H2020 5G-Transformer Project

Grant No. 761536

Definition of vertical service descriptors and SO NBI

Abstract

This deliverable defines vertical service descriptors and blueprints. It defines the initial architecture of the vertical slicer, the interface towards the verticals, and the API of the service orchestrator. Additionally, it defines the workflows among verticals and the vertical slicer as well as the workflows among the components of the vertical slicer.

Document properties

Document number	D3.1
Document title	Definition of vertical service descriptors and SO NBI
Document responsible	Thomas Deiß (NOK-N)
Document editor	Thomas Deiß (NOK-N)
Editorial team	Juan Brenes Baranzano (ATOS), Carla Chiasserini (POLITO), Thomas Deiß (NOK-N), Adlen Ksentini (EURECOM), Giada Landi (NXW), Luis Miguel Contreras Murillo (TID), Josep Mangues (CTTC), Nuria Molner (UC3M)
Target dissemination level	Public
Status of the document	Final
Version	1.0

Production properties

Reviewers	Carlos J. Bernardos (UC3M), Thomas Deiß (NOK-N), Chia-Lin Lai (ITRI), Giada Landi (NXW), Barbara Martini (SSSA), Céline Merlet (B-Com), Farouk Messaoudi (B-COM), Nuria Molner (UC3M), Jorge Martin Pérez (UC3M), Charles Turyagenda (IDCC)

Disclaimer

This document has been produced in the context of the 5G-TRANSFORMER Project. The research leading to these results has received funding from the European Community's H2020 Programme under grant agreement № H2020-761536.

Table of Contents

List of Contributors

Partner Short Name	Contributors
UC3M	Luca Cominardi, Antonio Pastor, Carlos J. Bernardos
TEI	Giuseppe Imbarlina, Paola Iovanna
ATOS	Juan Brenes Baranzano
NOK-N	Thomas Deiß
IDCC	Charles Turyagyenda
TID	Luis Miguel Contreras Murillo
ORANGE	Thouraya Toukabri
CRF	Marina Giordanino
BCOM	Farouk Messaoudi, Cao-Thanh Phan, Céline Merlet
NXW	Giada Landi, Marco Capitani
CTTC	Josep Mangues, Ricardo Martínez, Iñaki Pascual, Jorge Baranda, Javier Vílchez, Laia Nadal
POLITO	Carla Chiasserini
EURECOM	Adlen Ksentini
ITRI	Shazoob Bilal, Chia-Lin Lai

List of Figures

List of Tables

List of Acronyms

Acronym	Description
5GC	5G Core
5GT-MTP	Mobile Transport and Computing Platform
5GT-SO	Service Orchestrator
5GT-VS	Vertical Slicer
AAA	Authentication, Authorization, Accounting
AN	Access Network
API	Application Programming Interface
AppD	Application Descriptor
AS	Application Server
BSS	Business Support System
CN	Core Network
CP	Connection Point
CPD	Connection Point Descriptor
CSMF	Communication Service Management Function
DF	Deployment Flavour
DoS	Denial of Service
E2E	End to end
eMBB	enhanced Mobile Broadband
EPC	Evolved Packet Core
EPCaaS	EPC as a Service
ETSI	European Telecommunication Standardization Institute
EVPN	Ethernet VPN
FIFO	First-in First-out
FM	Fault Management
GRE	Generic Routing Encapsulation
HSS	Home Subscriber Server
IaaS	Infrastructure as a Service
ICA	Intersection Collision Avoidance
KPI	Key Performance Indicator
L2VPN	Layer 2 VPN
L3VPN	Layer 3 VPN
LC	Lifecycle
LCM	Lifecycle Management
LCMO	LCM Operation
M&E	Media and Entertainment
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
MEO	Multi-access Edge Orchestrator
MEP	Multi-access Edge Platform
MIoT	Massive Internet of Things
MME	Mobility Management Entity
MNO	Mobile Network Operator
MTC	Machine Type Communication
MVNE	Mobile Virtual Network Enabler
MVNO	Mobile Virtual Network Operator
NaaS	Network as a Service
NBI	Northbound Interface
NF	Network Function

NFP	Network Forwarding Path
NFV	Network Function Virtualization
NFV-NS	NFV Network Service
NFV-NSO	Network Service Orchestrator
NFVI	Network functions virtualisation infrastructure
NFVIaaS	NFVI as a Service
NFVO	NFV Orchestrator
NFVO-RO	Resource Orchestrator
NS	Network Slice
NS-DF	Network Service Deployment Flavour
NSaaS	Network Slice as a Service
NSD	Network Service Descriptor
NSDF	Network Service Deployment Flavour
NSI	Network Slice Instance
NSMF	Network Slice Management Function
NSSI	Network Slice Subnet Instance
NSSMF	Network Slice Subnet Management Function
NST	Network Slice Template
OLE	On site Live Event Experience
OSS	Operating Support System
PLMN	Public Land Mobile Network
PM	Performance Management
PNF	Physical Network Function
PNFD	PNF Descriptor
PoP	Point of Presence
QoS	Quality of Service
RAN	Radio Access Network
REST	Representational State Transfer
SAP	Service Access Point
SAPD	SAP Descriptor
SCS	Service Capability Server
SDK	Software Development Kit
SDO	Standard Developing Organization
SLA	Service Level Agreement
SLAT	Service Level Agreement Template
SLO	Service Level Objective
SLPOC	Single Logical Point of Contact
SP	Service Platform
SPGW-C	Serving/Packet Data Network Gateway Control Plane
SPGW-U	Serving/Packet Data Network Gateway User Plane
SST	Slice/Service Type
TETRA	Terrestrial Trunked Radio
TN	Transport Network
TOSCA	Topology and Orchestration Specification for Cloud Applications
TSC	5G-TRANSFORMER Service Consumer
TSP	5G-TRANSFORMER Service Provider
TTL	Time to Live
UC	Use Case
UE	User Equipment
URLLC	Ultra-reliable Low Latency Communication
VA	Virtual Application

vEPC	virtual Evolved Packet Core
VIM	Virtual Infrastructure Manager
VL	Virtual Link
VLD	Virtual Link Descriptor
VM	Virtual Machine
VNF	Virtual Network Function
VNFD	VNF Descriptor
VNFDF	VNF Deployment Flavour
VNFFG	VNF Forwarding Graph
VNFFGD	VNFFG Descriptor
VPN	Virtual Private Network
VSC	Vertical Service Consumer
VSD	Vertical Service Descriptor
VSI	Vertical Service Instance
VSP	Vertical Service Provider
WIM	Widearea network Infrastructure Manager
YAML	YAML Ain't Markup Language

Executive Summary and Key Contributions

This deliverable introduces the vertical slicer (5GT-VS) as the main entry point for verticals to the 5G-TRANSFORMER system. The 5GT-VS allows verticals to describe vertical services by selecting a blueprint and providing missing information. The 5GT-VS then translates Vertical Service Descriptors (VSD) into ETSI NFV Network Service Descriptors (NSD), used to describe network slices. The 5GT-VS takes also care of arbitration among vertical service instances in case of resource shortage.

