. They are selected via the marketplace dashboard, and then instantiated and deployed over the execution platform (NFV Infrastructure). The T-NOVA orchestrator, specifically the NFV Manager, controls the execution of VNFs over the NFV Infrastructure.

The NF Store is mainly a repository for the VNFs and their metadata. A VNF can be versioned. Whenever a new VNF is added to the repository, the NF Store informs the orchestrator. Then the orchestrator retrieves the VNF image and metadata from the NF Store. The same procedure applies for VNF removal, or for any other modification to a VNF in the NF Store.

According to this approach the T-NOVA Orchestrator can be considered a single point of knowledge, meaning that the information shall always be easily and effectively accessible to the orchestrator.

3.2. NF Common Components

This introductory chapter describes the components and behaviour of T-NOVA VNFs. All the topics will be deeply analysed in dedicated sections. The specific details for implementation will be described in later work by the technical workpackages (from WP3 to WP6).

3.2.1. NF structure and properties

Generally speaking, a Virtual Network Function (VNF) is an executable software program that implements the whole or part of some network functions and it can be hosted in a virtualised platform. VNFs are located between computer machines (virtualised or not) and physical networking devices, and are transparent to the computer nodes. In other words, from the computer nodes' point of view, it seems that they interact with physical networking devices.

© T-NOVA Consortium

Taking as a fact the quick convergence of computing, storage and networks and the very high volume of standard servers that are shipped, the idea is to exploit the flexibility and agility of software implementations in comparison with hardware proprietary boxes. In this way, the launch of new telecom services in short time is relatively easy. Most of custom hardware appliances (e.g., network processors, digital signal processors, firewalls, etc.) can be implemented as VNFs and moved where and when needed.

In the scope of T-NOVA project, a VNF is a virtual machine or a group of virtual machines that implement network functions in software. The VNF high level architectural model is represented in figure 3-2. A VNF is characterised by two attributes: the operational functionalities and the management behavior. The operational part explicitly defines the network functions that are supported, while the management part is responsible for the VNF lifecycle (i.e. start, stop, pause, scaling, etc.).

Starting from the operational part, a VNF application may support one or more application network interfaces for communicating with other VNFs in application layer. Incoming data traffic and outcoming data traffic (after being processed by the VNF) are passed through the Virtual Network Interfaces (VNIs).

On the other hand, two specific interfaces exist for management purposes. More specifically, the VNF management socket interacts with Virtualised Infrastructure Manager(s) -VIM(s)- and the second management interface communicates with the T-NOVA orchestrator framework (see T-NOVA deliverable D2.21 [3]for more details).

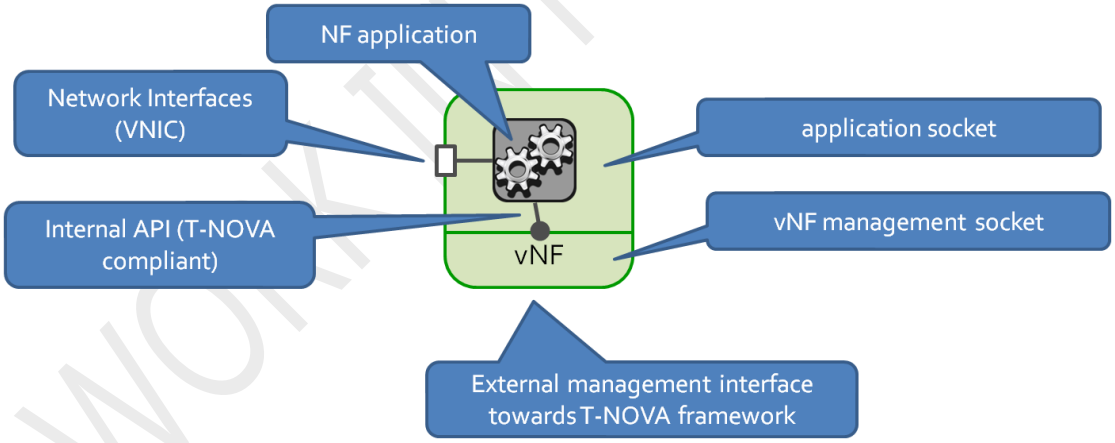


Figure 3-2. VNF high-level structure

3.2.1.1. VNF composition

A VNF can be composed from the interconnection of more than one NF applications. In this case, each NF is called a component of the VNF and is referred as Network Function Component (NFC). A Network Function Component is actually implemented by a Virtual Machine.

WORK IN PROGRESS

The diagram in figure 3-3 represents a VNF composed by two VNFCs (i.e. Service Chaining case), one of them exposing one network interface, while the other being internal to the VNF.

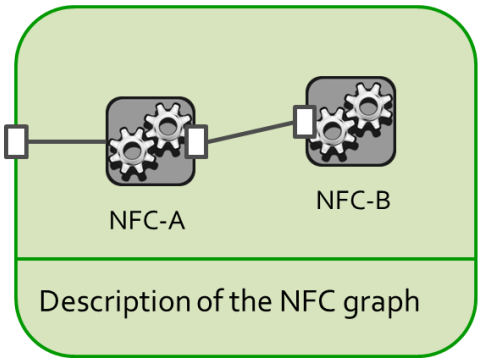


Figure 3-3. VNF composition

Proper VNF Descriptors and VNFC Descriptors should be in place, that indicate operational and management behaviors, used in launching, initiating and terminating a VNF. For example, the VNF Descriptors pose the minimum VNF infrastructure resource requirements that a VNF instance needs. Moreover, VNF Descriptors provide details about the instance topology and VNF instance lifecycle operations. In addition, the VNF provides the description of the group of VNFC that it contains. Note that a VNF should be a self-contained entity, since a direct interaction with T-NOVA Orchestrator is provisioned by the T-NOVA System architecture.

The VNF and VNFC properties, contained in corresponding descriptors, will be exposed through the VNF Management socket.

In T-NOVA, all relevant information will be expressed using a metadata language. Since metadata permits the autonomous operation of each VNF and VNFC, direct interaction between T-NOVA Orchestrator and VNFCs located inside of VNFs has been considered. The interaction with the marketplace is indirect through metadata descriptors.

Service graph or forwarding graph

In another direction, the usage of metadata for VNF description permits the service chaining and the service insertion, building E2E services by composition. The VNF Forwarding Graph visualises this interconnection of VNFs and the traffic flow between them. More specifically, more than one VNF can be connected for providing a new service, and this connection can be visualised through a graph. The T-NOVA Orchestrator can use the VNF descriptors for creating this service graph or forwarding graph by interacting with the VNICs of the VNFs. An example of a Service Graph is presented in figure 3-4.

WORK IN PROGRESS

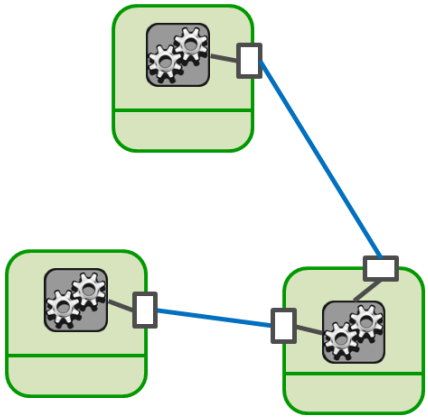


Figure 3-4. VNF Forwarding Graph

3.2.2. Metadata in T-NOVA

The VNF metadata descriptor provides the information for describing what the VNF functionalities are and how to manage them. Many approaches are currently available on how to implement this template, such as:

- Amazon AWS CloudFormation [15]
- OpenStack Heat Orchestration Template (HOT) [16]
- ETSI VNF Descriptor Template [4]

They provide the technical information for configuring, running and operating the VNF, i.e. the virtual machines with the software application images providing services in a cloud environment.

In T-NOVA we will extend this information with business relevant aspects that allow the registration and trading of a VNF in the marketplace.

The VNF Descriptors in T-NOVA will include:

- the VNF name and the version;
- the vendor of the VNF;
- the VNF behaviour (i.e., the description of what the VNF does);
- an SLA description offer;
- a price offer;
- the reference to the software image associated with the VNF;
- the minimum required hardware resources (e.g. CPU unit numbers, virtual memory size, virtual disk size);
- the network interfaces (i.e., the connection points with the network);
- selected performance metrics (i.e. how VNF shall perform in typical configurations);
- the default deployment parameters;
- some statistics (i.e. monitoring data generated during the active state of the VNF);
- information about how to properly dimension the VNF (i.e., scaling procedures);

WORK IN PROGRESS

- possible functional or network constraints;

From the moment that all metadata of a VNF will be available to the T-NOVA platform, the marketplace can use operational and business information for selecting the most suitable VNF according to customer's needs and willingness to pay. At the same time, the orchestrator will exploit the metadata for managing the VNF according to the VNF lifecycle.

Besides this high level approach, an in-depth analysis about metadata structure and template creation will be made in future work in workpackages WP3, WP4, WP5, and WP6.. The outcomes of the in-depth analysis will be integrated in the future release of this document.

3.2.3. T-NOVA NF Framework and ETSI NFV comparison

This paragraph makes a comparison between the Virtual Network Function (VNF) framework that will be used in T-NOVA and the one specified by ETSI.

In ETSI NFV specifications, the objective of NFV is to separate software that defines the network function (the VNF) from hardware and software used to create generic hosting functions infrastructure which executes the VNF (the NFVI: Network Function Virtualization Infrastructure). Therefore, the requirement is that VNF and NFVI are separated, as depicted in Figure 3-5.

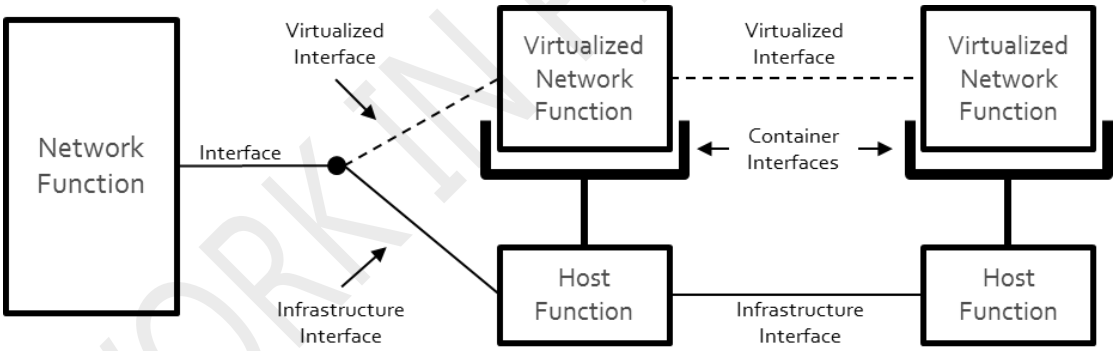


Figure 3-5. Virtualisation of network functions in ETSI

As a result:

- the Network Function functional block is split into a host function plus a Virtualized Network Function (VNF);
- a container interface between the VNF and the host is created;
- the NF Interface is split into virtualized and infrastructure interfaces;
- the interfaces between non-virtualized and virtualized network functions are homogenous.

An important concept must be taken into account: the VNF cannot be considered a functional block independent of its host function. The VNF cannot exist

WORK IN PROGRESS

autonomously, since it is an abstract view of the host function when configured by VNF.

The NFV architecture is thus not defined by functional blocks but using the following entities:

- Host Functions and their associated container and infrastructure interfaces;
- VNFs with their associated container and virtualized interfaces.

Regarding management and orchestration, the ETSI framework is split into management and orchestration of infrastructure (NFVI) and network functions (VNFs), as depicted in figure 3-6. A similar approach for management and orchestration has been adopted in T-NOVA, as described in [3] and [2].

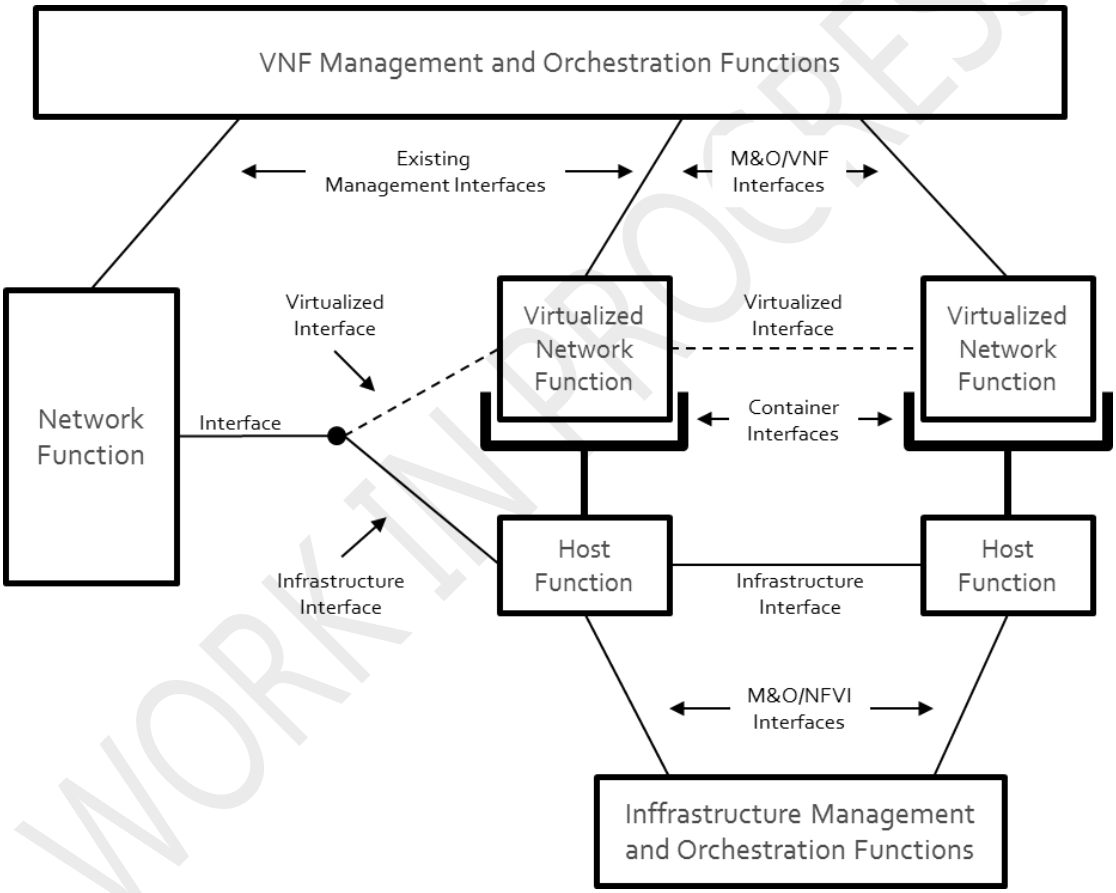


Figure 3-6. Management and orchestration of NFVs in ETSI

With particular regard to the structure of the VNF, which is the focus of this section of the document, ETSI specifies a number of interfaces, which are relevant to the VNF software architecture, as depicted in figure 3-7.

WORK IN PROGRESS

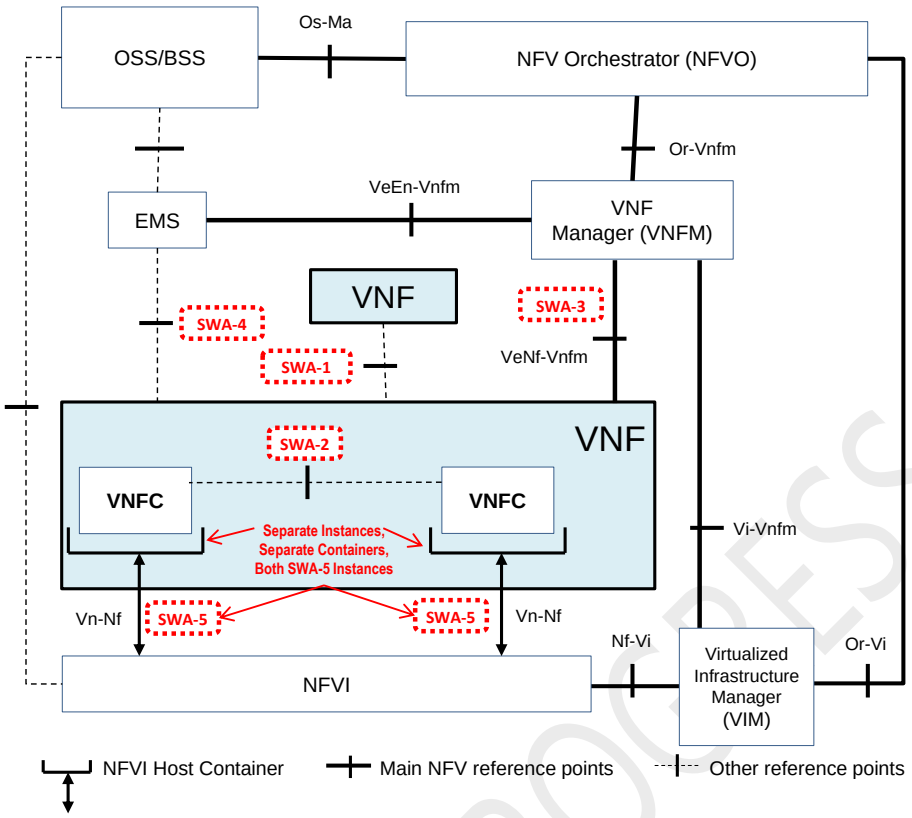


Figure 3-7. NFV architectural framework and interfaces in ETSI

In the following list, a brief explanation of these interfaces is provided.