The main contributions in this deliverable are the definition of the requirements on the 5GT-VS, the design of the internal architecture of the 5GT-VS and its Northbound and Southbound Interfaces (NBI, SBI), a definition of the descriptors used, and the workflows among 5GT-VS internal components. Next, we enumerate these key contributions and the associated outcomes:

- The **requirements**, see Section 3, on the 5GT-VS are split into two major groups of business related and functional requirements. The functional requirements are grouped along the different phases of a vertical service instance: discovery, fulfillment, assurance, and decommissioning.
- The **design of the internal architecture of the 5GT-VS**, see Section 5, defines components to manage vertical services and network slices and corresponding catalogues. Two key components of the 5GT-VS are 1) the VSD/NSD Translator component, which takes a vertical service descriptor and maps it to a network service descriptor, and 2) the Arbitrator component, which ensures that resources of a vertical are made available to high-priority vertical services. An initial description of the 5GT-VS and its use of vertical service descriptors and blueprints has been described in [7] and [8].
- The **5GT-VS NBI** provides the operations to verticals to select blueprints, prepare vertical services, and to control their lifecycle. The northbound interface provides also the operations for the 5G-TRANSFORMER service provider to manage vertical service blueprints, tenants, and the SLA agreements with them.
- The **5GT-VS SBI**, which coincides with the 5GT-SO NBI, provides the operations to manage and control network services. This interface is based to a large extend on the corresponding interface of ETSI NFV. Necessary extensions are described in this deliverable.
- The **definition of vertical service blueprints (VSB), VSDs, and NSDs**, see Section 6, provides an easy for a vertical to define a vertical service. For VSDs we propose a tabular format with fields to indicate general information such as a description of a vertical service, structural information on the service in terms of atomic functional components and their connections, and fields to express SLA and other requirements. We have defined fields to describe network slices as defined in [41]. VSBs are similar to VSDs, but certain fields or elements of fields have been left open to be provided by a vertical.
- The **workflows among the components of the 5GT-VS**, a vertical, and the 5GT-SO indicate which information is passed in which order among these components, see Section 7. Here we focus - among others - on the most important workflows, such as onboarding a vertical service blueprint, preparing a vertical service descriptor from it, and instantiating a corresponding vertical service instance.

These main contributions are complemented by the 5G-TRANSFORMER system architecture [9], see Section 2, a description of the state of the art on service descriptions, see Section 4, and an outlook to future work, see Section 8. A list of use cases and references architectures are described in two annexes, see Section 10 and Section 11. The summary of use cases and the 5G-TRANSFORMER system architecture is text common to [3], [4], and this deliverable.

In this deliverable we focus on the support for verticals to deploy services by selecting and completing entries from the blueprint catalogue. In addition to the example VSBs and VSDs in this deliverable, we plan to provide blueprints and corresponding VSDs for the use cases of this project, see [1], Section 5. We plan to investigate, how the blueprint based approach of defining vertical services can be complemented by allowing the verticals to compose their vertical services from smaller building blocks. This would allow more flexibility to the verticals to define their services. In addition to support the instantiation of vertical services, we plan to investigate further the support for NFVIaaS and NSaaS. If needed, the 5GT-VS architecture will be extended.

Essential aspects of the 5GT-VS will be implemented [6] and demonstrated [5] during the project. We aim to demonstrate that the 5GT-VS actually allows verticals to define their services in an easy way and to deploy them in a short timeframe. This will enable 5G-TRANSFORMER service providers to offer their 5G and general IT infrastructure to verticals, enabling new business cases. The demonstrations for the 5GT-VS will focus on the use cases of the vertical industries participating in this project.

1 Introduction

The vertical slicer (5GT-VS) is the entry point of verticals to the 5G-TRANSFORMER system. It allows the verticals to define their vertical services from a catalogue of blueprints. A vertical can then instantiate, update, terminate, and monitor its vertical service instances (VSI). Verticals may define pools of resources and priorities to use them for their VSIs. The 5GT-VS arbitrates resources among VSIs in case of resource shortage in the infrastructure, thereby ensuring that resources are available to high priority services.

The 5GT-VS supports verticals in creating a Vertical Service Descriptor (VSD) from a blueprint, including the specification of service levels. It maps such VSDs to Network Service Descriptors (NSD), describing the network slice instance (NSI) into which the vertical service is deployed. These NSDs are then passed to the underlying service orchestrator (5GT-SO) for the actual deployment of the vertical service instances.

Within 5G-TRANSFORMER the 5GT-VS and the overall 5G-TRANSFORMER system are used for use cases from different vertical domains. The 5GT-VS is a part only of the overall 5G-TRANSFORMER system. The use cases and the architecture of the 5G-TRANSFORMER system are summarized in Section 10 and Section 2, resp., such that the 5GT-VS can be seen within its context. In this section, we also relate the 5GT-VS to reference architectures such as ETSI NFV, see Section 11. The 5G-TRANSFORMER system architecture will be described in more detail in [2], the mobile transport and computing platform (5GT-MTP) in [3] and the 5GT-SO in [4].

The use cases of the verticals have technical requirements on the actual vertical services. The use cases imply requirements as well on the 5G-TRANSFORMER system and specifically on the 5GT-VS on how to define, instantiate, and operate a vertical service. We present these requirements in Section 3, including those related to business relations.

The 5GT-VS can be considered as part of operating and business support systems (OSS/BSS) on top of orchestration functionality. Research efforts so far have concentrated on managing infrastructure and virtual network functions (VNF) and to orchestrate them. Nevertheless, we present state of the art related to the 5GT-VS in Section 4, here we focus on descriptions for network services.

Starting from the requirements on the 5GT-VS we describe the architecture of the 5GT-VS itself in Section 5. We present an overview of its architecture and describe the components of the 5GT-VS. The main components are the VSI/NSI Coordinator and LC manager to define vertical services from blueprints, to handle their lifecycle, and to handle the lifecycle of network slices, the VSD/NSD Translator to translate vertical services to network services, and the Arbitrator to make decisions among vertical service instances in case of resource shortage. Additional components handle monitoring and provide catalogues of defined or instantiated vertical services. We define the northbound interfaces (NBI) of the 5GT-VS and of the 5GT-SO. The NBI of the 5GT-SO, coinciding with the southbound interface (SBI) of the 5GT-VS was jointly developed with WP4.

So far, we have just mentioned vertical service descriptors and blueprints. In Section 6 we define vertical service descriptors and blueprints. Thereby we consider a blueprint as an incomplete or parameterized VSD, allowing a vertical to tailor a blueprint to its specific needs. The parameters can be of quite different types, being IP addresses of other servers, latency constraints, or names of virtual machine (VM) images. The actual values of these parameters are used in the mapping to NSDs, which are used at the 5GT-SO to describe network slices. NSD are based on ETSI NFV IFA 014 [38] with extensions to include multi-access edge (MEC) applications and to express location constraints. This definition of NSDs has been joint work with WP4.

The dynamic aspects of the system are described in Section 7. Here we describe the workflows among verticals and the 5GT-VS and within the 5GT-VS. Other system-level workflows among the 5GT-VS and the 5GT-SO are described in [2].

In Section 8 we conclude this deliverable and provide an outlook to further work on the 5GT-VS within the 5G-TRANSFORMER project.

In order to keep the main body of the document as short as possible, several annexes are included at the end, containing additional information and results.

2 5G-TRANSFORMER System Overview

To describe the 5GT-VS within its context, we present in this section a summary[1] of the system architecture described in [2]. The use cases used to define the system architecture are presented in an annex in Section 10 and relevant reference architectures for the 5G-TRANSFORMER system architecture in Section 11.

The 5G-TRANSFORMER project explores how the network can better serve the needs of 5G-TRANSFORMER customers (i.e., vertical industries and M(V)NOs) by offering the abstraction, flexibility, and dynamic management capabilities they require. In terms of notation, it is important to differentiate between (vertical) service, i.e., that is requested by the customer of the 5G-TRANSFORMER system, from the underlying network service deployed to fulfill the requirements of the vertical. An example of the former is a car manufacturer requesting the deployment of an automotive intersection collision avoidance service. The latter will be deployed in the form of an NFV network service, in general.

The key architectural concept to support such adaptation to the needs of verticals and M(V)NOs is network slicing. The term network slice aligns network functionality to business needs [18], since it allows customers to request not just functions, but also business objectives (e.g., quality of service, security, etc.), as a sort of intent. The scope of a slice may be a single customer facing service (using TM Forum terminology [30]) or a group of such services. The system will also allow infrastructure providers to share the 5G mobile transport and computing infrastructure efficiently among verticals and M(V)NOs, hence enhancing 5G-TRANSFORMER provider network usage efficiency. In terms of deployment, network slices can be implemented by means of ETSI NFV network services.

The architecture is conceived to support multiple combinations of stakeholders by introducing open Application Programming Interfaces (API) among components [1]. Through these APIs, the system hides unnecessary details from the verticals, allowing them to focus on the definition of the services and the required Service Level Agreements (SLAs). As for interfaces, particularly relevant for the goals of the project are east-westbound interfaces, which enable service and resource federation across different administrative domains, allowing 5G-TRANSFORMER service providers to enhance their service offerings to their customers by peering with other providers.

We envision a system of three major components: vertical slicer (5GT-VS), service orchestrator (5GT-SO) and mobile transport and computing platform (5GT-MTP), see Figure 1. The 5GT-VS is the entry point for the vertical requesting a service, and it handles the association of these services with slices as well as network slice management. The 5GT-SO is responsible for the end-to-end orchestration of services across multiple domains and for aggregating local and federated (i.e., from peer domains) resources and services and exposing them to the 5GT-VS in a unified way. Finally, the 5GT-MTP provides and manages the virtual and physical IT and network resources on which service components are eventually deployed. It also decides on the abstraction level offered to the 5GT-SO.

[1] This is text common to [3], [4], and this document.

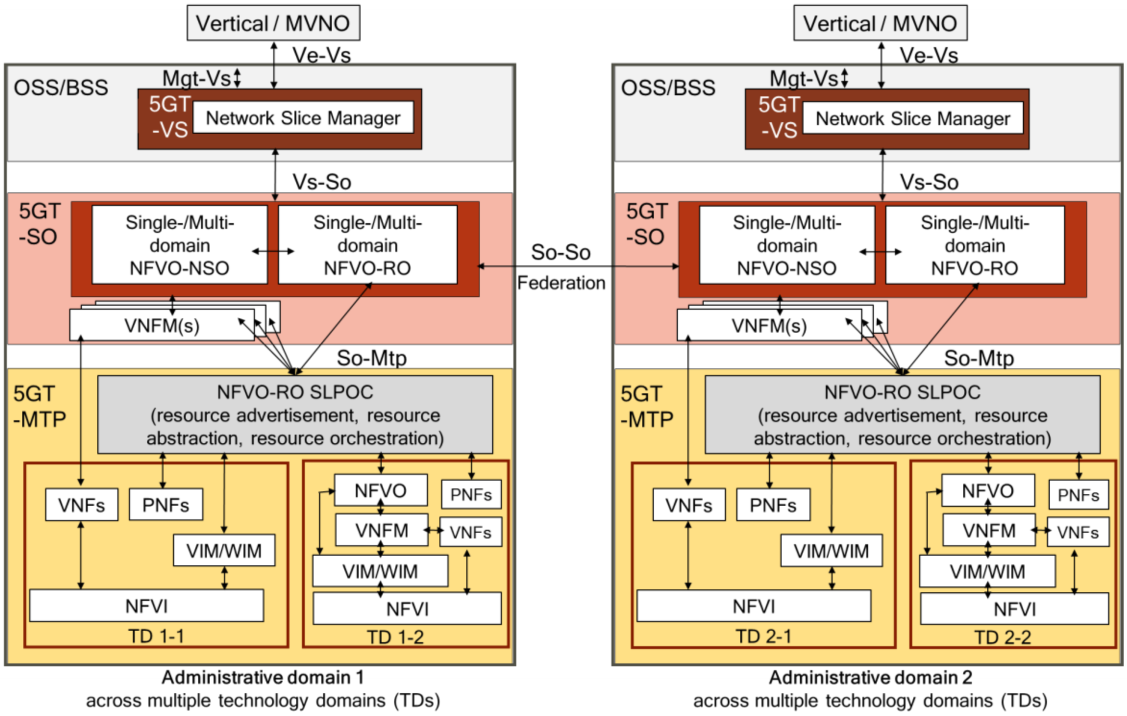


FIGURE 1: 5G-TRANSFORMER SYSTEM ARCHITECTURE

2.1 Vertical Slicer (5GT-VS)

The 5GT-VS is the common entry point for all verticals into the 5G-Transformer system. It is part of the operating and business support systems (OSS/BSS) of the 5G-TRANSFORMER service provider (TSP) [1]. Vertical services are offered through a high-level interface at the 5GT-VS northbound that is designed to allow verticals to focus on the service logic and requirements, without caring on how they are eventually deployed at the resource level. This latter issue would be up to 5GT-SO. Therefore, vertical services will use those services offered by the TSP. In fact, the 5GT-VS offers a catalogue of vertical service blueprints, based on which the vertical service requests are generated by the vertical. The role of the 5GT-VS is to trigger the actions allowing the 5G-TRANSFORMER system to fulfill the requirements of a given incoming service request. After the appropriate translation between service requirements and slice-related requirements by the VSD/NSD Translator and Arbitrator, corresponding to the Communication Service Management Function (CSMF), as defined in [13], a decision is taken on whether the service is included in an already existing slice or a new one is created.