- SWA-1: this is a well-defined interface used to connect various network functions in a forwarding graph.
- SWA-2: the SWA-2 interfaces refer to VNF internal interfaces, i.e. for VNFC to VNFC communication. In fact, a VNF can be decomposed and made up from sub-parts or components (VNFCs) which are themselves VNFs interconnected by the infrastructure.
- SWA-3: the SWA-3 interfaces the VNF with the NFV management and orchestration, specifically with the VNFM.
- SWA-4: the interface SWA-4 is used by the EMS to communicate with a VNF.
- SWA-5: SWA-5 corresponds to VNF-NFVI container interfaces.

In T-NOVA, the VNF structure has been completely described in paragraph 3.2.1. Figure 3-8 shows the T-NOVA VNF high level structure and the mapping with the VNF's interfaces specified by ETSI.

WORK IN PROGRESS

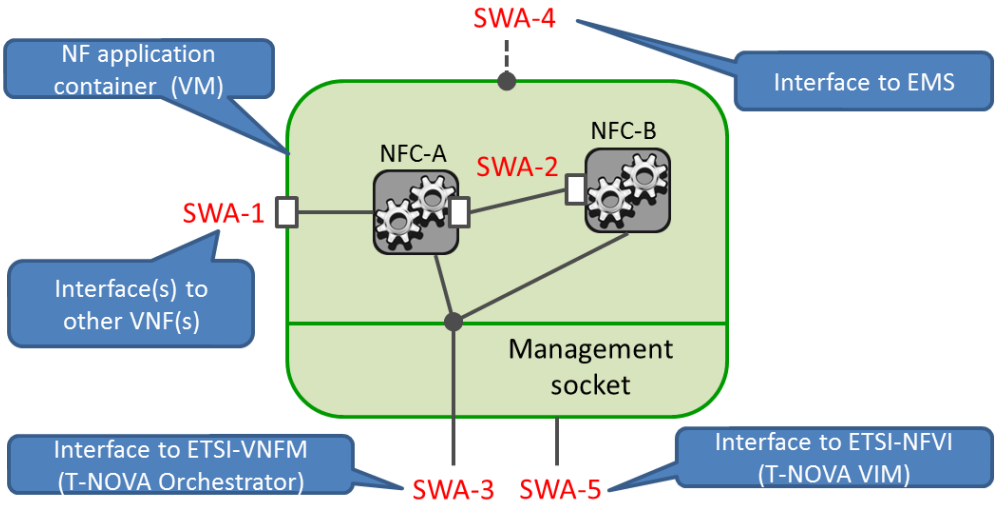


Figure 3-8. T-NOVA NFV structure mapping to ETSI framework

It is worth noting that ETSI interfaces SWA-1 and SWA-2 correspond to interfaces used in T-NOVA for communication between VNFs and VNFCs, respectively. Regarding management, the T-NOVA framework is aligned with ETSI SWA-3 and SWA-5 as well, by adopting a couple of interfaces for the control of VNFs from the T-NOVA Orchestrator and the control of infrastructure by the T-NOVA VIM. The interface SWA-4 with the Element Management System (EMS) is not further specified by ETSI because it is considered already present in current network architectures. For the same reason in T-NOVA we consider this interface as available.

Besides the comparison of the ETSI and T-NOVA VNF frameworks, a brief explanation of common patterns in VNF design and operations defined by ETSI and the corresponding support of relevant features in T-NOVA is provided in Table 3-1.

ETSI VNF feature		*Description*	**T-NOVA support**
VNF design patterns	VNF Internal Structure	VNF(C) composition is possible	Supported.
	VNF Instantiation	Each VNFC of a VNF is either parallelizable or non-parallelizable.	Supported.
	VNFC States	Each VNFC of a VNF may need to handle state information.	Supported.
	VNF Load Balancing Models	There are different types of load balancing, typically 4 models are identified (internal, external, e2e and infrastructure).	Ready to support it. Not studied in the deliverable.
	VNF Scaling	Auto, on-demand and	Driven by

WORK IN PROGRESS

ETSI VNF feature		*Description*	T-NOVA support
	Models	management based scaling.	T-NOVA Orchestrator.
	VNF Component Re-Use	Different models have been studied regarding component re-use but only one case has been agreed as relevant for ETSI NFV.	Component re-use is possible and encouraged. Not studied in details because out-of scope for T-NOVA.
VNF Update and Upgrade	VNF Update and Upgrade Overview	The key difference is that VNF upgrades may require planning on network service level, while VNF updates can be deployed without coordination with other VNFs participating in the same VNFFG.	Update and upgrade require creating new service instance using updated VNFs versions available in the Function Store.
	VNF Update & Upgrade Requirements for VNF Provider	The VNF package shall provide an automatic procedure for upgrade/update of the VNF instance. The procedure shall support control of the progress of this process, including the allocation of virtual resources from the NFV management and orchestration. Roll-back will be supported as well.	Same as above.
VNF's Properties	Hardware Independence	HW dependent, partly COTS, COTS.	Only COTS considered until now.
	Virtualization and Container Awareness	Hypervisor (agnostic, dependent), OS containers, HL container tech, no virtualized & no container, partly virtualized.	Hypervisor agnostic.

WORK IN PROGRESS

ETSI VNF feature		*Description*	**T-NOVA support**
	Elasticity	No, any, in/out only, up/down only	Only in/out.
	Scaling Automation	No, VNF-triggered, triggered by the NFV management and orchestration functions.	Triggered by T-NOVA Orchestrator.
	VNF Policy Management	Full policy, not policy based. This feature is related to the ability of a VNF to support rule-based provisioning needed for automated tasks (e.g. scale-in, scale-out, or migration based on threshold crossing).	Full policy is supported in T-NOVA metadata.
	Migration operations	No live, Live, Partial, other.	Not addressed yet.
	VNF State	Stateful, stateless.	Supported.
	VNF Internal Structure	Simple, structured.	Supported.
	Reliability	Dependent on fault detection and fault reporting mechanism.	Reliability is up to each VNF. It is provided at application level.
	Location Awareness	Dependencies on position.	Location independent.
	Application Management	No, proprietary, standard, multiple service providers.	Supported.
	Diversity and Evolution of VNF Properties	In a VNF with multiple VNFCs, each VNFC may have different properties.	Supported.
Attributes describing VNF's Requirements	VNF Topological Characteristics	The deployment configuration and operational behaviour of a VNF shall be described according to a template called	T-NOVA metadata.

WORK IN PROGRESS

ETSI VNF feature		*Description*	T-NOVA support
		Virtualized Network Function Description (VNFD).	

Table 3-1 T-NOVA support of ETSI VNF features

Finally, the data model used to describe VNFs in T-NOVA is considered.

In ETSI, the Virtualized Network Function Description (VNFD) is a specification template provided by the VNF Provider for describing virtual resource requirements of a VNF. It is used by the NFV management and orchestration functions to determine how to execute VNF lifecycle operations (e.g. instantiation, etc.). VFND is fully described in [4].

The template captures the general characteristics of each VNF and is used to on-board the VNF, in order to support on-demand instantiations of the VNF in an operator's network.

The deployment configuration and operational behaviour of a VNF shall be described according to the same VFND template. The deployment configuration defines the state and environment for a VNF to be deployed, whereas the operational behaviour defines the needed functions for a VNF to be operated and managed properly [5].

The VNFD is composed of the following main information elements groupings:

• VNF identification data, including:

- Data to uniquely identify the VNF vendor/provider.
- Type and description of the VNF, which help to identify the Network Function that is implemented as a Virtual NF, and enable interoperability of VNFs manufactured by different VNF Providers.
- Version.

• VNF specific data, including:

- Specific VNF configuration data.
- Connectivity requirements and inter-dependencies of VNFCs.
- VNF lifecycle workflow scripts.
- Deployment flavours, specifying how many instances of each VNFC type and version to be instantiated based on service KPIs.
- Deployment constraints.

• VNFC data, including:

- Type and identification, uniquely identifying the VNFC type.
- Specific VNFC configuration data and scripts.
- Deployment constraints.
- Virtual container files/images references, including the possibility to define packages of: VNFC binaries plus operating system, empty operating system, and/or empty virtual container (i.e., unloaded operating system).

• Virtualized resource requirements, including:

WORK IN PROGRESS

- Compute resources, e.g. virtual CPU and virtual RAM assigned to the virtual container.
- Storage resources, e.g. size of the virtual disk assigned to the virtual container.
- Network resources, e.g. number and type of virtual NICs that are assigned to the virtual container, including also requirements for network bandwidth.

Besides all this information, the metadata used in T-NOVA (section 3.2.2) includes some additional information for the marketplace, which is used to build and maintain an ecosystem for an easy and efficient brokering of VNFs among different stakeholders in the market (FPs, SPs and Customers).

3.3. NF Lifecycle

The VNF lifecycle encompasses the different stages that a VNF needs to pass through. In the T-NOVA project, we have defined the following stages:

- Development:
  - Software implementation of Network Functions (NFs) is performed by Function Providers. NFs are published and aggregated in the T-NOVA Function Store.
- Validation:
  - The validation procedure aims at providing a certification that the developed NFs will work as expected.
- Publication:
  - NF publication is performed at the NF Store, whose repository will host both the function image (as stand-alone application or integrated VM) and the associated description/metadata
- Brokerage:
  - Brokerage is undertaken by the brokerage module in the marketplace that will match users' service requirements with the technical capabilities provided by the NFs, thus ensuring that the resources required for NFs deployment are available.
- Selection:
  - Finally the customer selects the most suitable NFs according to his or her needs.
- Deployment:
  - The VM image and its metadata are transferred from the NF Store
- Management:
  - This is the running phase of the NF. An Application Programming Interface (API) will be exposed to the ochestrator for setup and real-time control or management. The running phase can be preceded by a network re-configuration procedure.
- Termination:
  - Involves the removal of the NF instance from the virtualised infrastructure, including network re-configuration, if needed.

WORK IN PROGRESS

The VNF lifecycle is represented in figure 3-9.

Figure 3-9. VNF lifecycle

Actual interaction between T-NOVA Orchestrator and VNF happens in management and termination states of the VNF lifecycle, as highlighted in figure 3-9. This consideration brings to the definition of the extended VNF lifecycle presented in figure 3-11. It highlights the status diagram internally to the management status.

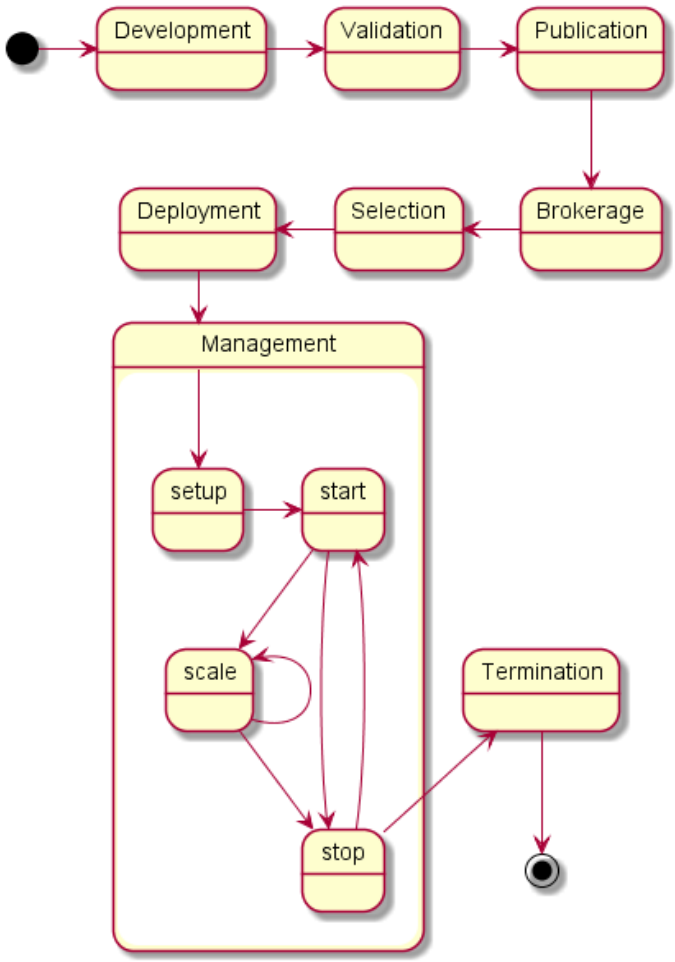


Figure 3-10. Extended VNF lifecycle

The remaining of this chapter is devoted to the definition of the VNF lifecycle.

© T-NOVA Consortium

3.3.1. Development

The development covers the implementation phase of the VNF.

The VNF shall be uploaded on the Function Store using an appropriate T-NOVA API. The NF application will run in the execution environment under the coordination of T-NOVA Orchestrator.

The VNF developer shall also provide the metadata description of the VNF. The description includes both functional and non-functional information that will be used by different elements of T-NOVA framework.

T-NOVA does not intend to use a specific programming language nor to impose a given SDK. The result of the VNF development process is one or more virtual machine images and the related metadata files. The information model is described in a section3.2.2 of this document. As for the implementation details, they will be addressed in the WPs dedicated to VNF implementation.

3.3.2. Validation

The validation process ensures that the VNF:

- supports T-NOVA API,
- operates as expected,
- performs as expected.

In T-NOVA it is required that only validated VNFs can be uploaded to the NF Store.

VNF validation is performed off-line and prior to any upload. This process will in particular generate some information that will be reflected in the VNF metadata.

Implementing automated tools for VNF validation is out-of-scope of T-NOVA. Nevertheless, T-NOVA will provide guidelines and describe validation best practices based on the experience acquired from the effective implementation of VNFs in the T-NOVA framework.

The validation process shall include executing a number of test suites. The minimal requirement is to verify that the VNF supports the T-NOVA API for the whole VNF lifecycle.

3.3.3. Publication

The publication process consists in registering the VNF and its metadata into the NF Store. Only validated VNFs can be uploaded to the NF Store. The interaction with the NF Store occurs through a user interface supported by the T-NOVA FP dashboard.

This stage of the VNF lifecycle needs to be further detailed to address the updating of a VNF or also its withdrawal from the NF Store.

WORK IN PROGRESS

© T-NOVA Consortium

The publication stage in the VNF lifecycle is therefore composed by:

- Publication,
- Modification,
- Withdrawal.

of a VNF and its metadata.

In all the cases, the NF Store informs the T-NOVA Orchestrator about the modification in the NF Store. In case of publication or modification of the VNF, the orchestrator gets the updated VNF and metadata from the NF Store. In case of withdrawal the orchestrator just removes the VNF from its internal database.

The diagram in figure 3-11 describes the publication procedure. The same workflow applies to the modification of the VNF by adapting the function calls accordingly.

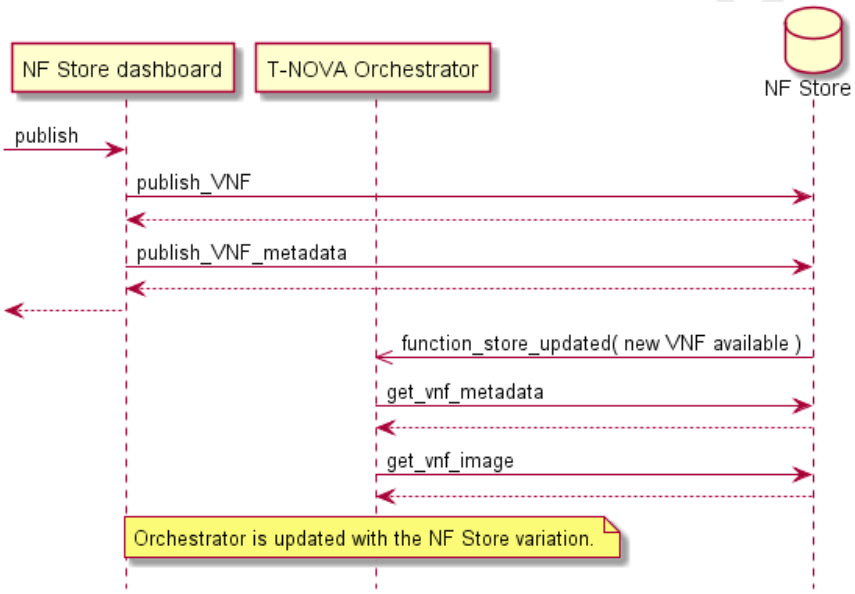


Figure 3-11. VNF publication in the NF Store

The workflow for withdrawing the VNF from the NF Store is slightly different. It is represented in figure 3-12.