The vertical slicer is the component of the system that is conscious of the business needs of the vertical, their SLA requirements, and how they are satisfied by mapping them to given slices. Intra-vertical arbitration is also part of the vertical slicer, by which intra-vertical contention is resolved to prioritize those services that are more critical, according to the agreed SLA.

The VSI/NSI Coordinator and LC Manager is the core component of the 5GT-VS. It contains functionality that can be mapped to that of the Network Slice Management

Function (NSMF) and Network Slice Subnet Management Function (NSSMF), as defined in [13]. More specifically, the NSMF is in charge of lifecycle management of network slice instances. All possible combinations between vertical services and network slices exist; that is, a network slice can be shared by different vertical services, but a given vertical service may be mapped to multiple network slices as well. In turn, network slices may be composed by network slice subnets, which may offer part of the functionality required by a given network slice. And network slice subnets may be shared by multiple network slices.

The final result of all this process is a request sent by the 5GT-VS towards the 5GT-SO to create or update the NFV network services (NFV-NS) that implement the slices.

In summary, through this process, the 5GT-VS maps vertical service descriptions and instantiation parameters at the vertical application level into an NFV-NS (existing or new) implementing the network slice. In turn, such NFV-NS will be updated or created through a network service descriptor (NSD), which is a service graph composed of a set of virtual (network) functions (V(N)F) chained with each other, and the corresponding fine-grained instantiation parameters (e.g., deployment flavour) that are sent to the 5GT-SO. Given the operations carried out through it, the VS-SO interface takes ETSI NFV IFA013 [17] as basis.

2.2 Service Orchestrator (5GT-SO)

The NFV-NS from the 5GT-VS is received by the 5GT-SO through the VS-SO interface. The main duty of the 5GT-SO [10] is to provide an end-to-end orchestration of the NFV-NS across multiple administrative domains by interacting with the local 5GT-MTP (So-Mtp reference point) and with the 5GT-SOs of other administrative domains (So-So reference point). If needed (e.g., not enough local resources), the 5GT-SO interacts with 5GT-SOs of other administrative domains (federation) to take decisions on the end-to-end (de)composition of virtual services and their most suitable execution environment. Even if a service is deployed across several administrative domains, e.g., if roaming is required, a vertical still uses one 5GT-VS to access the system, and so, the 5GT-SO hides this federation from the 5GT-VS, and thus, the verticals.

The 5GT-SO embeds the network service orchestrator (NFV-NSO) and the resource orchestrator (NFVO-RO) with functionalities equivalent to those of a regular NFV orchestrator and it may be used for single and multi-domains [11].

Since the network slices handled at the 5GT-VS will in general serve complex end-to-end services, in the general case, the corresponding network service will be a composition of nested NFV-NSs. The lifecycle management of this complex NFV-NS is the role of the NFV-NSO.

In case a network service is requested that must be distributed across multiple domains, the 5GT-SO receiving the request becomes the parent NFV-NSO and sends ETSI NFV IFA013 [17] requests for each of the constituent NFV-NSs to other NFV-NSOs. Therefore, a hierarchy of NFVO-NSOs is established. The child NFVO-NSOs may belong to the same 5GT-SO that received the request from the 5GT-VS or to a peer 5GT-SO, which, in turn, may establish an additional hierarchy, which is transparent for the parent NFVO-NSO. The child NFVO-NSO belonging to the same 5GT-SO would be in charge of the lifecycle management of the constituent service that is eventually deployed over the local 5GT-MTP, i.e., the 5G-MTP with which the 5GT-SO has a direct relationship through the SO-MTP interface. When a child NFVO-NSO belongs to a different domain, there is service federation.

Eventually, a resource-related request is generated towards the underlying NFVO-RO to assign virtual resources towards the deployment of the (constituent) NFV-NS. The NFVO-RO functionality of the 5GT-SO handles resources coming from the local 5GT-MTP (real or abstracted) and from the 5GT-SOs of other administrative domains (abstracted). The NFVO-RO will decide on the placement of the Virtual Network Functions (VNF) of the NFV-NS based on the information available in the NFVI resources repository and the NFV instances repository. Most likely, the information available in these repositories will be more detailed when coming from the local 5GT-MTP than from a federated domain.

As for the NFV infrastructure as a service (NFVIaaS) use case, the 5GT-VS will request the 5GT-SO for a set of virtual resources, as opposed to a complete E2E NFV-NS as before. Therefore, this request is directly handled by the NFVO-RO, which is in charge of allocating resources either from the local 5GT-MTP or from a peer 5GT-SO. The latter option corresponds to resource federation. In this case, the request from the local NFVO-RO will reach the NFVO-RO of the peering domain. In all cases, the interaction between NFVO-ROs is based on ETSI NFV IFA005 [19]. This also includes the interface with the 5GT-MTP, where an additional NFVO-RO lower in the hierarchy is embedded, as explained below.

Notice that the NFVI resources handled by the NFVO of the 5GT-SO based on which decisions are taken will have a higher or lower abstraction level depending on the policies applied in this respect by the 5GT-MTP and the peering 5GT-SO. In general, the NFVO-RO of the local 5GT-SO will take coarse-grained decisions and the 5GT-MTP and peer 5GT-SO will take finer-grained ones, since they are closer to the actual resources.

The 5GT-SO also embeds the Virtual Network Function Managers (VNFM) to manage the lifecycle of the VNFs composing the NFV_NS. ETSI NFV IFA006-based interfaces [43] will be used to allow the VNFM interacting with the NFVO-RO Single Logical Point of Contact (SLPOC) entity in the 5GT-MTP, as well as peer SOs for resource allocation requests involving the VNFs under its control. For managing the VNF instances, ETSI NFV IFA008-based interfaces [31] will be used to allow the VNFM to directly configure the VNF instances running in the 5GT-MTP.

2.3 Mobile Transport and Computing Platform (5GT-MTP)

The 5GT-MTP [12] is responsible for orchestration of resources and the instantiation of V(N)Fs over the infrastructure under its control, as well as managing the underlying physical mobile transport network, computing and storage infrastructure. In general, there will be multiple technology domains (TD) inside an 5GT-MTP (e.g., data centres, mobile network, wide area network). The 5GT-MTP NFVO-RO acts as end-to-end resource orchestrator across the various technology domains of the 5GT-MTP. The computing and storage infrastructure may be deployed in central data centres as well as distributed ones placed closer to the network edge, as in MEC [16]. Therefore, the 5GT-MTP is in charge of managing the virtual resources on top of which the NFV-NSs are deployed.

In terms of resource orchestration, the NFVO-RO acts as a single entry point, i.e., the single logical point of contact (SLPOC) in ETSI NFV IFA028 [22] terminology, for any resource allocation request coming from the 5GT-SO. The SO-MTP interface is based on ETSI NFV IFA005 [19] and ETSI NFV IFA006 [43]. The former allows the NFVO-RO of the 5GT-SO to request resource allocations to the NFVO-RO of the 5GT-MTP, whilst the latter allows the VNFM of the 5GT-SO to request resource allocations for the VNF under its control.

In terms of managing VNF instances, the SO-MTP interface also consists of ETSI NFV IFA008-based interfaces [31] to allow the VNFM of the 5GT-SO to directly configure the VNF instances running in the 5GT-MTP.

Depending on the use case, the 5GT-MTP may offer different levels of resource abstraction to the 5GT-SO. However, the 5GT-MTP NFVO-RO has full visibility of the resources under the control of the Virtual Infrastructure Managers (VIM) managing each technology domain, since they belong to the same administrative domain. ETSI NFV IFA005-based interfaces [19] are deployed between the 5GT-MTP NFVO-RO and the 5GT-MTP VIMs. Therefore, when receiving a resource allocation request from the 5GT-SO, the 5GT-MTP NFVO-RO generates the corresponding request to the relevant entities (e.g., VIM or WAN Infrastructure Manager (WIM)), each of them providing part of the virtual resources needed to deploy the VNFs and/or configure the relevant parameters of the PNFs that form the NFV-NS. As a special case, a resource request may be translated into an ETSI NFV IFA013-based NFV-NS request [17] towards a mobile network technology domain [11]. This option is offered to hide the complexity of the mobile network to the rest of the system whilst keeping the required flexibility inside the mobile domain (e.g., to decide on the most appropriate functional split). Therefore, a full ETSI MANO stack is represented in technology domain 1-2 (see Figure 1) even if the focus of the 5GT-MTP is handling virtual resources and not NFV-NSs. In any case, this NFV-NS is hidden to the 5GT-SO, since it is abstracted as a virtual link.

2.4 Monitoring Architecture

In the 5G-TRANSFORMER framework, each architectural component (i.e. 5GT-VS, 5GT-SO, 5GT-MTP) includes a monitoring service able to provide performance metrics and failure reports targeting the logical entities managed by each component. Following this approach, the 5GT-MTP monitoring service will produce monitoring data about the local physical and virtual resources, the 5GT-SO monitoring service will produce monitoring data about the managed VNFs and NFV network services, while the 5GT-VS monitoring service will produce monitoring data about network slices and vertical services. This hierarchy of monitoring services is shown in Figure 2, where the arrows indicates a consumer-provider interaction. In particular, the 5GT-SO monitoring service can be a consumer of the monitoring service provided by the underlying 5GT-MTP or by a federated 5GT-SO, while the 5GT-VS can be a consumer of the monitoring service provided by the local 5GT-SO.

The monitoring data generated at each layer can be used to feed internal decisions within each architectural component or to serve external consumers of monitoring data. For example, the 5GT-SO monitoring service can elaborate performance metrics about an NFV network service, and these metrics can be used by the 5GT-SO to take scaling decisions for the involved VNFs. On the other hand, the performance metrics computed at the 5GT-SO monitoring service can be provided to the 5GT-VS monitoring service for further elaboration. When metrics and alerts are exchanged between two monitoring services, the level of visibility and disclosure of monitoring information should be regulated based on authorization policies and business agreements, in particular when monitoring data that belongs to different administrative entities. This may be the case, for example, between the 5GT-MTP and the 5GT-SO monitoring services when they are handled by different actors or between the monitoring services of federated 5GT-SOs.

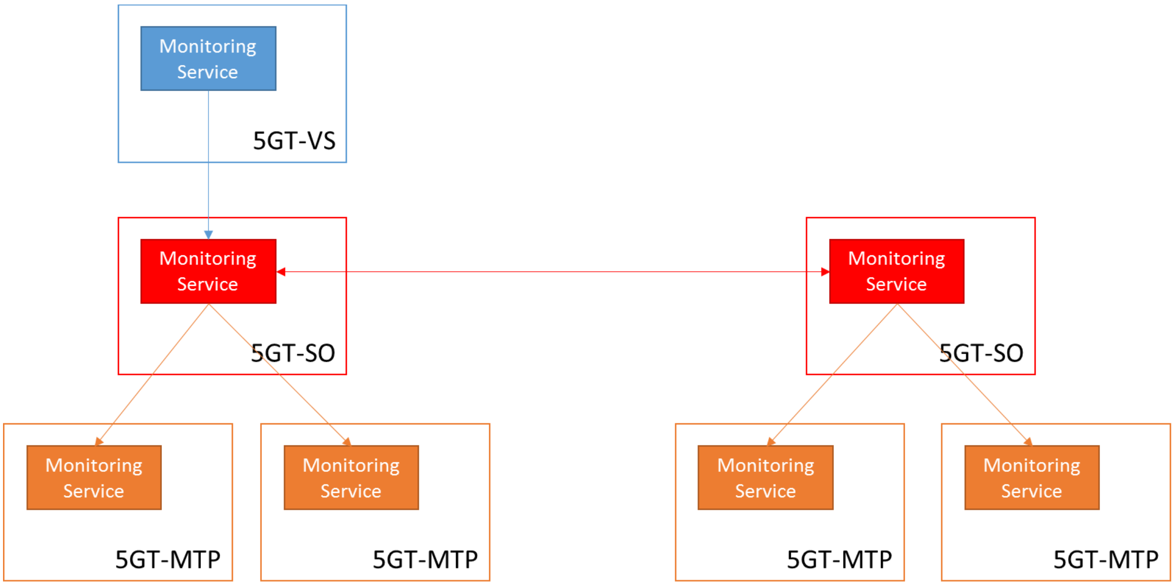


FIGURE 2: HIERARCHY OF MONITORING SERVICES IN 5G-TRANSFORMER ARCHITECTURE

It is important to highlight that the 5G-TRANSFORMER architecture does not impose any constraint on the monitoring platform implementation, but defines just the expected behavior of the service and the external APIs that each monitoring platform should expose to the consumers of its monitoring data. This means that each actor may implement its own specific monitoring platform and in case of overlapping roles, like in the 5GT-VS and 5GT-SO case where they are owned and managed by the same administrative entity, a single monitoring platform may be deployed for both of them.