© T-NOVA Consortium

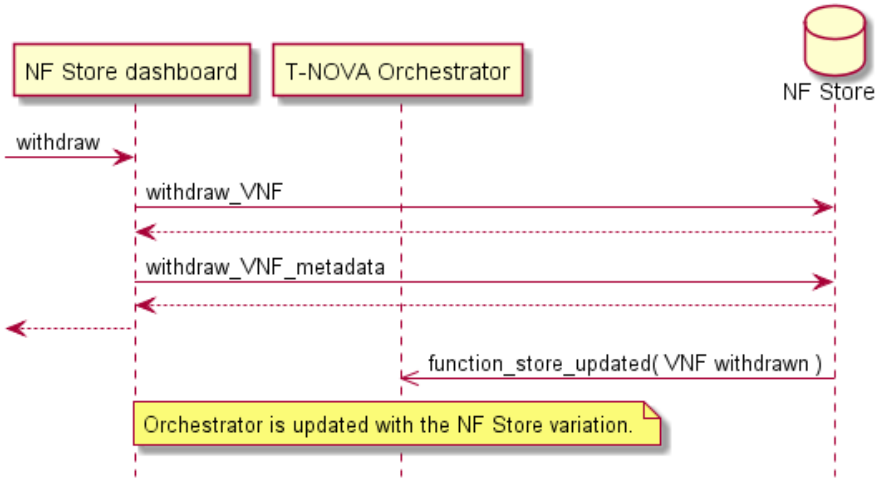


Figure 3-12. VNF withdrawal from the NF Store

3.3.4. Brokerage and Selection

These operations are performed by the marketplace. The metadata of the VNFs is accessed by the brokerage module in the marketplace to perform trading among Function Providers offering a similar VNF. The final goal is to find the best price for the Service Provider, considering the VNFs description and offered SLA (see section 2.6.2.3.).

3.3.5. Deployment

Deploying a VNF means installing and initializing the VNF in the NFV infrastructure. Then the VNF is ready to be activated.

The deployment of a VNF consists of transferring the VNF VM image(s) containing the VNF from the NF Store to the NFV infrastructure.

The deployment phase requires interaction between T-NOVA Orchestrator and the VIM.

The workflow is fully described in Deliverable D2.31 related to Orchestration and Virtual Infrastructure specification [2].

3.3.6. Management

The management stage is dedicated to real-time management of the running VNF.

This is composed by a set of sub-stages such as:

- set-up
- start

WORK IN PROGRESS

- stop
- scaling
- monitoring

The management phase is controlled by the T-NOVA Orchestrator that directly interacts with the VNF and the VIM. There is also an indirect interaction between the VNF and the Virtual Infrastructure because of the execution of the virtual machines composing the VNF. This interaction is common to all the VMs in a cloud environment.

3.3.6.1. Set-up

The VNF set-up consists of the initialization phase of the VNF. The orchestrator asks to the VIM to start the VNF. Then it can directly interact with the VNF, for instance to configure the IP interfaces (NICs). This interaction happens through the T-NOVA API supported by the VNF. The bootstrap initialization results from an indirect interaction with the NFVI. The NIC configuration is a direct interaction with the orchestrator. The latter uses the VNF metadata information for learning about the VNF IP interfaces. Moreover, it needs to know the service description for configuring the service graph or forwarding graph in the right way. If needed, the ochestrator can also run network configuration tasks. In this case the T-NOVA Orchestrator asks the VIM to reconfigure the network.

The set-up phase in figure 3-13 represents a case of a VNF composed by a single VM.

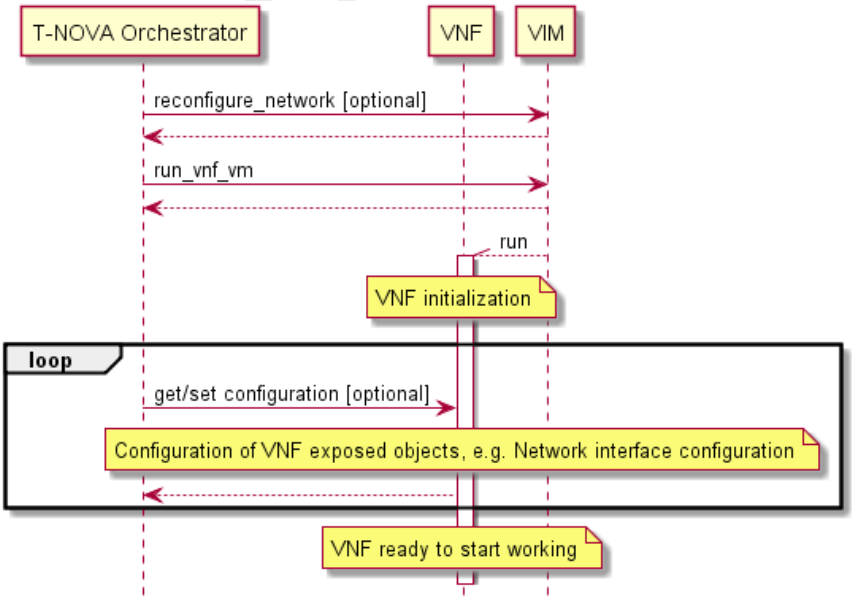


Figure 3-13. VNF set-up

In case the VNF is composed by more VMs or even more VNF Components the orchestrator shall repeat the procedure for all the VMs. In this case the NIC configuration could require more complex interactions with the VNF.

© T-NOVA Consortium

3.3.6.2. Start

The start command instructs the VNF to start providing its services.

The start phase in figure 3-14 represents a case of a VNF composed by a single VM.

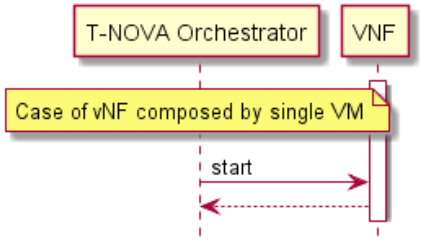


Figure 3-14. VNF start

3.3.6.3. Stop

The stop command instructs the VNF to stop providing its services. After a stop command, the VNF can receive a new start command. Otherwise, the lifecycle evolves to the termination stage.

We can distinguish between graceful and immediate stop. Graceful stop is a request to the VNF to stop accepting new service session request. However, active service sessions continue to be supported until their natural termination. Immediate stop request forces the termination of all active service session.

The stop phase in figure 3-15 represents a case of a VNF composed by a single VM.

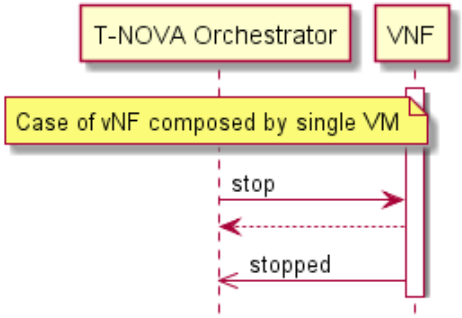


Figure 3-15. VNF stop

3.3.6.4. Scaling

Scaling exploits the elasticity property of a service based on a cloud environment. In other words, it is the capability to dynamically increase or reduce the amount of resources used for providing the service. According to ETSI NFV, scaling is the ability to dynamically extend or reduce resources granted to a VNF [17].

Scaling policies are known as scaling out/in, and scaling up/down. Scaling out/in means allocating or releasing resource instances, i.e. creating new VMs or terminating VMs. Scaling-in adds, while scaling-out removes resource instances. Scaling up/down

WORK IN PROGRESS

means increasing or decreasing the resources allocated to an instance, i.e. increasing or decreasing memory or computation power to a VM. Scaling-up increases, while scaling-down decreases resources.

Scaling out/in does not require additional capabilities to the VNF or to the VIM. It is up to the orchestrator to interact with the VIM for instantiating a new VM and then with the VNF to configure it for collaborating in the service provisioned. In a similar way a VNF can be terminated and then removed.

Implementing scaling up/down needs additional features not widely available in present operating systems and hypervisors. This scaling policy consists in modifying the hardware configuration of a machine while it is running. Both the hypervisor and the operating system of the virtual machine shall support live adaptation of computation, memory, storage, and network resources. Moreover, these operations shall be coordinated between the hypervisor and the operating system. Also for this scaling policy it is up to the orchestrator to initiate and manage the process. In this case it shall require dedicated VIM capability while it seems that the VNF application is not particularly involved in the process.

Independently from the chosen policy, scaling can be the strategy adopted by a Service Provider (SP) to match customer's expectation of an optimally performing system. More technically, scaling might be a follow-up of monitoring activity on SLA fulfillment. Monitoring data are collected from the system monitoring platform (in the orchestrator) and also from the VNF itself. Monitoring data will be compared against SLA threshold provided by the SLA management module in the marketplace. Whenever, some threshold is overcome, the orchestrator shall perform the most appropriate scaling procedure according with the description contained in the VNF metadata. In case the SLA is not fulfilled by scaling procedures, this would be taken into account by the SLA management module to take the appropriate actions tha will impact in billing (section 2.6.2.5.).

The diagram in figure 3-16 gives an example of scaling-out procedure. As soon as the demand increases for the service a new VNF instance is activated and configured for collaborating in efficiently providing the service.

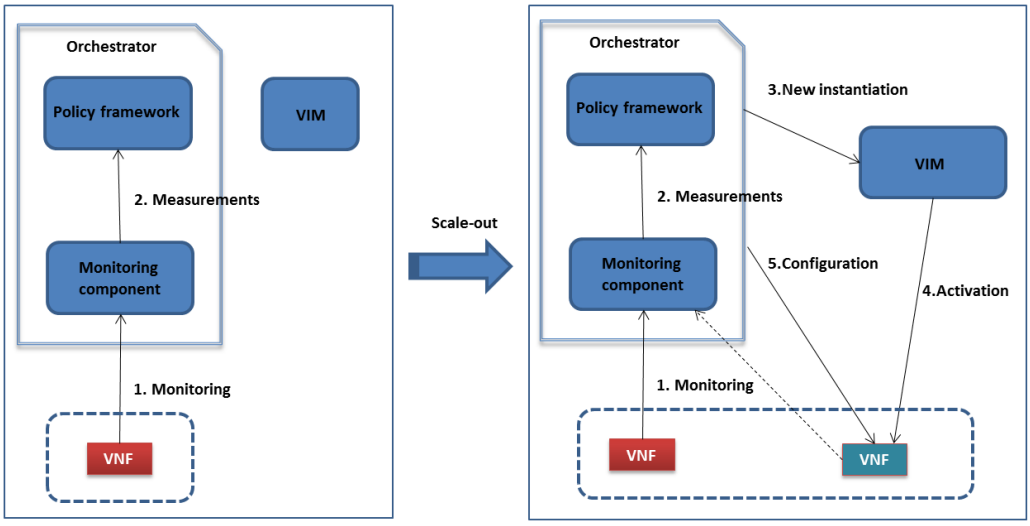


Figure 3-16. Scaling-out example

© T-NOVA Consortium

The pre-condition is that the service has been deployed, configured and running. In order to guarantee service quality, some KPIs (e.g. network delay, packet loss, bandwidth, jitter, CPU threshold, max RTT, etc.) have been set. The corresponding information is measured by the monitoring component and transmitted to the policy framework for evaluation that means to detect the violation of thresholds imposed by the SLA. Should any violation occur, the T-NOVA Orchestrator instructs the VIM to add resources in order to comply with the SLA. As a final result, a new instantiation of network resource(s) will be achieved.

Scaling does not require the VNF to support additional operations besides the set described in this chapter.

3.3.6.5. Monitoring

The provision of an effective monitoring for checking the availability and the health status of the VNFs is mandatory. This will help in better understanding how the VNFs are using the resources, and whether the resources are correctly used. In T-NOVA, the monitoring mechanism will collect and display different types of data including memory, virtual storage and virtual network resource usage. To be more specific, the monitoring mechanism is part of the orchestrator and will be in charge of monitoring on the one hand the infrastructure through the VIM and on the other side the VNFs that are part of the service. Here, the Function Provider (FP) needs to provide what to monitor at the VNF level, and the Service Provider (SP) needs to perform the same task at the composed network service.

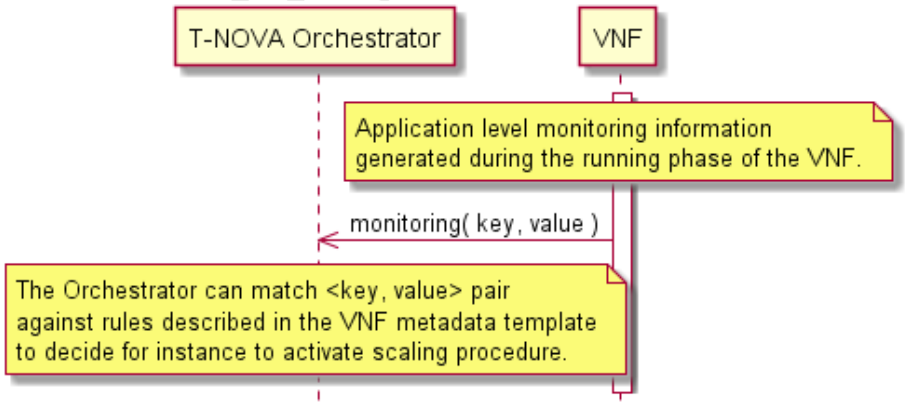


Figure 3-17. VNF monitoring

In Figure 3-17 the KPIs defined by the Function and Service providers as well as their corresponding values that are generated during the running phase of the VNFs are sent to the orchestrator. The latter will match the received information against the "rules" that are defined through the VNF metadata or through the SLAs. Based on this comparison, an action such as activating a scaling procedure might be taken. The KPIs that can be used for the monitoring mechanism include network related data

© T-NOVA Consortium

(network delay, packet loss, bandwidth, Jitter, CPU threshold, etc.) and data related to the service provided by the VNF (in case of SBC could be number a SIP messages in a given time slot, number of concurrent active sessions, etc.).

3.3.7. Termination

This is the final phase of the VNF lifecycle that occurs at the end of the provisioning of the service implemented by the VNF.

Termination phase consists in the removal of VNF instance from the virtualized infrastructure. It can require also network re-configuration.

The termination phase in figure 3-18 represents a case of a VNF composed by a single VM.

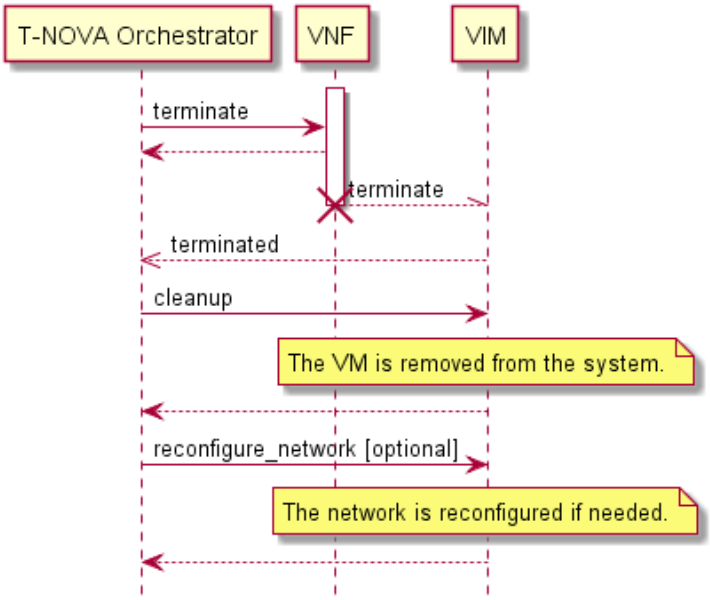


Figure 3-18. VNF termination or clean-up

In case the VNF is composed by more VMs or also more VNFCs the orchestrator shall repeat the procedure for all the VMs. In this case the NIC configuration could require more complex interactions with the VNF.

3.3.8. T-NOVA vs ETSI NFV lifecycle

In this paragraph, a comparison of the lifecycle defined in T-NOVA project with respect to the lifecycle defined in ETSI NFV will be carried out. The focus will be on the lifecycle management functions that are required to manage the instantiation, maintenance and termination of a VNF (or NS).

In ETSI NFV, the state transition diagram for a VNF is depicted in Figure 3-19.

© T-NOVA Consortium

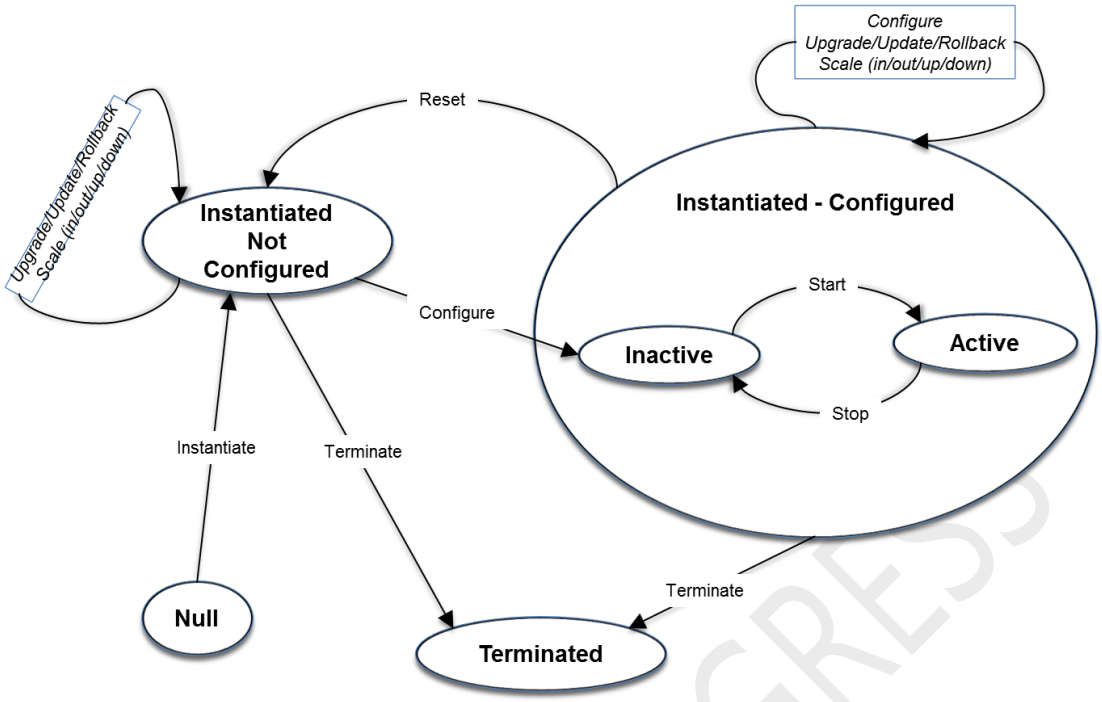


Figure 3-19. VNF instance state transitions

According to the diagram, a VNF can be found in the following states:

- Null: VNF Instance does not exist and is about to be created.
- Instantiated Not Configured: VNF Instance does exist but is not configured for service.
- Instantiated Configured – Inactive: VNF Instance is configured for service.
- Instantiated Configured – Active: VNF Instance that participates in service.
- Terminated: VNF Instance has ceased to exist.

Making a comparison between T-NOVA and ETSI VNF lifecycle is not a trivial task since the ETSI specification documents are still in draft status. The ETSI reference documents used take carry out this analysis are:

- Network Functions Virtualisation (NFV) - Virtual Network Functions Architecture [5] that describes the software architecture of the NFV framework;
- Network Functions Virtualisation Management and Orchestration [4] that describes the functions collectively provided by NFVO, VNFM, and VIM.

The VNF lifecycle is controlled by a set of operations necessary for a VNF to provide its expected functionality. The prerequisite for all lifecycle operations is the VNF on-boarding (process of registering the VNF with the NFVO and uploading the VNF data (VNFD, SW images etc).

In Table 3-2, an overview is given of T-NOVA lifecycle operations and how they map to (or differ from) their corresponding ETSI lifecycle operations.

WORK IN PROGRESS

Lifecycle operation		ETSI Reference
T-NOVA	**ETSI NFV**	
Development In T-NOVA, development is the process to build an NF application compatible with the T-NOVA environment.	- Development is not explicitly described in the ETSI framework.	-
Validation In T-NOVA, validation is an off-line process that has to be made prior to any VNF upload in the NF Store.	Validation In ETSI, a validation step is required during on-boarding by the NFVO.	[4].5.4.1
Publication In T-NOVA, a method is exposed by the T-NOVA Orchestrator for notification change(s) in the NF Store. After notification, the T-NOVA Orchestrator requests the new data.	On-boarding VNF on-boarding refers to the process of submitting a VNF Package to the NFV Orchestrator to be included in the catalogue. In ETSI, the sender will submit directly the VNF Package data to the Orchestrator.	VNF Package management [4].7.2.1 [4].B2.1
Modification	Update	
Withdrawal	Delete	
- - - Managed by T-NOVA Orchestrator.	Disable Enable Query	VNF Package management [4].7.2.1
Brokerage Operation performed in the Marketplace.	- Not in ETSI scope.	-
Selection Operation performed in the Marketplace.	- Not in ETSI scope.	-
Deployment	Instantiate Instantiate will include configuration if specified by the VNF deployment template. In ETSI, a feasibility check	[4].5.4.2 VNF lifecycle

WORK IN PROGRESS

Lifecycle operation		ETSI Reference
T-NOVA	**ETSI NFV**	
	is (optionally) performed before instantiation.	management [4].7.2.4 [4].B.3.1.2
Management::Set-up	Configure	VNF configuration [4].7.2.6
In T-NOVA, Get, Create and Set operations will be supported.	Get In ETSI, the Configure interface will include the following operations: Get, Create, Set, Delete and Notify.	
Management::Start	Start	[5].6
Management::Stop	Stop	[5].6
Management::Scaling In T-NOVA, only in/out scaling will be supported. Sources of scaling will be either the VNF or the monitoring platform itself and the operation will be performed by the T-NOVA Orchestrator.	Scale In ETSI, scaling options are: in/out, up/down. Several sources of scaling have been considered (MANO.B4) that can be reduced to two categories: (a) automated scaling, where decision is done by or forwarded to the VNF Manager and (b) scaling based on management request, where the scaling request is coming from some sender (OSS/BSS or operator) and received by the NFV Orchestrator.	VNF lifecycle management [4].7.2.4 [5].6
Not supported in T-Nova.	In ETSI, the VNF lifecycle management interface will include the following oprations: Query Check Heal Update Modify Upgrade Reset	VNF lifecycle management [4].7.2.4 [5].6
Management::Monitoring In T-NOVA, a VNF monitoring method is exposed by the T-NOVA	In ETSI, monitoring is related to VNF performance and VNF fault management. VNF information exchange is based on a notify/get	VNF performance management [4].7.2.7

WORK IN PROGRESS

Lifecycle operation		ETSI Reference
T-NOVA	**ETSI NFV**	
Orchestrator.	mechanism.	VNF fault management [4].7.2.8
Termination	Terminate	VNF lifecycle management [4].7.2.4 [4].B.6

Table 3-2 Comparison of T-NOVA with ETSI NFV lifecycle operations

In conclusion, the lifecycle management of VNFs and related packages adopted in T-NOVA is mostly aligned with the one adopted in ETSI. However, some differences exist in publication and brokerage, which are T-NOVA specific. In fact, T-NOVA target is an open market environment, while ETSI focus is limited to traditional telco operations.

Due to the fact that neither the specifications in ETSI nor the works in T-NOVA project are consolidated yet, some more investigation on this subject could be required in the future.

3.4. Network Function Framework APIs

The VNF framework described in this section interacts with the T-NOVA system through the interfaces represented in Figure 3-1. The interfaces are exposed by the NF Store and also directly by the VNFs.

The NF Store re exposes interfaces for interacting with the database of the VNF made available to the system.

Interface T-Da-Nfs shown in Figure 3-1. VNF framework high level architecture is used by the dashboard for interacting with the NF Store for publishing, modifying, and withdrawing a VNF and its related metadata.

Interface T-Or-Nfs connects the NF Store with the orchestrator for providing information about the VNF metadata description and its executable image making the VNF available to the T-NOVA system.

Interface T-Ve-Vnfm is exposed by each VNF to the orchestrator. During the active phase of a VNF, the interface T-Ve-Vnfm allows the orchestrator to directly interact with the VNF. This interaction takes place in the "Management" and "Termination" states of the VNF lifecycle. In particular, the whole active phase of a VNF (set-up, start/stop, scaling, monitoring) is controlled by means of direct communication

WORK IN PROGRESS

between orchestrator and VNF. Such communication is performed through calls to the exposed functions of the VNFs API.

3.4.1. APIs high level logical description

For each interface exposed by the Network Function Framework we provide the description of the high level logical primitives or API functions. In-depth analysis and detailed specification of the API will be developed in T-NOVA WP5.

3.4.2. Network Function Store APIs

These APIs allows interacting with the NF Store for managing the uploading, downloading, and deletion of the VNF image and the related metadata descriptor.

Interface T-Da-Nfs contains the following operations:

- publish_VNF, publish_VNF_metadata
  - Upload in the NF Store the VNF image and the VNF metadata descriptor.
- withdraw_VNF, withdraw_VNF_metadata
  - Delete from the NF Store the VNF image and the VNF metadata descriptor.

Interface T-Or-Nfs contains the following operations:

- get_VNF_metadata, get_VNF_image
  - Read from the NF Store the VNF image and the VNF metadata descriptor.

3.4.3. VNF API

The API described in this chapter allows the VNF to be managed by the orchestrator in T-NOVA system. Any VNF must implement this API to be part of T-NOVA.

Interface T-Ve_Vnfm contains the following operations:

- get_configuration, set_configuration
  - These operations access to the configuration parameters supported by the VNF. They allow configuring the service provided by the VNF. Typical usage is assigning the IP address of the VNF's network interfaces.
- start
  - Asks to the VNF to start providing its services.
- stop
  - Asks to the VNF to stop providing its services. Two types of stop are supported: hard and soft. A soft stop ensures a graceful cease of the VNF. In particular, asks to complete any in-progress operation before

WORK IN PROGRESS

stopping the VNF. The hard stop asks to immediately cease all the operations of the VNF. The VNF remains active after the stop operation is completed, i.e. the MV is up and running; only the service provisioning is interrupted.

- terminate
  - Asks to the VNF to terminate its execution. This operation shuts-down the VMs composing the VNF.

3.5. Network Function Store design

The NF Store is mainly a repository for the VNFs and their metadata. High level breakup of NF Store includes:

- NF repository
  - Archive of VNF images and related metadata. This data is versioned so that different versions of the same VNF can be present in the NF Store.
- NFS manager
  - Provides interfaces for interacting with the NF repository.

The NF repository is responsible for concurrent operations that are performed on the store (CRUD operations) on images and metadata.

The NFS Manager block implements the interfaces to the repositories. In addition, it delegates some traits of its responsability (like authentication and authorization access) to an external AA module. Security mechanisms can be adopted for data exchange over these interfaces.

The NF Store provides interfaces to the orchestrator (T-Or-Nfs) and marketplace's Dashboard (T-Da-Nfs) [Figure 1-1]. An additional interface (not shown in the picture) for console management is foreseen.

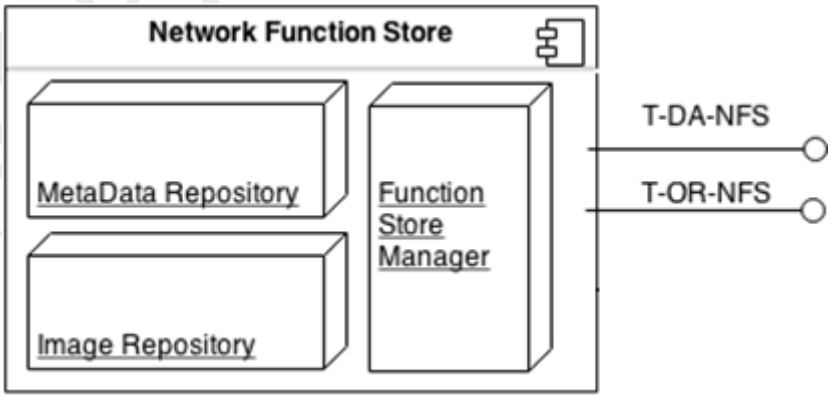


Figure 3-20. NF Store architecture

The requirements capture process for the Network Function Store has been performed according to the procedure explained in section 2.5 for the marketplace components. Theyare listed in Annex 6.1.2.

WORK IN PROGRESS

4. Interaction between Network Function Framework and the Marketplace

In this section we aim to summarize and to make clearer the relation between the Network Function Framework and the T-NOVA Marketplace according what have been described in the previous sections.

Firstly, the way of purchasing VNFs it is summarized in section 4.1, and the interfaces between the NF Store and the marketplace are explained in section 4.2.

4.1. Purchase of VNFs

The purchase of a VNF through the T-NOVA marketplace should go through the following steps:

VNF storage

The T-NOVA Marketplace will advertise only validated Network Functions (NFs). After being tested and validated, the NFs will be uploaded into the NF Store. To prevent misuse of the NF Store, the Function Provider (FP) needs to authenticate itself and be authorized to upload the VNF. The same authentication procedure enables the FP to upload and upgraded version of its VNF. In addition to the VNF, the corresponding metadata information has to be made available.

VNF advertisement

The marketplace will advertise the VNFs stored in the NF Store based on the metadata that accompanies them and their availability provided by the orchestrator. In case a new version of the NF under consideration is uploaded, the NF Store will inform the orchestrator about this and consequently, the VNF advertisement on the marketplace will be updated.

VNF purchase

A customer willing to purchase a service from T-NOVA will need to undertake the following:

Look for available services. This will be achieved by selecting available VNFs or Network Services (NSs), and setting the start and end day and time of the service. The customer might also need to insert the information related to the ingress and egress routers that will form the end points of the path that will carry on the service traffic. The customer will also need to specify the SLA requirements he is interested in. This could include.

Once the customer selects a service, it will need afterwards to configure the related parameters. This will vary from one VNF to another. The parameters might include (content type, protocols to be used, traffic to be allowed or denied, prioritization rules if needed).

WORK IN PROGRESS

© T-NOVA Consortium

After the configuration, the service is ready to run, however, the customer needs first of all to select the means by which it will be charged.

4.2. Marketplace – Function Store interfaces

The interface between the dashboard and the NF Store despicted in Figure 2-4 has the single functionality of providing the API for the Function Providers to insert their VNFs offerings and associated metadata in the Function Store.

Once the VNFs metadata are stored in the FS, the available VNFs to compose a service will be ready for the brokerage module by means of the orchestrator, which will store this information in the VNF catalogue in the orchestrator. This architectural design decision has been taken in order to avoid that the marketplace could allow the SP to create a service starting from VNFs that although there are available at market level, they could be still not technically ready to be part of a service yet.

WORK IN PROGRESS

© T-NOVA Consortium

5. CONCLUSIONS

5.1. Summary

This document provides the specification and high level design of the Network Function Framework and T-NOVA Marketplace for later development. Analysing the state of the art, including main standardization activities, previous research projects and commercial solutions it has been gathered that at this stage a proper marketplace together with the Function Store as T-NOVA proposes to deliver VNFs as a Service does not exist.

The T-NOVA Marketplace has been designed to be used by three kinds of stakeholders according to the use cases analysis performed [1], therefore a three views dashboard will be implemented as well as and access control module that will provide AA functionalities to control their different permissions. The T-NOVA Marketplace will allow Virtual Network Functions (VNFs) provided by a variety of software developers (Function Providers) to be published and traded by means of a brokerage module that will implement pricing mechanisms e.g. auctioning, when a new Network Service (NS) is going to be composed by a Service Provider (SP). T-NOVA Customers will be able to browse and select among the available network service offerings in the marketplace by means a business service catalogue as well as negotiate the associated SLAs&price. The billing procedure contemplates not only final customers of T-NOVA network services, but also the commercial relationship between the Service Provider and Function Providers.

In relation to the specification of the Network Function Framework two main tasks have been performed: the description and the specification of the VNFs and the design of the NF Store. The first one includes the APIs allowing the VNF to be managed by the T-NOVA system as well as the information elements that shall be present in the VNF metadata; a key piece of data of the Network Function Framework is the metadata descriptor associated with the actual software implementation of the VNF. Besides the structural definition of a VNF, its behaviour has been studied defining a lifecycle that is common to all the VNFs in T-NOVA. This lifecycle can be split in an off-line and on-line part. In relation to the on-line one, the lifecycle states where the VNF is up and running over a virtualised execution platform. On the other hand, the off-line lifecycle states span from the software implementation of the VNF to its uploading into the NF Store that can be thought as the place where the VNFs are stored. The NF Store APIs provide interfaces by means of the dashboard with Function Providers for uploading, updating, and withdrawing the VNF software images and metadata description, and interface with the rest of the T-NOVA system for making this information available for service orchestration.

WORK IN PROGRESS

© T-NOVA Consortium

5.2. Contributions to standards

T-NOVA Marketplace will be built considering the current NFV ETSI architecture and considering the on-going TMForum Best Practices for business services delivery under its frameworx (business process, information, application, and integration frameworks) but not restricted to it. For instance it has been decided to include a business catalogue to provide the marketplace offerings being aligned with TMForum proposes in its integration framework.

From the state of the art analysis performed, we gathered that the T-NOVA Marketplace concept is completely novel in regards to ETSI view [4]. T-NOVA introduces the Marketplace aiming at opening the NFV market to third party developers for the provision of VNFs, a concept that currently falls outside the technical view of ETSI NFV.

On the other hand, supported by the TR228 TM Forum Gap Analysis related to MANO Work [8] it has been gathered that ETSI does not provide so far any more insight on the OSS/BSS (Operating Support System / Business Support System) of the operator apart from the definition of an interface; a detailed implementation model on how to manage operational and business support systems in a hybrid legacy and virtualized environment is something that ETSI is not addressing so far.

Though being out of the T-NOVA scope the interface between the MANO architecture and the existing OSS/BSS system of operators, T-NOVA aims to 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 external operator, and what could be a first input for latest studies in the interoperability with OSS/BSS existing systems, which also TM Forum ZOOM intends to address in the future [6]. Other future work of TMForum that T-NOVA Marketplace is aligned to is the impact of the SLA Management in virtualization.

In relation to the NFV framework, we conclude that the lifecycle management of VNFs and related packages adopted in T-NOVA is mostly aligned with the one adopted in ETSI. However, some differences exist in publication and brokerage, which are T-NOVA specific. In fact, T-NOVA target is an open market environment, while ETSI focus is limited to traditional telco operations.

Due to the fact that neither the specifications in ETSI nor the works in T-NOVA project are consolidated yet, some more investigation on this subject could be required in the future; therefore the second and final version of this document will investigate in this direction, which could bring contribution to ETSI activities.