3 Requirements of the 5GT-VS

Technical requirements on the 5G-TRANSFORMER system and the vertical services have been defined already in [1]. These requirements have focused on properties of the vertical services and the corresponding network slices. More general requirements on the 5G-TRANSFORMER system are described in [2]. In this section, we will define specific requirements on the 5GT-VS. In Section 3.1 we present requirements from the business perspective point of view, in Section 3.2 we present requirements corresponding to the different phases of the lifecycle of vertical services. The notation used for requirements and generic requirements on vertical services, derived from ETSI NFVI IFA 013 [17] on network services are described in annexes, see Sections 12 and 13.

3.1 Business Requirements

The 5GT-VS forms part of the business front-end of the 5G-TRANSFORMER system. It is a component that directly interacts with the customer through the service request, which is internally transformed into a network slice instantiation.

To provide context for the business requirements we describe first the kind of services we consider. We base this on a categorization of services by 3GPP, see [13].

- The first distinguishing feature is the type of business service. The service provided to 5G-TRANSFORMER service customers can be E2E connectivity as a service (CS), network slice as a service (NSaaS), and virtualized infrastructure as a service (NFVIaaS). The type of business service is further detailed below.
- The second important property is the service type which is characterized in the form of functional and non-functional (performance, scalability, reliability, etc.) descriptions of the network slice. 3GPP in SA2, see [14] and Table 1, has defined some values for slice/service types (SST) such as enhanced Mobile Broadband (eMBB), Ultra-Reliable Low Latency Communication (URLLC) and massive IoT (mIoT); however the consumer and the provider may have a common understanding of service behavior and a mutually agreement on a service type definition. For instance, the service provider may dedicate a service type other than V2X for a collision avoidance service.

TABLE 1: 3GPP STANDARDIZED VALUES OF NETWORK SLICE TYPES

Slice/Service type	SST value	Characteristics
eMBB	1	Slice suitable for the handling of 5G enhanced Mobile Broadband
URLLC	2	Slice suitable for the handling of ultra- reliable low latency communications
MIoT	3	Slice suitable for the handling of massive IoT

Existing and future networks will be intrinsically multi-service, then this view from 3GPP can be augmented to cover other potential business needs than 5G. Some potential new SSTs to consider are described in Table 2:

TABLE 2: ADDITIONAL SLICE SERVICE TYPES

Slice/Service type	SST value (proposed)	Characteristics
Enterprise	4	Slice suitable for enterprise connectivity services. This kind of service typically connects a number of sites from a given organization, including voice, data and video (e.g., connection to private data centers for database access, tele-presence communications, etc.).
(NFV)IaaS	5	Slice suitable for the handling and operation of an (NFV) infrastructure, focused on the availability of raw compute and storage resources

To improve the slice designation accuracy, additional requirements specific to the service type (e.g. the end-device class related to the transmission mode in MIoT is not significant in an eMBB slice may be also considered. In 5G-TRANSFORMER, the vertical service extends the 3GPP communication service to all kind of services a vertical can provides to its end-users. It includes E2E connectivity to applications, E2E connectivity and/or applications. In addition, E2E connectivity can be based on 3GPP and non-3GPP managed components. Table 3 summarizes the different levels of categorization.

TABLE 3: HIERARCHY LEVELS OF CATEGORIZATION OF NETWORK SLICE, [13]

Level 1	Level 2	Optional	Attributes and their ranges
Categorization based on Business Service Type (VSaaS, NSaaS, NFVIaaS)	Categorization based on service characterization eMBB, URLLC, mIoT, other non-standardized slice types	Additional levels (3, 4, 5), e.g. exposure levels. Service types with higher variation in service requirements may be further classified into sub-categories to allow an operator to offer a finer granularity to its customers. For example, eMBB type may be classified into QoS ranges, or QoS based categorization may be classified into user to server and user to user service.	e.g.: Capacity, throughput, delay, number of users, geographical identifications, authentication level etc.

The following business-related requirements are identified:

TABLE 4: BUSINESS REQUIREMENTS

ID	Requirement	F/NF
ReqVS.B.01	The 5GT-VS should include a portal to interface with the vertical customer.	F

ReqVS.B.02	The 5GT-VS shall support different kinds of vertical customers.	NF
ReqVS.B.03	The 5GT-VS shall be open and extensible to support any new kind of vertical customer.	NF
ReqVS.B.04	The 5GT-VS should be able to accept the service specification from the vertical customer including both functional and non-functional requirements expected for the requested service.	F
ReqVS.B.05	The 5GT-VS should be agnostic with regard to any collaboration of federated 5G-TRANSFORMER administrative domains for the deployment of the service.	F
ReqVS.B.06	The 5GT-VS shall expose appropriate interfaces to external customers to allow them to consume services offered by the 5G-TRANSFORMER system.	F
ReqVS.B.07	The 5GT-VS shall be able to perform translation and mapping of the vertical customer service request to network slice(s) implementing such service.	F
ReqVS.B.08	The 5GT-VS shall adhere to industry multi-tenancy requirements including isolation, scalability, elasticity and security, where security is meant to provide protection to prevent attacks, denial of service or information leaking.	NF
ReqVS.B.09	The 5GT-VS shall allow to negotiate and monitor service SLAs, with appropriate granularity according to the final service characteristics.	F
ReqVS.B.10	The 5GT-VS shall provide vertical customers with service catalogue information about available service offers and capabilities, in order to facilitate the automated provisioning of services.	F
ReqVS.B.11	The 5GT-VS shall provide a mechanism to perform vertical service accounting and charging. This information should be available internally and externally (for the vertical customer).	F
ReqVS.B.12	The 5GT-VS should be able to support long-live and short-lived services, i.e. the 5GT-VS should be able to express / contain duration characteristics.	F
ReqVS.B.13	The 5GT-VS should be reliable.	NF
ReqVS.B.14	The 5GT-VS should be available (as carrier class component providing 5 nines availability.).	NF
ReqVS.B.15	The 5GT-VS should keep responsiveness for vertical customer requests. The 5GT-VS should provide response times as an interactive system.	NF
ReqVS.B.16	The 5GT-VS shall allow blueprints composed of other blueprints.	F

ReqVS.B.17	The 5GT-VS shall allow VSDs composed of other VSDs.	F
ReqVS.B.18	The 5GT-VS shall allow a vertical to specify which vertical service instance to use in a composed vertical service.	F
ReqVS.B.19	The 5GT-VS shall support the specification of preferred, non-preferred, and prohibited virtual infrastructure providers.	F
ReqVS.B.20	The 5GT-VS system shall allow a vertical to define whether a constituent service of a composed service instance has the same lifecycle as the composed service instance or whether it has a lifecycle of its own.	F
ReqVS.B.21	The 5GT-VS system shall allow a vertical to define whether a vertical service instance can be a constituent service of several composed services, i.e. whether it can be shared among other vertical service instances.	F
ReqVS.B.22	The 5GT-VS shall support to specify indirectly the deployment area based on KPIs[2] of another service.	F

3.2 Functional Requirements

The 5GT-VS is involved in the service lifecycle at different phases, each having different requirements. The phases are discovery, fulfillment, assurance, and decommissioning. All requirements in this section are functional ones, therefore the corresponding marking in the table was removed.