5.3. Future Work

The specification provided in this document has been based on the requirements at T-NOVA system level described in previous deliverables together with some relevant

WORK IN PROGRESS

parts of the ETSI NFV work and TMForum best practices. The information assembled via this process has been the critical input into a two stage process: Stage 1 consisted on a research and design phase, where a system engineering approach was adopted to define the key functional components (Deliverable 2.21) [3]. Stage 2 presented in this document has defined both the reference architecture and its functional entities and interfaces in a technology-agnostic manner to decouple the specifics of the implementations details. An additional third stage which constitutes the activities within WP5/6 will address the details of the suitable technologies and their operation.

It is expected that the abovementioned technical WPs will give feedback in terms of requests of clarification and further developing of certain contents that are essential for effective implementation. This feedback will be use to elaborate the second and final version of the present document that has been scheduled to be released on September 2015.

WORK IN PROGRESS

© T-NOVA Consortium

6. ANNEXES

6.1. Annex A - Requirements Specification

To specify requirements, the following template has been used, with the following fields:

Field	Meaning
Req. id	Requirement ID, of the form YYY_xx, in which YYY identifies the component and xx is numbered sequentially, starting from 01 por each different component. • Dashboard – D.x • Access Control – AA.x • SLA management module– SLA.x • Brokerage Module – B.x • Accounting module – A.x • Business Service catalogue store – BSC.x • Billing module – Bil.x • Interfaces marketplace - orchestrator – IMO.x • Network Function Store – FS.x
Use Case	Use case(s) in Deliverable D2.1 [1] from which the requirement is originated.
Domain	Technical domain to which the requirement belongs, selected out of the list: • Management and orchestration • Security • Service continuity • Operations • Market / Commercial operability
Requirement Name	Short requirement name
Requirement Description	Full requirement description. It usually corresponds to a sentence including the word "shall" (for mandatory requirements), ou "should" (for optional requirements).
Justification of Requirement	Rationale behind the requirement

Every requirement has an implicit severity level, which is indicated by the verb used to express it, in accordance to IETF RFC 2119 [1]:

WORK IN PROGRESS

- SHALL corresponds to an absolute requirement, something that must be supported by the implementation.
- SHOULD corresponds to a recommended, but optional, requirement – paraphrasing RFC 2119, this means that "there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course".

Generally speaking, the criterion for assessing the severity level of each T-NOVA requirement was basically whether or not that specific requirement is indispensable for the system to deliver its basic function.

WORK IN PROGRESS

© T-NOVA Consortium

6.1.1. Detailed marketplace components requirements specification

Dashboard requirements for the three views:

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.1	UC1	Security - AA	Authentication and access control	The dashboard SHALL provide a "login in" page for the different stakeholders to be authenticated	Stakeholders interacting with the T-NOVA system should be authenticated and authorised in order to be able to browse the Business Service Catalogue, issue SLA requests, or upload NFVs
D.2	UC1.1	Security - AA	Authentication and access control	The "login" page in the SHALL offer to the different stakeholders means to use (username, password, OpenID, Google API) for authentication	Stakeholders interacting with the T-NOVA system should be able to use different authentication techniques to access T-NOVA
D.3	UC1.2	Security - AA	Authentication and access control	The "login" page in the SHALL offer to the different stakeholders means to remember credentials when loging on	Stakeholders interacting with the T-NOVA system should not be obliged to insert credentials when accessing again the system
D.4	UC1.1	Management & Orchestration	Web access	The Dashboard SHALL be accessible to authorized users via the Internet	The Dashboard will provide the necessary interface in order to be viewed over the Internet
D.5	UC1.1	Management & Orchestration	Parallel Access	The Dashboard SHOULD provide multiple users login and no less than 10	The Parallel access will provide the necessary tools for every user to be able to provide his content
D.6	UC1.1	Management & Orchestration	Availability	The Dashboard SHOULD be available 24/7 365 days per year	The Dashboard must be always on in order to control every part of the T-NOVA infrastructure.

WORK IN PROGRESS

Dashboard requirements for the Customer view:

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.7	UC 1.1	Service continuity + Market/commercial operability - brokerage	Service offerings selection	The dashboard SHALL be able to allow a customer to watch offerings, selecting one or several of them	Service, SLA and price information need to be visualized by the customer to allows him to perform a selection
D.8	UC 1.1	Service continuity + Market/commercial operability – SLA management	SLA selection	The dashboard SHALL be able to allow a customer to select /negotiate among different SLA levels for a specific service	Service, SLA and price information need to be visualized by the customer to allows him to perform a selection
D.9	UC 4.1	Service continuity-SLA management	SLA visualization by customer	The dashboard SHALL be able to allow a customer to visualized SLA fulfilment information when he asks for it	Customer has to be able to visualize SLA fulfilment information when he asks for it
D.11	UC5	Market / commercial operability-billing	Bills visualization by customer	The dashboard SHALL be able to allow a customer to visualize his billing information when he asks for it	Customer has to be able to visualize his bills when he asks for it
D.12	UC1	Service Continuity-orchestrator	Service termination by the customer	The dashboard SHALL allow the customer to request service termination	The duration of the NS will be specified in the SLA, when the NS is no longer needed the system should de-compose the NS and cancel the SLA. Alternatively the SLA can be terminated by the customer on-demand.
D.13	UC1, UC1.3	Service continuity-Brokerage	Service Provision	The dashboard SHALL be able to make available the most suitable offerings that the T-NOVA marketplace, can provide to the customer.	The customer must be able to select from a list of Services.
D.14	UC2	Management & Orchestration–orchestrator (service configuration)	Customer service portal	The dashboard SHALL allow the customer to provide the means to configure the VNFs of a specific NS.	Any parameters required to configure the VNF (e.g. IP prefixes, traffic classes, etc.) must be accessible by the customer through the service portal.
D.15	UC2, UC3	Management & Orchestration–orchestrator (service monitoring)	Customer notification - VNF starts / fails to start	If the VNF starts correctly, the T-NOVA system SHALL be able to notify the customer about this event. The customer service portal shall provide this information to the customer. If the service fails to start correctly, the T-NOVA SHALL be able to notify the customer about this event.	The customer must get feedback about success or failure of his/her service request

WORK IN PROGRESS

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.16	UC3	Management & Orchestration, Elasticity	Customer service portal - Scale In/Out	The dahsboard SHALL provide a means for a customer to request either a "scale out" or "scale in" of a deployed VNF Service. When the customer requests a VNF "scale out" or "scale in" the customer will have the option to reuse a previous configuration or to specify a new configuration.	The T-NOVA system must provide the ability for customers to request additional VNF services or to request the removal of VNF services.
D.17	UC3.2	Management & Orchestration, Elasticity	Customer/SP Scale Down VNF service	The Dashobard SHALL be able to allow the customer to scale down an existing VNF service	Scale down is necessary to ensure the T-NOVA system can meet the changing needs of the customer
D.18	UC6_1 - UC6_4	Management & Orchestration-Brokerage	Service catalogue	The dashboard SHALL be able to provide all services that are active for the authorised customers.	Service catalogue is essential, because it provides to the authorized customers the possibility to identify the service that needs to be terminated
D.19		Dashboard, Service Configuration	Collect NS parameters	The dashboard SHALL collect the service's pre-defined parameters when a customer selects a specific NS	See **[1]**. Note that the **Customer** can only choose from a list of authorized NS.
D.20		Dashboard, Service Configuration	Display NS parameters	For the NS chosen by the customer, the Dashboard SHALL display the parameters for a selected NS by the customer, taking into account their properties (default values, enumerated values, read-only, inter-relationship, etc.)	Display should take into account related/inter-dependent parameters (see [1])
D.21		Dashboard, Service Configuration	Validate NS parameters	If a validation function is provided by the Service Provider (at the service composition moment) for any parameter or set of parameters, apply it and give feedback to the Customer	
D.23		Dashboard, Service Monitorisation	Collect and display metrics	The dashboard SHALL display the service's pre-defined metrics	See [3].
D.24		Dashboard, Service Monitorisation	Manage configuration of service paramenters (Customer)	The dashboard SHALL allow the customer to configure some service parameters	Includes not only the graphical configuration (e.g., pie-graph vs. bar-graph, graph title, etc.)

WORK IN PROGRESS

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.25		Dashboard, Service Monitorisation	Display configured metrics	The Dashboard SHALL display the configured metrics	

Dashboard requirements for the Function Provider view:

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.26	UC1.1	Service continuity + Market/commercial operability – Function Store	VNFs and SLAs description by the FPs	The dashboard SHALL be able to allow Function Providers to describe their VNFs and metadata (conditions, SLA, price).	VNFs and SLAs description need to be stored in the system to allow the customer to browse through this information
D.27	UC1.2	Security-AA	FP authentication, certification	The dashboard SHALL allow the FPs to be authorised by the system in order to advertise, upload and modify any VNF. The acceptance of a FP will be subject of bilateral discussions between the developer and the T-NOVA operator.	This is essential in order to control the access to the Brokerage and the Function Store and increase security.
D.28	UC1.2	Operational-FS	VNF Advertisement	The dashboard SHALL allow FPs to be able to advertise the VNF capabilities in the system.	The FP for each VNF that is uploaded or modified needs to notify the Orchestrator in order that the Service catalogue is updated. This action is called advertisement of the VNF. The advertisement request should contain the VNF metadata, including name, id, pricing information, requirements and VNF capabilities.

WORK IN PROGRESS

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.29	UC1.2	Operational-FS	VNF withdrawal	The dashboard SHALL allow FPs to be able to removed one of tis VNFs from the system.	The FP for each VNF that is withdrawn needs to notify the Orchestrator in order that the Service catalogue is updated.
D.30	UC1.2	Operational-FS	VNF Upload	The dashboard SHALL allow FPs to be able to upload the packaged VNF to the Functions Store.	The system should offer a method to the FP for uploading and storing the packaged VNF to the Function Store. Both VNF image and its metadata are uploaded. When a particular VNF is requested the Orchestrator will instantiate this VNF to the appropriate NFVI-PoP.
D.31	UC1.2	Security/Operational-FS	VNF validation	VNF validation SHOULD be checked.	The VNF shall be validated before being included into the Function Store. Validation is an off-line process. It is out-of scope of T-NOVA.
D.32	UC1.2	Operational- FS	FP VNF status monitoring	All the VNFs of the same developer SHALL be browsable in the developer dashboard, from where the developer is able to monitor the status and other statistical data (popularity, rating, comments etc.).	This requirement covers the need for supporting the monitoring of each VNF by the FP in terms of availability, popularity, malfunctions and alerting.

WORK IN PROGRESS

Dashboard requirements for the Service Provider view:

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.33	UC1.1	Service continuity + Market/commercial operability – service catalogue	Services and SLAs description by the SP	The dashboard SHALL be able to allow Service Providers to describe their services and conditions description (service (+ service description), SLA, price)	Services and SLAs description need to be stored in the system to later allow the customer to browse through this information
D.34	UC 4.1	Service continuity-SLA management	SLA visualization by the SP	The dashboard SHALL be able to allow a customer to visualize SLA fulfilment information on demand	Customer has to be able to visualized SLA fulfilment information when he asks for it
D.35	UC5, UC1	Market / commercial operability-billing	Bill cycle agreement	The dashboard SHALL allow the SP to be assigned a bill/billing mode cycle with the FPs	Billing procedure needs to know when a bill cycle finishes (for non pay- as-you-go services).
D.36	UC5	Market / commercial operability-billing	Bills visualization by SP	The dashboard SHALL be able to allow a SP to visualize his billing information on demand	Customer has to be able to visualized his bills when he asks for it
D.37	UC1, UC2, UC3	Operational, Service Continuity, Management & Orchestration-orchestrator	NS Composition by the SP	The dashboard SHALL allow the SP to compose a NS from atomic VNF instances available at the NF Store and define the logical topology among the several components (see [2])	The creation of a NS from the combination of atomic/simple VNF is important in order to simplify the process provision of NS to the customers and avoid complex path calculations
D.38	UC2, UC3, UC4	Management & Orchestration-Orchestrator	Resource monitoring	The dashboard SHALL allow the SP to monitor and collect information about consumption and availability of resources (computational, storage, network) on a real time basis, including the resources consumed by each specific VNF instance.	Monitoring is essential to ensure that the deployment of VNF's onto hosting infrastructure is performed adequately. Monitoring is also r provides essential metrics required by operations such as rescaling, billing, etc.
D.39	UC3	Management & Orchestration, Elasticity-orchestrator	SP service portal - Scale Up/Down	The dashboard SHALL provide the means for a SP to request either to up or down scale the resources allocated to a deployed VNF Service based on the SLA evaluation.	The T-NOVA system must provide the ability for customers to request additional resources or the removal of resources from a deployed VNF service.

WORK IN PROGRESS

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
D.40	UC3	Management & Orchestration, Elasticity-orchestrator	SP notification - VNF is removed	A notification SHALL be shown to the SP if the VNF and its host VMs are successfully removed from the T-NOVA system.	The customer must get feedback about the success or failure of their service request
D.41	UC3	Management & Orchestration; Elasticity-orchestrator	SP notification - VNF rescale	The dashboard SHALL notify the SP when their request to rescale a VNF has been successfully completed.	The customer must get feedback about the success or failure of their service request
D.42	UC3.2	Management & Orchestration, Elasticity-orchestrator	SP Scale Out VNF	The dashboard SHALL allow the SP to choose to scale in/out an existing VNF.	Customers will request increases in VNF services to meet business needs
D.43		Dashboard, Service Composition	Browse available VNFs	The Dashboard SHALL allow the Service Provider to browse the available (and authorized) VNFs	The list of available VNFs is a combination of the list provided by the Orchestrator and the list of authorized services for the Service Provider (in a multi-Service Provider scenario).
D.44		Dashboard, Service Composition	Select and configure one or more VNFs	The dashboard SHALL allow the SP to compose a Network Service from a set of available VNFs [2].	Note that more than one instance of the same VNF can be used in the composition of a NS. This configuration includes the definition of the selected VNFs' connection graph. This connection graph is, in the simplest case, the connection of the output of a VNF instance to the input of another VNF instance. In more complex cases there might be some form of processing the output (e.g., max/min/average, step, etc.) before delivering to the input.
D.45		Dashboard, Service Composition	Define composed service parameters	For a given composed service, the Dahsboard SHALL allow the SP to define its parameters and map those to the parameters of the VNFs used in the composition.	Includes the definition of related/inter-dependent parameters (for displaying purposes, see [2]) and mapping VNF's parameters to the composed service

WORK IN PROGRESS

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
					parameters, as well as the indication of default values, mandatory or read-only properties, etc.
D.46		Dashboard, Service Composition	Define composed service indicators	For a given composed service, the dashboard SHALL allow the SP to define its metrics, mapping from those available from each of the VNF used in the composition	Includes the definition of related/inter-dependent indicators (for displaying purposes, see [2]) and mapping VNF's metrics to the composed service metrics.
D.47		Dashboard, Service Composition	Manage life-cycle of composed services	The Dashboard SHALL allow the SP to manage the life-cycle of the composed services.	Note that if a composed service is being used, it can not be removed entirely bit can be dropped from the list of available services in the Marketplace
D.48		Dashboard, Service Monitorisation	Collect and display metrics	The Dashboard SHALL display the service's pre-defined metrics	See [3].
D.49		Dashboard, Service Monitorisation	Manage configuration of metrics (SP)	The Dashboard SHALL allow the configuration of service paramenters by the SP.	Includes not only the visibility of the metric and any other graphical configuration (e.g., pie-graph vs. bar-graph, graph title, etc.) but also eventually some form of combining two or more metrics. A subset of these metrics will likely be related to the agreed SLA.
D.50		Dashboard, Service Monitorisation	Display configured metrics	The Dashboard SHALL display the configured metrics	

WORK IN PROGRESS

Access Control

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
AA.1	UC1	Security	Authentication and access control	AA module SHALL support mechanisms for authentication and authorisation for the different stakeholders. The authorization is based on the associated roles and permissions	Stakeholders interacting with the T-NOVA system should be authenticated and authorised in order to perform tasks such as uploading a NF or purchase a VNF
AA.2	UC1	Security	Secure communication	AA framework SHOULD provide encryption.	Encryption should be used, in order to prevent eavesdropping.
AA.3	UC1.2	Security	FP authentication, authorization	AA SHALL authenticate and authorise the FPs in order to advertise, upload and modify VNFs	Each FP that interacts with the Brokerage and the Function Store needs to be authenticated and authorised by the system. This is essential in order to control the access to the Broker and the Function Store and increase security.
AA.4		Security	Stakeholders Authorization	Profiles are created according to T-NOVA roles and stakeholders are assigned roles based on their responsibilities and qualifications	Based on the different stakeholders responsibilities and qualifications and the tasks that they will undertake, roles will be created
AA.5		Security	Stakeholders Authorization	Roles can be reassigned or granted new permissions if needed. Which means they SHALL be updatable and revocable	As responsibilities might change or extended/reduced over time, roles SHALL be flexible enough to be reassigned, extended with new features, or revoked

Brokerage:

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
B.1	UC1.3	Management & Orchestration	Service Catalogue	The Brokerage SHALL be able to communicate with Service catalogue	The Brokerage must be able to seek the available services in the Service catalogue before creates a new service.

WORK IN PROGRESS

B.2	UC1.3	Operational	Trading/Bidding	The Brokerage SHALL be able to perform auctions among Service provider and Function providers	The Brokerage must be able to initiate auctions whenever it is required.
B.3	UC1.3	Management & Orchestration	Service Provision	The Brokerage SHALL be able to provide new service offerings to the dashboard	The Brokerage must be able to provide new services to in the Service catalogue in order the Customer to be able to browse/select them in the dashboard.

Business service catalogue

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
SC.1	UC1.1	Service continuity + Market/commercial operability	Services and SLAs description	The service catalogue SHALL be able to store all the available NSs in the T-NOVA marketplace, specifying SLA level and price.	Services and SLAs description need to be stored in the system to allow the customer to browse through this information
SC.2	UC1.1	Service continuity + Market/commercial operability	Services and SLAs description	The service catalogue SHALL be browsable by the SP	

SLA management module

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
SLA.1	UC1.1	Service continuity	SLA information customer-SP storage	The SLA management module SHALL store all the SLA agreements between a customer and the SP for each service.	SLA agreements must be stored in order for service monitoring to determine if the SLA has been fulfilled or not

WORK IN PROGRESS

SLA.2	UC1.1	Service continuity	SLA information SP-FPs storage	The SLA management module SHALL store all the SLA agreements between the SP and the FPs for each VNF.	SLA agreements must be stored in order for a VNF monitoring to determine if the SLA has been fulfilled or not
SLA.3	UC3, UC4, UC4.1	Management & Orchestration, Operations, Service Continuity	SLA – orchestrator interface	The SLA management module SHALL be connected to the orchestrator to let it know about the agreed SLA for each service. (When the SLA is not fulfilled the orchestrator will have to initiate the applicable action, e.g. rescaling)	SLA management and monitoring is considered essential for the commercial applicability of the T-NOVA system. The T-NOVA system must determine when an SLA is in breach and trigger corrective actions.
SLA.4	UC 4.1, UC5	Market / commercial operability	SLA fulfilment information storage (from the orchestrator)	The SLA management module SHALL store all the information about SLA fulfilment for eventual compensations or penalties for later billing.	In order to compensate the customer economically for not achieving the SLA agreed, the billing system must have this information
SLA.5	UC 4.1, UC5	Market / commercial operability	SLA – accounting/billing interface	The SLA management module SHALL be connected the accounting/billing system to let it know about eventual compensations or penalties when the SLA has not been fulfilled for a specific service.	In order to compensate the customer economically for not achieving the SLA agreed, the billing system must have this information
SLA.6	UC 4.1	Service continuity	SLA visualisation by customer and SP	The SLA management module SHALL be connected to the Dasboard to allow a customer and SP to visualize SLA fulfilment information when requested.	Customers and SPs has to be able to visualize SLA fulfilment information since this is the real service level they are getting and paying for.
SLA.7	UC1.1	Service Continuity	SLA procedure mechanisms	The SLA management module SHALL provide mechanisms to get an agreement presented and agreed between the parties	The involved parties need to be informed about the SLA offer from the other party and be able to agree or disagree

WORK IN PROGRESS

Accounting module

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
Ac.1	UC2	Management & Orchestration	Accounting notification - VNF starts	The accounting system SHALL know if a VNF starts correctly.	For billing purposes, the accounting system has to be notified about the start of the VNF service instance
Ac.2	UC 5	Market / commercial operability	Resources usage for billing	The accounting system SHALL store all the information about resources usage by each service for later billing purposes.	Billing procedure needs to know the services that have taken place
Ac.3	UC 5	Market / commercial operability	Price information for billing	The accounting system SHALL store the information about prices agreed by each customer for later billing purposes.	Billing procedure needs to know the price to be applied for service
Ac.4	UC 4.1, UC5	Market / commercial operability	SLA billable items	The accounting system SHALL receive and store the information about SLA fulfilment for billing compensations or penalties.	In order to compensate the customer economically for not achieving the SLA agreed, the billing system must have this information
Ac.5	UC5, UC1	Market / commercial operability	Bill cycle	The accounting module SHALL store the billing cycle information for each customer, and SP.	Billing procedure needs to know when a bill cycle finishes (for non pay- as-you-go services).

Billing module

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
Bil.1	UC 5	Market / commercial operability	Price information for customer billing	The billing module SHALL receive the information about prices agreed by each customer for each service.	Billing procedure needs to know the price to be applied for service
Bil.2	UC 5	Market / commercial operability	Price information for SP billing	The billing module SHALL receive the information about prices agreed by each SP for each VNF.	Billing procedure needs to know the price to be applied for service
Bil.3	UC 5	Market / commercial operability	Bill issuing	The billing module SHALL issue bills when the customer's bill cycle finishes or service pay-as-you-go finishes and stores them within the customer profile.	Billing procedure needs to know when a pay-as-you-go service finishes.

WORK IN PROGRESS

Bil.4	UC5		Billing-accounting interface	The billing SHOULD receive all the information needed for billing from the accounting module.	

Marketplace – orchestrator interface

Req. id	Domain	Requirement Name	Requirement Description	Justification of Requirement
IMO. 1	Orchestrator, Marketplace	Provide available VNFs	The Marketplace SHALL use this interface with the Orchestrator to provide the Service Provider with a list of the VNFs, so that it can select and parameterise them, or use them in the composition of a new network service.	It is assumed that this VNF metadata includes a URL/repository name from which to fetch the actual VNF software and install it on the previously allocated infrastructure (see NFVO.10 below). Note that, although this information will most certainly have to be cached on the Orchestrator's side for performance reasons, the available VNFs will be dynamic, so updates to this cached information will be rather frequent.
IMO. 2	Orchestrator, Marketplace	Provision a new network service	The Marketplace SHALL use this interface to inform the Orchestrator to provision the network service, after the Customer has selected and parameterised the network service. The Orchestrator SHALL read the SLA and the date/time to start the new network service. Each NS can be composed of one or more VNFs.	The date/time of start/end the service are part of the SLA.
IMO. 3	Orchestrator, Marketplace	Change configuration of a deployed network service	The Marketplace SHALL use this interface to change the configuration of an already provisioned network service on the Orchestrator.	Can this functionality be included in NFVO.02? It is assumed that information about scaling (up/down/in/out) is included in the SLA (or at least reasonable values can be inferred).
IMO. 4	Orchestrator, Marketplace	Provide network service state transitions	The Marketplace SHALL use this interface to know about the state transitions of a given network service,	It is assumed that each NS has a pre-defined state-diagram, like 'Ready to run', 'Running',

WORK IN PROGRESS

			e.g. to allow start and stop billing for the service.	'Stopped', etc., that is also known to the Marketplace.
IMO.5	Orchestrator, Marketplace	Provide network service monitoring data	The Marketplace SHALL use this interface to show the Customer how the subscribed network service is behaving, how it compares to the agreed SLA and bill the service usage.	This interface will very likely have to support very high volume traffic.
IMO.6	Orchestrator, Marketplace	Terminate a provisioned NS	The Marketplace SHALL use this interface to request the Orchestrator to terminate provisioned NSs	It is assumed that the impact on the dependent modules like billing, are taken care by the Marketplace (see NFVO.04). SLA Management is part of the Marketplace. Either after a customer's request or by the pre-defined ending date had been attained, the SLA Management notifies the Orchestrator of the end of the SLA.
IMO.7	Orchestrator, Marketplace	Secure communication	Interfaces between the Marketplace and the Orchestrator should be secured.	Encryption should be used, in order to ensure security against eavesdropping. Even between the Marketplace and the Orchestrator, since the Marketplace is really a set of distributed apps.

WORK IN PROGRESS

6.1.2. Detailed Network Function Store requirements specification

Req. id	Use Case	Domain	Requirement Name	Requirement Description	Justification of Requirement
FS.1	UC1.2	Operational	VNF Upload (or Publish)	The Function Store SHALL be able to store all the packaged VNFs and their associated metadata.	The system shall offer a method to the FP for uploading and storing the packaged VNF to the Function Store. When a particular VNF is requested the Orchestrator will instantiate this VNF to the appropriate NFVI-PoP.

FS.2	UC1.2	Security/ Operational	VNF validation	VNF validation SHOULD be checked.	The submitted VNF is validation by the T-NOVA Function Store in order to increase security and integrity of the VNF package. This is out of the scope of T-NOVA
FS.3	UC1.2	Security/ Operational	VNF Identification	The Function Store SHALL provide a unique identification ID to each certified, advertised VNF.	The VNF ID will be the reference name used by the system for monitoring purposes.
FS.4	UC1.2	Operational	VNF Modification/ Withdrawal	The Function Store SHALL allow modification/ withdrawal of the packaged VNFs and their associated metadata.	The system will offer a method to the FP for modification and/or withdrawal of the packaged VNF in the Function Store.
FS.5		Operational	VNF Download	The Function Store SHALL allow downloading of the packaged VNFs and their associated metadata.	The system will offer a method for downloading information associated to VNFs from the repository.
FS.6		Operational	VNF Directory	The Function Store SHALL allow listing of packaged VNFs and their associated metadata.	The system will offer a method to list the content of the data stored in the repository.
FS.7		Operational	VNF Notification	The Function Store SHALL be able to notify changes in its status.	The system will provide a mechanism to notify an update or change in the content of the repository. This is basically used to make the Orchestrator databases in sync with the Function Store.
FS.8		Security/ Operational	VNF Authentication	The Function Store SHALL allow access only to authenticated users.	The system (AA module) will grant operation and control of the Function Store upon authentication of FPs.
FS.9		Operational/ Management	VNF User Privilege	The Function Store SHALL provide different levels of privileges to users (e.g. user and root levels).	FP will have user access to the repository. In addition, the Function Store will offer a powerful management interface to administrators.
FS.10		Operational/ Performance	VNF User Concurrency	The Function Store SHOULD provide multi-user capability.	Concurrent requests should be managed by the Function Store repository.
FS.11		Operational/ Security	VNF Data Protection	The Function Store SHOULD guarantee security mechanisms for transmission of data.	The data should be protected (e.g. encrypted) when transferred to/from the Function Store repository.
FS.12		Operational/ Security	VNF Data Oblivion	The Function Store SHOULD guarantee that cancelled data will be completely disregarded.	The FP should be able to delete his/her data from the repository in a definite manner.
FS.13		Operational/ Performance	VNF Service Continuity	The Function Store SHOULD be available without interruption regardless of time or day (24/7/365).	The Function Store service should be offered as a nonstop service.