3.2.1 Discovery

The discovery phase facilitates the 5GT-VS to understand what are the capabilities and services (in terms of descriptors) supported by the 5GT-SO. That information will be exposed to the vertical customers for 5G-TRANSFORMER service offering.

The following requirements are identified:

TABLE 5: REQUIREMENTS ON THE DISCOVERY PHASE

ID	Requirement
ReqVS.Di.01	The 5GT-VS must provide vertical customers with the means to send detailed requests including information regarding the location of resources, the location of service points, QoS, charging options.
ReqVS.Di.02	Service catalogue entries may contain a service manifest and a price tag (or an indicative price range from which the exact price can be extracted at run-time).
ReqVS.Di.03	The 5GT-VS shall provide the customer with the ability to request a service from the catalogue along with the expected SLA.

[2] As an example, intersection collision avoidance should cover critical intersections, where 'critical' is defined in terms of occurrence of abrupt braking manoeuvres in the past.

ReqVS.Di.04	Service catalogue entries and satisfied service requests should result in an SLA commitment for the corresponding service.
ReqVS.Di.05	The 5GT-VS should be able to keep up to date the network service catalogue entries as informed by the 5GT-SO.
ReqVS.Di.06	The 5GT-VS shall be able to translate the service request to one or more vertical service descriptors reflecting the agreed service levels.
ReqVS.Di.07	The 5GT-VS should keep association among the network slices generated as result of a single service request.
ReqVS.Di.08	The service request (and associated SLA) must contain a parameter describing the service aging or Time To Live (TTL).
ReqVS.Di.09	The 5GT-VS must support both private, i.e. towards specific vertical customer(s), and public dissemination of service offers.
ReqVS.Di.10	The 5GT-VS should provide a mechanism to set-up, scale, and terminate services.
ReqVS.Di.11	The 5GT-VS shall allow a vertical to create several instances of the same vertical service.
ReqVS.Di.12	The 5GT-VS shall allow a vertical to store its service descriptions persistently, and to create, retrieve, update, and delete vertical service descriptions[3].
ReqVS.Di.13	The 5GT-VS shall allow the TSP to onboard vertical service blueprints.

3.2.2 Fulfilment

During the service fulfilment phase, the 5GT-VS produces the necessary mapping from the customer request to the network slice templates, i.e. network service descriptors, to properly instruct the 5GT-SO.

The following requirements are identified:

TABLE 6: REQUIREMENTS ON THE FULFILMENT PHASE

ID	Requirement
ReqVS.Fu.01	Depending on the modality[4] of the contracted service, the 5GT-VS could be required to offer proper configuration and management interfaces to the slice capabilities honoring the service request, in order to manage them similarly as if they were owned and dedicated resources (tenant-managed slices).

[3] A vertical may provide the location of virtual machine images of its virtual applications as part of its service descriptions. These images may have to be certified by the 5G-TRANSFORMER system provider before actually onboarding them to the 5G-TRANSFORMER system.

[4] Modality refers to the possibility of requesting tenant-managed network slices (i.e., request of control and management capabilities of the allocated resources functions) or provider-managed network slices (i.e. the control and management is retained by the provider and the tenant simply uses the network slice).

ReqVS.Fu.02	Depending on the modality of the contracted service, the 5GT-VS could accommodate a vertical service instance in some existing network slice (provider-managed slices).
ReqVS.Fu.03	The 5GT-VS shall support the vertical to express policies[5].
ReqVS.Fu.04	The 5GT-VS could allow the vertical customer to specify policies[6] associated to the service to describe e.g. elasticity rules to be enforced when the service requires re-deployment/re-configuration.
ReqVS.Fu.05	The 5GT-VS should allow slice aggregation[7] (and optimization) as part of the slice lifecycle management.
ReqVS.Fu.06	The 5GT-VS shall support a vertical to manage the lifecycle of each of its vertical service instances separately.

3.2.3 Assurance

The 5GT-VS informs the vertical customer about events and performance of its vertical service instances. In consequence, the 5GT-VS should gather monitoring information and expose it to the vertical, who can take different actions accordingly. In addition, the monitoring information can be also consumed internally by the 5GT-VS to trigger events like arbitration.

The following requirements are identified:

TABLE 7: REQUIREMENTS ON THE ASSURANCE PHASE

ID	Requirement
ReqVS.As.01	The 5GT-VS must provide the vertical customer with tools to monitor the QoS attained for the requested service.
ReqVS.As.02	The 5GT-VS should provide a mechanism for the vertical for reward/penalty in service provisioning in case of SLA conformance/failure.
ReqVS.As.03	The 5GT-VS should be able to map the associated SLA into KPIs[8] to be monitored during the slice execution lifecycle.
ReqVS.As.04	The 5GT-VS should be able to support lawful interception mechanisms.
ReqVS.As.05	The 5GT-VS should prevent DoS attacks from a malicious behavior from vertical customers.

[5] QoS policies, charging policies, security policies, and regulatory policies where needed.
[6] As an example of such policies, it could be possible to specify the concentration of resources by night for an efficient management of energy consumption during low workloads.
[7] Aggregation means to scale up/down an existing slice, i.e. adding or removing resources according to the demanded services.
[8] The SLA will be expressed in terms of service parameters, while the KPIs represent technical parameters. For example, service latency (as part of the SLA) could require monitoring transport network latency but also VM execution latency (as two different KPIs to monitor).

ReqVS.As.06	The 5GT-VS should provide isolation among vertical customers' workflows and requests status.
ReqVS.As.07	The 5G-VS shall allow to dynamically set-up a traffic monitoring service in any given 5G network slice.
ReqVS.As.08	The 5GT-VS shall arbitrate resources among vertical service instances of one vertical based on priorities and SLA requirements.

3.2.4 Decommissioning

The 5GT-VS also participates in the proper release of services to the vertical once a service instance is no longer required.

The following requirements are identified:

TABLE 8: REQUIREMENTS ON THE DECOMMISSIONING PHASE

ID	Requirement
ReqVS.De01	The 5GT-VS should be able to identify the slice(s) to be decommissioned as a result of a service instance termination.
ReqVS.De.02	The 5GT-VS should be able to identify the monitoring mechanisms to be de-activated as a result of a service instance termination.
ReqVS.De.03	The 5GT-VS should have means for receiving acknowledgement of releasing resources.
ReqVS.De.04	The 5GT-VS should be able to notify the vertical customer about service instance termination.
ReqVS.De.05	The 5GT-VS should be able to keep track of charging and accounting information even after service termination for periods of time according to the local laws.