WORK IN PROGRESS

6.2. Annex B. Dashboard Mock-up

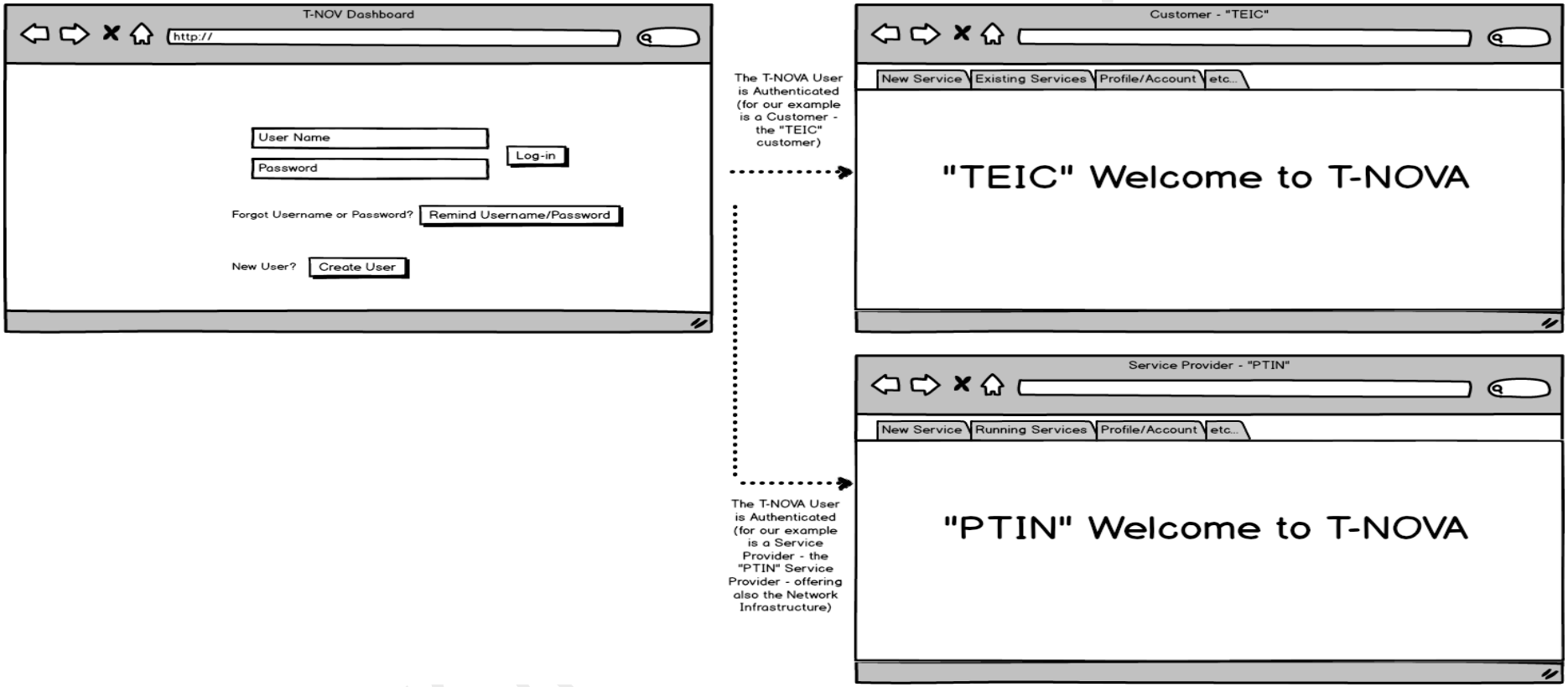


Figure 6-1 Dashboard Login Screen

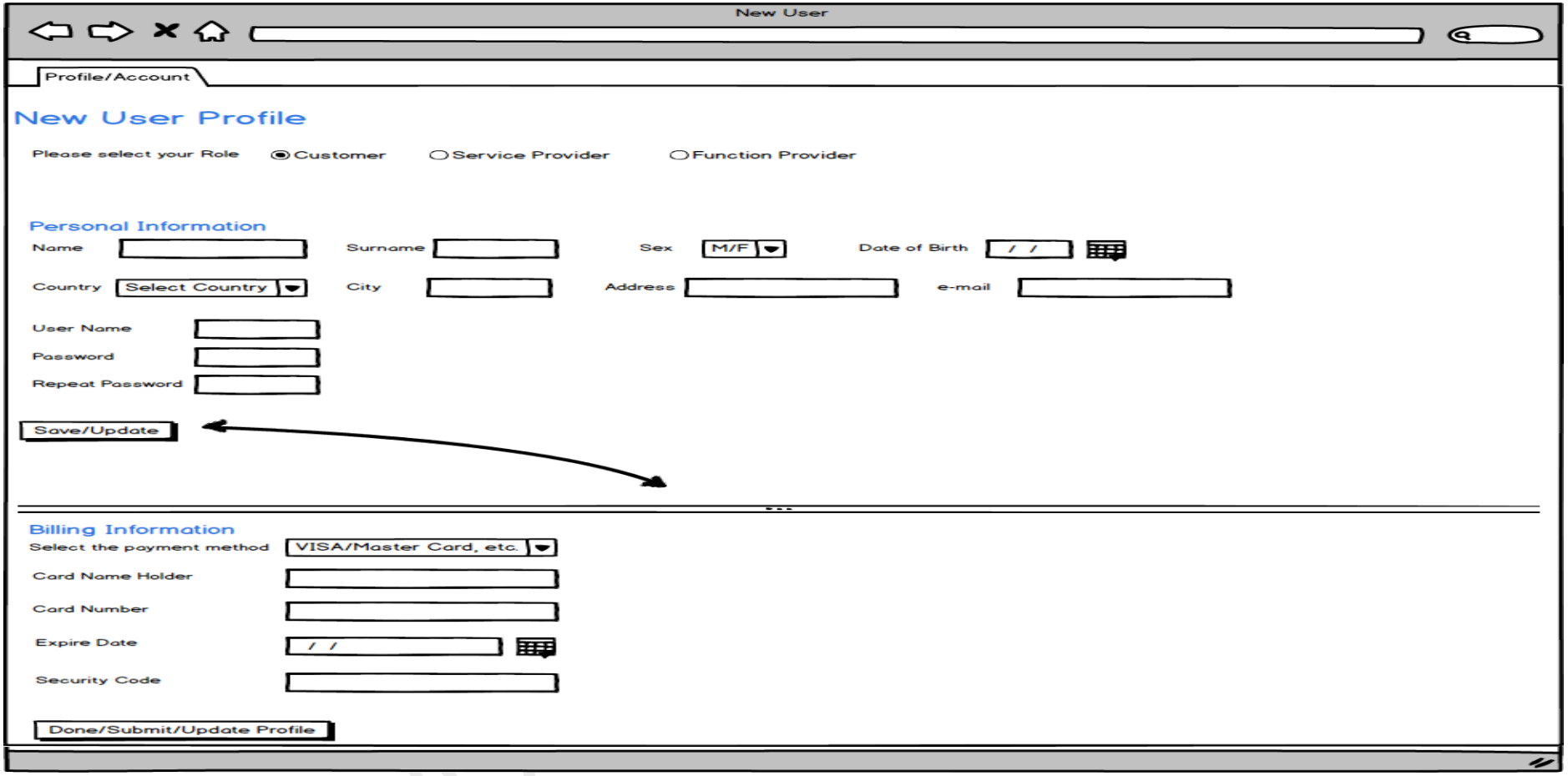


Figure 6-2 New User Profile

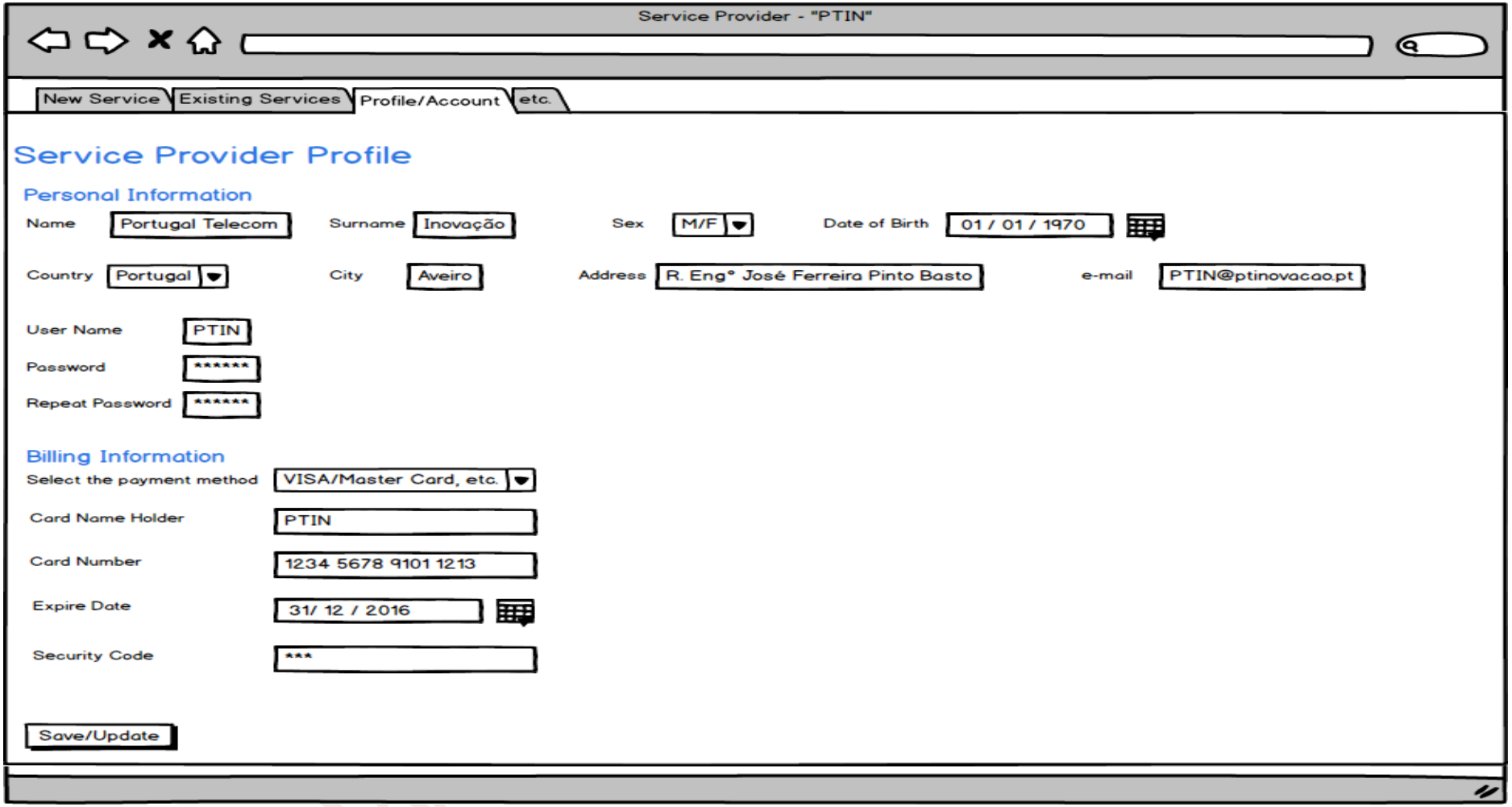


Figure 6-3 Service Provider Profile Screen

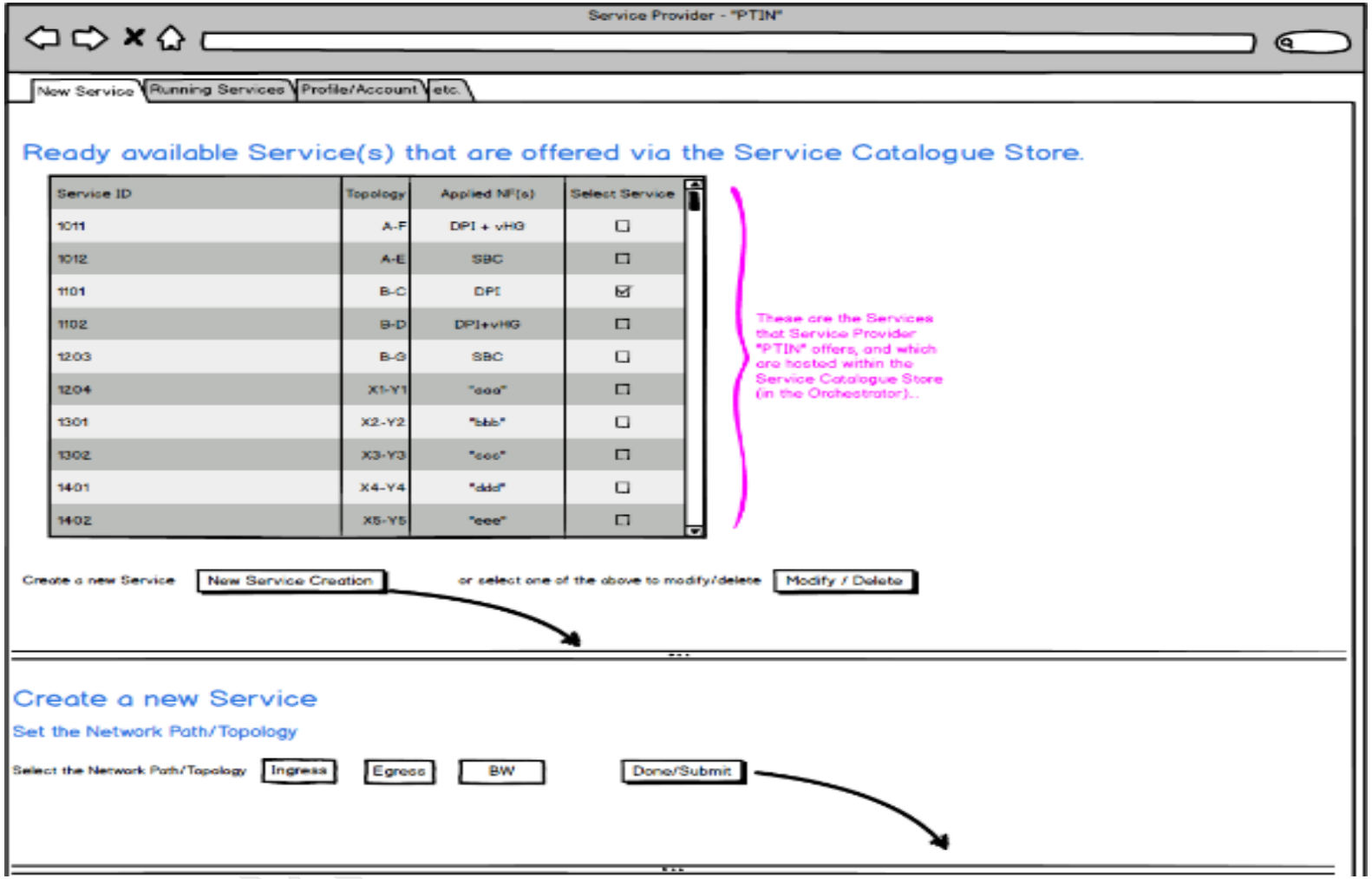


Figure 6-4 Service Provider New Service 1/2

Select the NF type to be deployed and define its main characteristics

Optional: Set the number of NF(s) to be deployed 1

Define the NF type to search for offerings DPI Search for NF(s) that can be offered at Fixed (F) Price / Tradable (T) Price OR Both (B) B

VM Memory Requirements less than 1GB VM Process Requirements less than 100PUs Billing method Fixed/Usage QoS????? Start Search

Maybe this is the Process Delay per packet or per MB????

FP Name	Function ID	Price (F/T)^v	Protocol Conf	Content Conf.	VM Memory Requir.	VM Process Requir.	Ranking/Popularity	Select Function
VIO	v001	100 (T)	Yes	Yes	1GB	100PU/Sec	#1	☐
VIO	v002	65 (F)	Yes	No	0.5GB	70PU/sec	#2	☐
FRAUNHOFER	FK003	65 (T)	No	Yes	0.7GB	70PU/sec	#3	☑
FRAUNHOFER	FK001	80(F)	Yes	No	0.3GB	60PU/sec	#4	☐
ITALTEL	IT012	85 (F)	Yes	No	0.5GB	80PU/sec	#5	☐
ITALTEL	IT022	65 (T)	No	Yes	0.7GB	70PU/sec	#6	☑

Select those for Trading with the corresponding NF Providers Start Trading OR directly select the NF (s) to be deployed Select NF(s)

Trade NF

Set the Start Time/Date / / Set the Stop Time/Date / / Select Trading Policy Auction Based with Closed / Open offers Open / Closed

Receive Offers

FP Name	Function ID	Price (F/T)^v	Protocol Conf	Content Conf.	VM Memory Requir.	VM Process Requir.	Ranking/Popularity	Select Function
FRAUNHOFER	FK003	60 (T)	No	Yes	0.7GB	70PU/sec	#3	☐
ITALTEL	IT022	55 (T)	No	Yes	0.7GB	70PU/sec	#6	☑

New Offers

Select the NF to be deployed by ticking it

Select the Network Nodes where the NF will be applied (NFV PoP) Ingress / Egress / Both

Create Service and store it in the Service Catalogue Store Done/Submit This Button registers the New Service to the "Service Catalogue Store" in the Orchestrator

Figure 6-5 Service Provider New Service 2/2

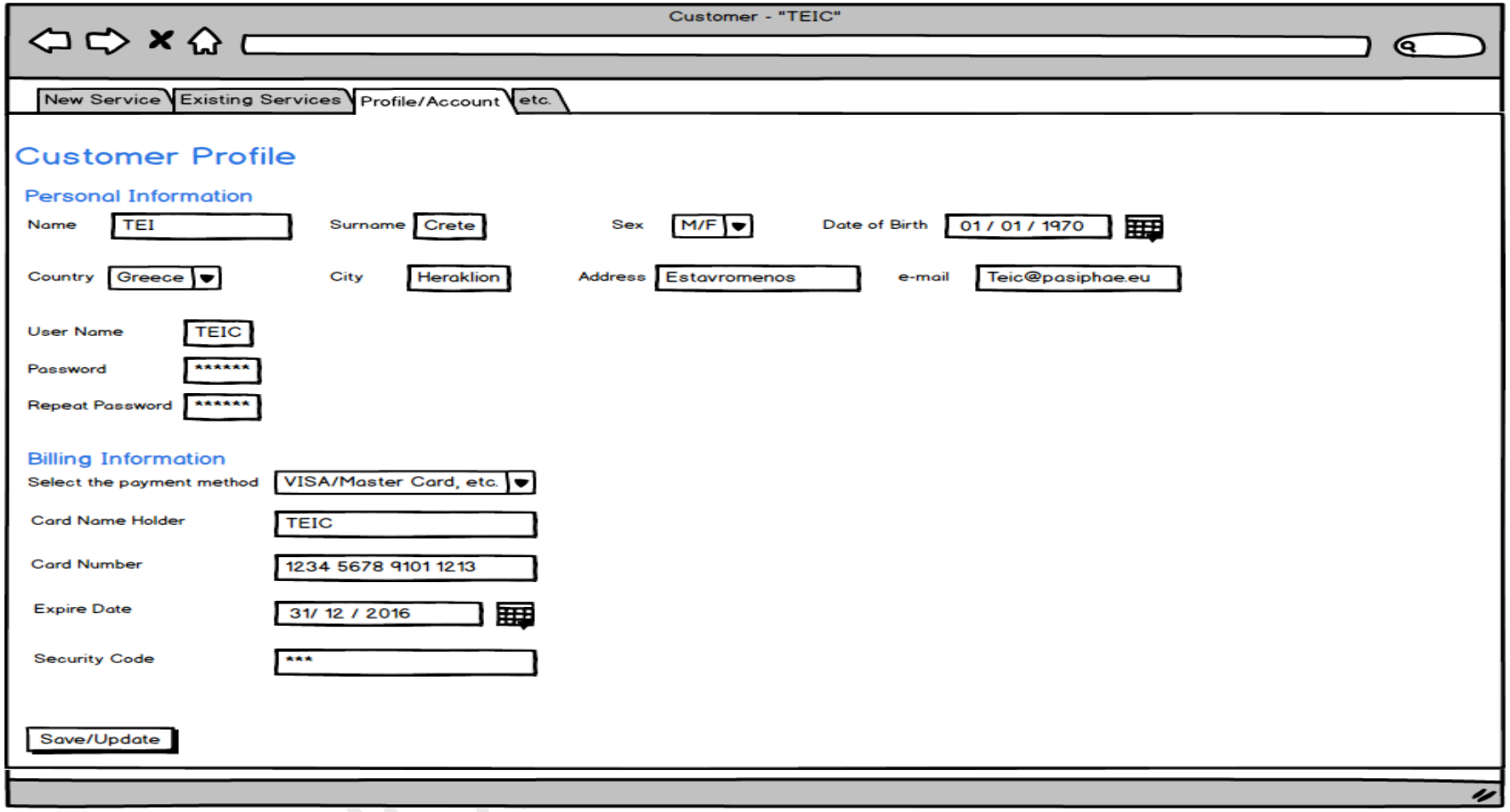


Figure 6-6 Customer Screen

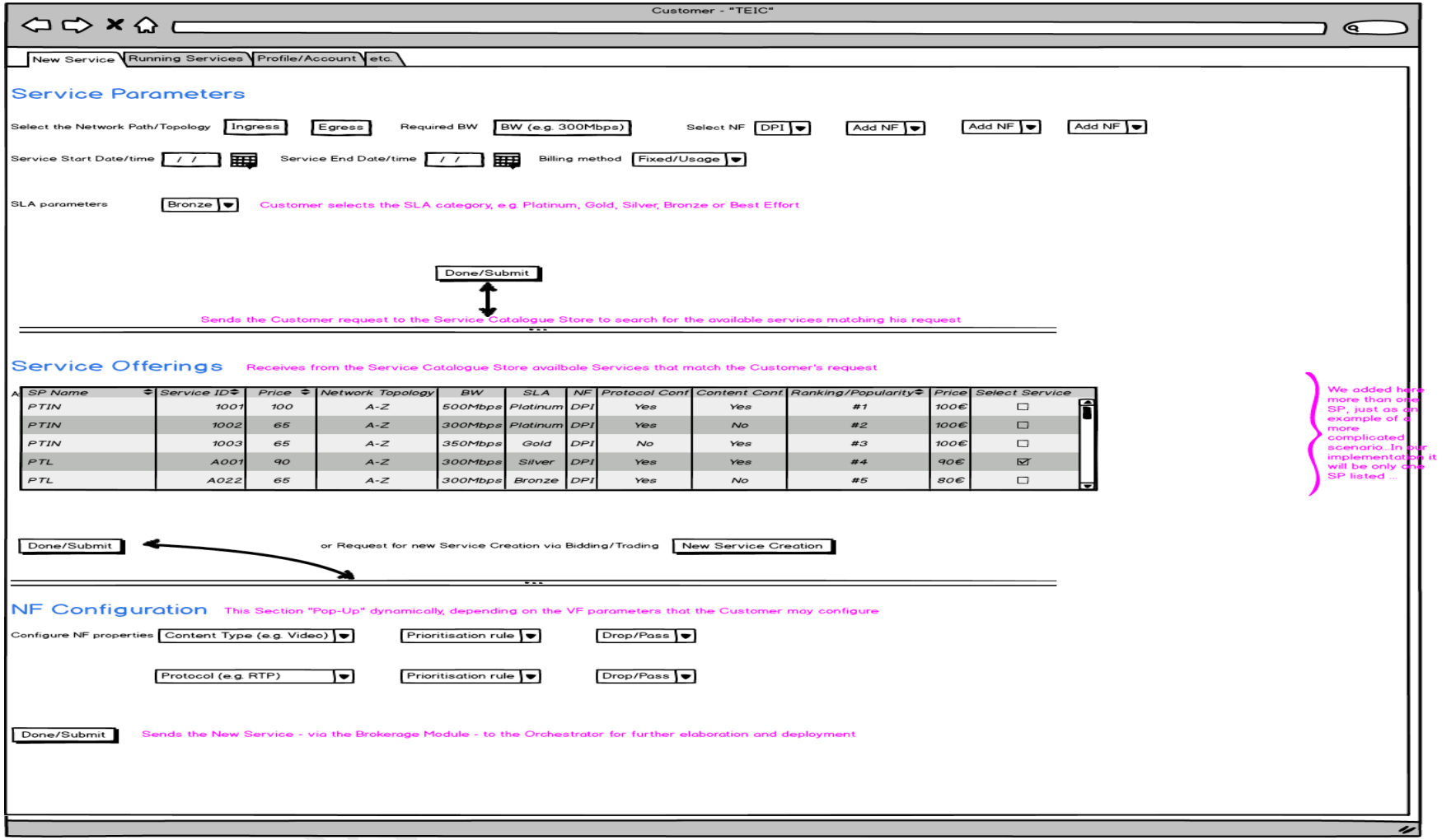


Figure 6-7 Custumer New Service

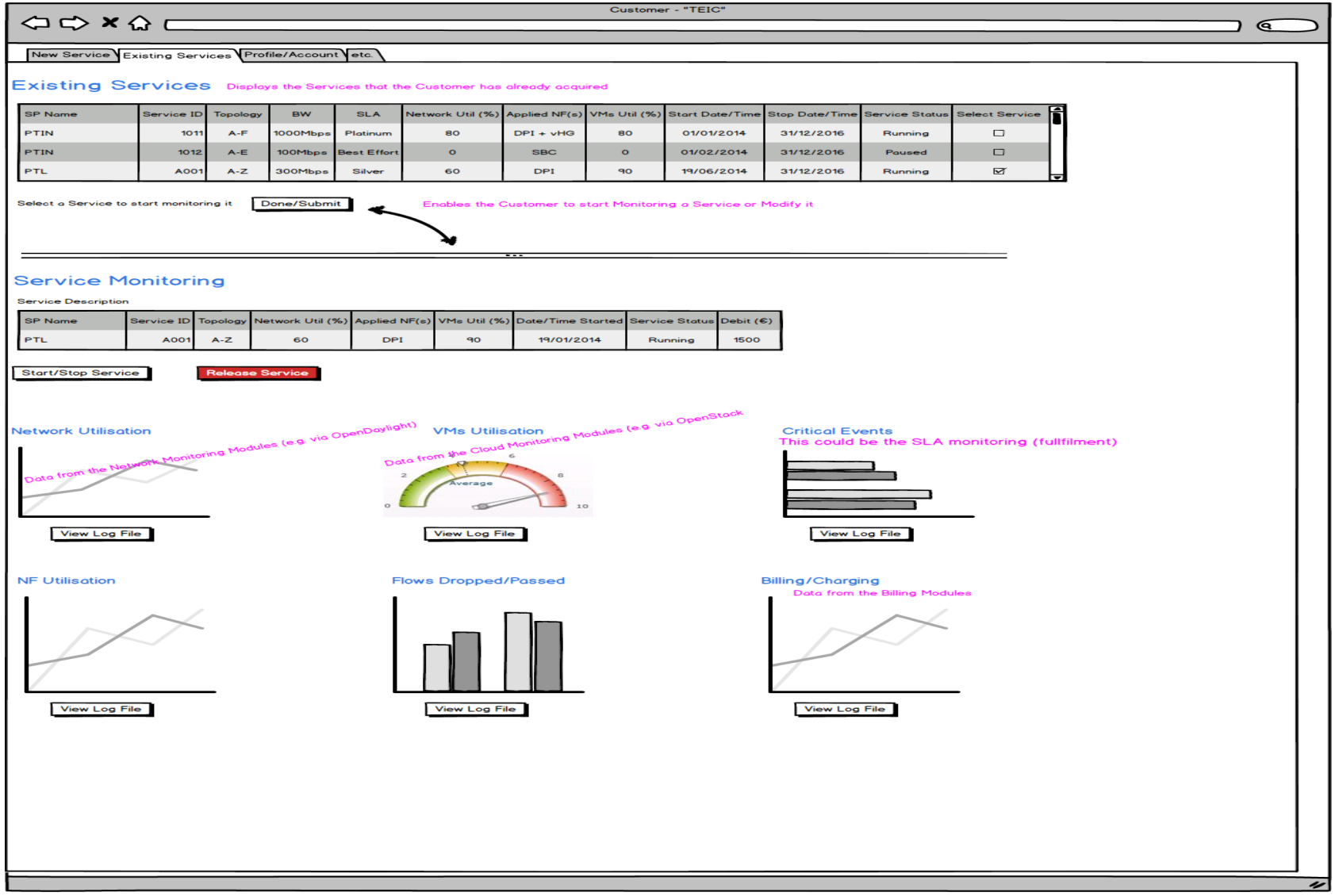
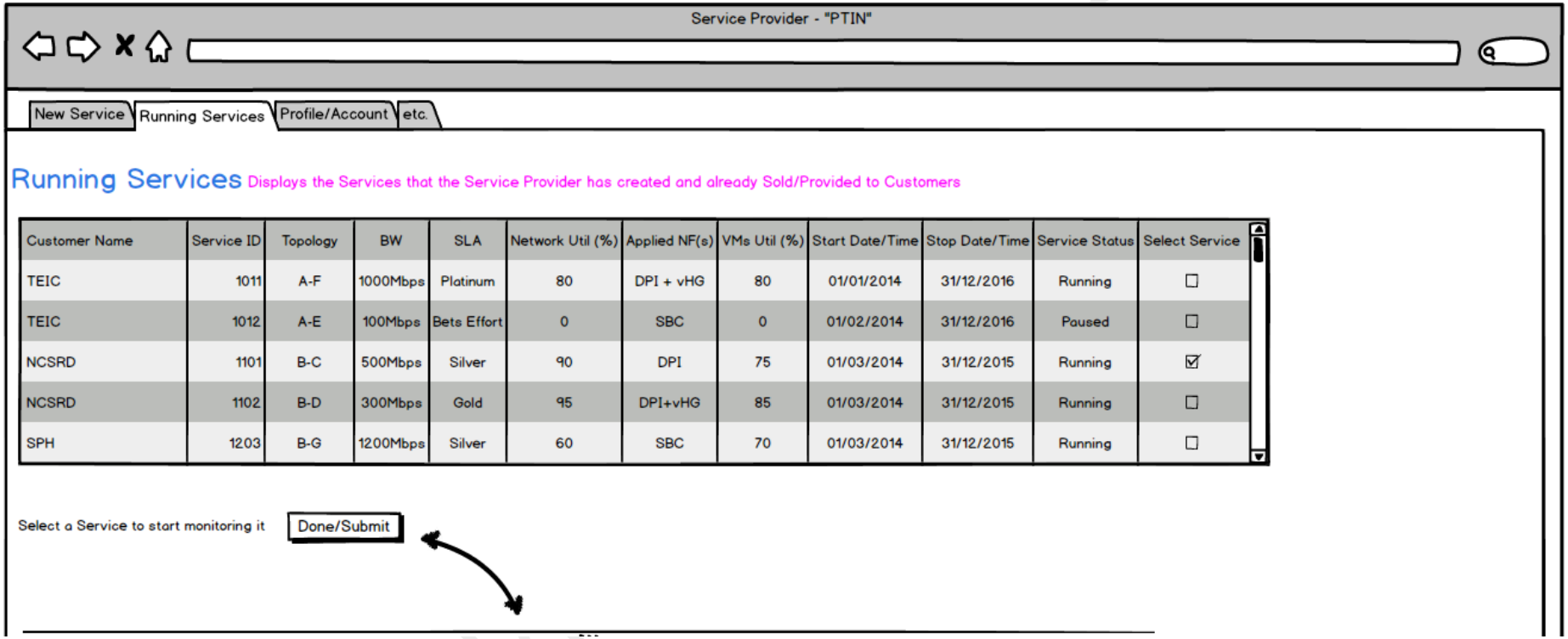

Customer - "TEIC"
New Service
Existing Services
Profile/Account
etc.
Existing Services
Displays the Services that the Customer has already acquired
SP Name | Service ID | Topology | BW | SLA | Network Util (%) | Applied NF(s) | VMs Util (%) | Start Date/Time | Stop Date/Time | Service Status | Select Service
PTIN | 1011 | A-F | 1000Mbps | Platinum | 80 | DPI + vHG | 80 | 01/01/2014 | 31/12/2016 | Running
PTIN | 1012 | A-E | 100Mbps | Best Effort | 0 | SBC | 0 | 01/02/2014 | 31/12/2016 | Paused
PTL | A001 | A-Z | 300Mbps | Silver | 60 | DPI | 90 | 19/06/2014 | 31/12/2016 | Running
Select a Service to start monitoring it
Done/Submit
Enables the Customer to start Monitoring a Service or Modify it
Service Monitoring
Service Description
SP Name | Service ID | Topology | Network Util (%) | Applied NF(s) | VMs Util (%) | Date/Time Started | Service Status | Debit (€)
PTL | A001 | A-Z | 60 | DPI | 90 | 19/01/2014 | Running | 1500
Start/Stop Service
Release Service
Network Utilisation
Data from the Network Monitoring Modules (e.g. via OpenDaylight)
View Log File
VMs Utilisation
Data from the Cloud Monitoring Modules (e.g. via OpenStack
Average
View Log File
Critical Events
This could be the SLA monitoring (fullfilment)
View Log File
NF Utilisation
View Log File
Flows Dropped/Passed
View Log File
Billing/Charging
Data from the Billing Modules
View Log File

Figure 6-8 Customer Existing Services



Customer Name	Service ID	Topology	BW	SLA	Network Util (%)	Applied NF(s)	VMs Util (%)	Start Date/Time	Stop Date/Time	Service Status	Select Service
TEIC	1011	A-F	1000Mbps	Platinum	80	DPI + vHG	80	01/01/2014	31/12/2016	Running	☐
TEIC	1012	A-E	100Mbps	Bets Effort	0	SBC	0	01/02/2014	31/12/2016	Paused	☐
NCSRD	1101	B-C	500Mbps	Silver	90	DPI	75	01/03/2014	31/12/2015	Running	☑
NCSRD	1102	B-D	300Mbps	Gold	95	DPI+vHG	85	01/03/2014	31/12/2015	Running	☐
SPH	1203	B-G	1200Mbps	Silver	60	SBC	70	01/03/2014	31/12/2015	Running	☐

Figure 6-9 Service Provider- Running Services 1/2

© T-NOVA Consortium

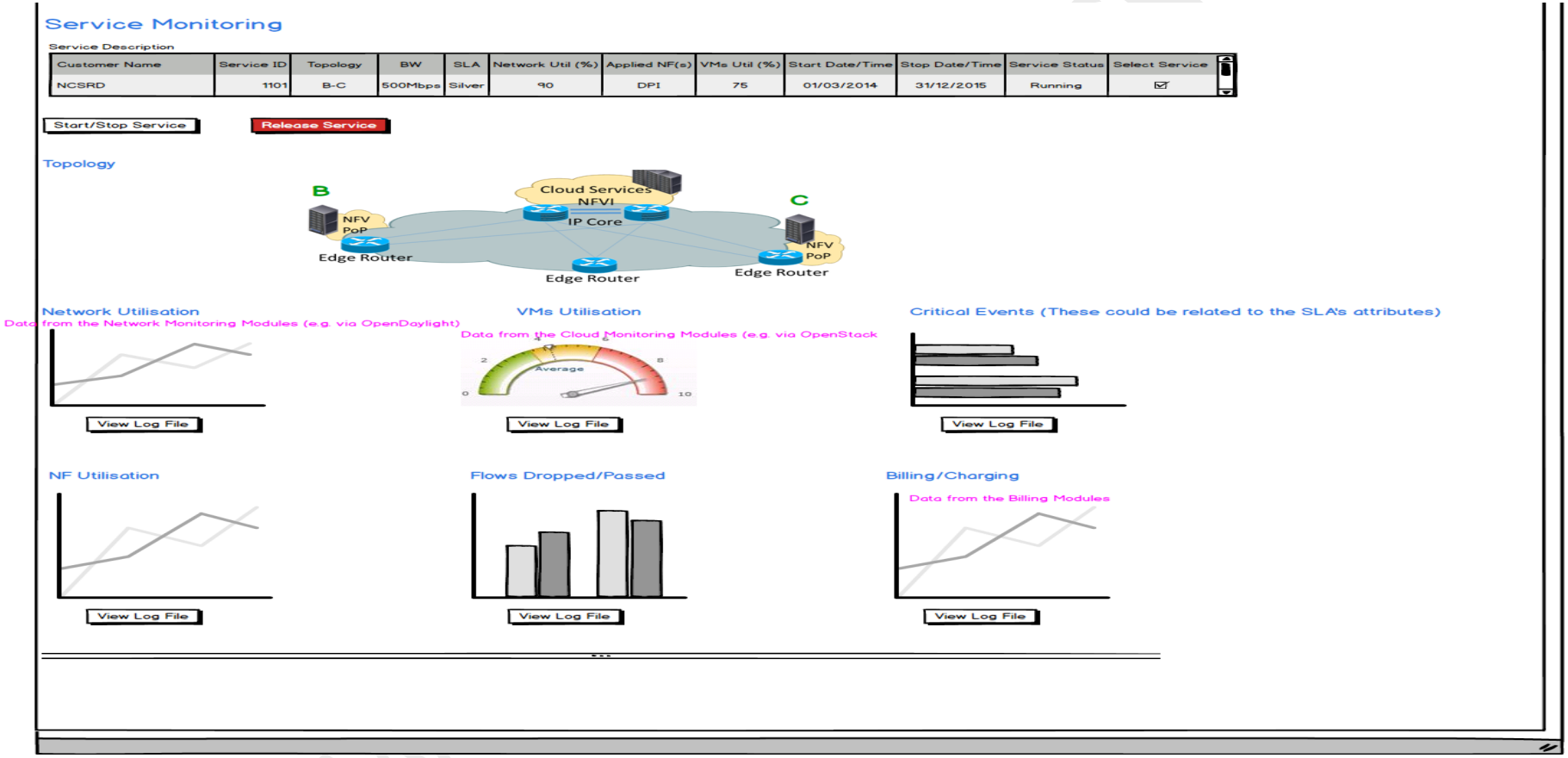


Figure 6-10 Service Provider- Running Services 1/2

7. REFERENCES

[1] Deliverable D2.1 – System Use Cases and Requirements. T- NOVA project.

[2] Deliverable D2.31 - Specification of the Infrastructure Virtualisation, Management and Orchestration - T-NOVA project.

[3] Deliverable D2.2 – Overall System Architecture and Interfaces. T- NOVA project.

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

[5] ETSI GS NFV-SWA 001" Network Functions Virtualisation (NFV)- Virtual Network Functions Architecture.

[6] TM Forum, "TM Forum WebSite," http://www.tmforum.org.

[7] TMF "TR227 TM Forum Specifications relevant for MANO Work V1.0.1", 2014.

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

[9] Project XIFI (FI-PPP), "Official Web Site," [Online]. Available: [Accessed 2014].https://www.fi-xifi.eu/home.html.

[10] TMForum Project "BandwithExchange" Available: [Accessed 2014] http://www.tmforum.org/BandwidthExchange/12792/home.html.

[11] TMF Catalyst Project: Service Bundling in a B2B2X Marketplace. Available: [Accessed 2014] http://www.tmforumlive.org/service-bundling-in-a-b2b2x/.

[12] CENX Ethernet Lifecycle Manager (ELM) Available: [Accessed 2014] http://www.cenx.com/.

[13] Equinix Platform – Marketplace. Available: [Accessed 2014] http://www.equinix.com/.

[14] V. Krishna, Auction Theory, 2010.

[15] Amazon AWS CloudFormation. Available online. [Accessed 2014]. http://aws.amazon.com/es/cloudformation/.

[16] OpenStackHot. Available [Accessed 2014] http://docs.openstack.org/developer/heat/template_guide/hot_spec.html..

[17] ETSI GS NFV 003 "Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV".

WORK IN PROGRESS

8. List of Acronyms

Acronym	Explanation
AA	Authentication and Authorisation
AAA	Authentication, Authorisation, and Accounting
API	Application Programming Interface
BSS	Business Support System
CRUD	Create Read Update and Delete
CPU	Central Processing Unit
DoW	Description of Work
eTOM	Telecom Operations Map
GUI	Graphical User Interface
EMS	Element Management System
ETSI	European Telecommunication Standard Institute
EU	End User
FI	Future Internet
FP	Function Provider
ISG	Industry Specification Group
ISP	Internet Service Provider
IT	Information Technology
KPI	Key Performance Indicator
MANO	Management and Orchestration
NFaaS	Network Functions-as-a-Service
NF	Network Function
NFC	Network Function Component
NFV	Network Functions Virtualisation
NFVI	Network Function Virtualization Infrastructure
NFVO	Network Function Virtualization Orchestrator
NS	Network Service
OSS	Operational Support System
QoS	Quality of Service
RBCA	Role Based Access Control

WORK IN PROGRESS

RTT	Round trip time
SaaS	Software-as-a-Service
SBC	Session Border Controller
SDK	Software Development Kit
SDN	Software-Defined Networking
SDO	Standards Development Organisation
SI	Service Integrator
SID	Shared Information/Data model
SIP	Session Initiation Protocol
SLA	Service Level Agreement
SP	Service Provider
TAM	Telecom Application Map
TIP	TM Forum Integration Program
UC	Use Case
VIM	Virtual Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Function
VNFaaS	Virtual Network Function as a Service
VNFD	Virtual Network Function Descriptor
VNFM	Virtual Network Function Manager
VNI	Virtual Network Interface
VNPaaS	Virtual Network Platform as a Service
WP	Work Package

WORK IN PROGRESS