4 State of the Art Solutions for the Descriptors

This section presents an analysis and comparison of the current methods/languages to describe the information a network descriptor should.

4.1 ETSI Network Service Descriptor (NSD)

ETSI has defined descriptors for NFV technologies in ETSI NFV MAN 001 [11] and IFA 013 [17]. Both documents provide a very detailed description of every information element present in the description of a NFV Network Service (NFV-NS), and the reader can identify the dependencies between them. The specifications give an exhaustive description of the different requirements that every NFV-NS can have as well as the properties that the Virtual Links (VL) and network functions have to meet. To satisfy variations in deployments of several services, the documents introduce the concept of deployment flavours; these are entities that can specify the number of instances, parameters to meet QoS, and monitoring parameters that can be related to the services deployed.

There is still a gap in how to deal with scenarios such as failure recovery, healing and scaling according to events that might be triggered during the Life Cycle Management (LCM).

4.2 TOSCA Network Function Virtualization (NFV)

Topology and Orchestration Specification for Cloud Applications (TOSCA) [34] is a data model standard used to orchestrate NFV services and applications. It helps to define topologies, dependencies and relationships between virtual applications (VA).

With the growing ecosystem for NFV and the need to model the relationships among VNFs, several organizations proposed languages to model the relationships. Thus. ETSI NFV defined standard fields and features that the languages have to satisfy. Correspondingly, the TOSCA community created the TOSCA NFV simple profile [35] to adapt these features and follow the standard version of ETSI. The work on this profile is still ongoing, extended with the new necessities the contributing companies are finding. Furthermore, several contributing companies are trying to extend the existing standards according their own needs and TOSCA is trying to introduce the extensions in the standard. The TOSCA NFV profile specifies a NFV specific data model using TOSCA language and following the standardized fields required by the information model of ETSI NFV.

The deployment and operational behavior requirements of each NFV-NS in NFV will be captured in a deployment template, stored during the NFV-NS on-boarding process in a catalogue, for future selection for instantiation. This profile using TOSCA as the deployment template in NFV, defines the NFV specific types to fulfill the NFV requirements.

TOSCA deals properly with the gap mentioned in ETSI NSD in the previous section and tries to specify what are the type of events that can trigger the life cycle management operations.

Two other relevant characteristics of TOSCA that are worth mentioning, due to the implication of the implementation of the VS, are the packaging and the notation used.

For the packaging, the TOSCA standard defines the Cloud Service Archive (CSAR) [36], which is a compressed file with a specific structure of files and folders. This file contains all the information required to describe the network service, and the scripts required by the lifecycle of the service. This standard is important as it is becoming one of the de-facto standards for service specification.

The notation used in most of the files within a CSAR packages follows the YAML [37] syntax. This notation has been widely adopted due to its simplicity and its capability to model complex relationships. In the 5G-TRANSFORMER project we aim to adopt both, the CSAR packaging and the YAML notations, to aid the on-boarding of new services.

4.3 Descriptors in H2020 SONATA project

The SONATA project [46] provides a tightly integrated Network Service SDK to design and debug network services and a Service Platform (SP) which manages the execution of network services. The information exchanged between the SDK and the SP relies in descriptors that convey all the information required for deploying and instantiating network services and their VNFs and setting up the networks. One of the key outcomes of the project is, therefore, the information model of these descriptors. In this sense the Sonata project proposes a new information model based on the functional model proposed in ETSI NFV IFA 007 [47].

[48] describes details of the information model and in particular all the proposed extensions in the following items:

- NFV specific entities: VNFD, NSD, etc.
- NFV hosting relationships
- NFV composition relationships

The idea within the 5G-TRANSFORMER project is to analyze how to adapt or extend this information model using the requirements we identify, see Section 6, and the examples for all the descriptors that we have in Section 14. Furthermore, the SONATA SDK and the SP use CSAR packages and YAML descriptors which we also aim to use within the 5G-TRANSFORMER project.

5 5GT-VS Architecture

We describe the architecture of the Vertical Slicer (5GT-VS), as well as its main functional blocks and the algorithms therein. We provide an overview of the 5GT-VS in Section 5.1, we detail its architecture in Section 5.2 and its main components in Sections 5.3, 5.4, 5.5, and 5.6. Eventually, we describe the northbound and southbound interfaces of the 5GT-VS in Sections 5.7 and 5.8. Note, the southbound interface of the 5GT-VS is the northbound interface of the 5GT-SO.

5.1 Vertical Slicer Overview

The 5GT-VS is the common entry point for all verticals into the 5G-TRANSFORMER system, being part of the OSS/BSS of the administrative domain of a 5G-TRANSFORMER service provider (TSP). The 5GT-VS coordinates and arbitrates the requests for vertical services. Vertical services are offered through a high-level interface focusing on the service logic and the needs of vertical services. It allows defining vertical services from a set of vertical-oriented service blueprints, which, along with instantiation parameters, will result in Vertical Service Descriptors (VSD). Then, the 5GT-VS maps the vertical service descriptions and requirements defined in the VSD onto a network slice. We describe network slices with extended ETSI NFV Network Service Descriptors (NSD) [38], which define forwarding graphs composed of a set of Virtual Network Functions (VNF) or Virtual Applications (VAs) connected with Virtual Links (VL), where some of them have the specific characteristics and constraints of MEC applications. Importantly, the 5GT-VS allows mapping several vertical service instances to one network slice, handling vertical-dependent sharing criteria and taking care of the necessary capacity changes in pre-established slices. In conclusion, the most fundamental tasks of the 5GT-VS are to provide the functionality for creating the vertical service descriptions, and to manage the lifecycle and the monitoring of vertical service instances and of the corresponding network slice instances.

In addition to such tasks, the 5GT-VS provides arbitration among several vertical service instances in case of resource shortage in the underlying infrastructure and based on global budgets for resource utilization, as defined in the SLAs. The arbitration component in the 5GT-VS maps priorities of services and SLA requirements to ranges of cardinalities of resources. These cardinalities are used by the 5GT-SO while deploying the NFV network services (NFV-NS) and, in case of actual resource shortage, to assign resources to the most important vertical service instances.

5.2 Vertical Slicer Architecture

The architecture of the 5GT-VS is shown in Figure 3. As mentioned, the 5GT-VS is part of the provider's OSS/BSS, and interacts with the vertical (or the M(V)NO) through its northbound interface (NBI) and with the service orchestrator through its southbound interface (SBI). The 5GT-VS interacts as well with the OSS/BSS Management Platform of the TSP. This in turn interacts with the TSP, but this interaction among management platform and TSP is out of scope.

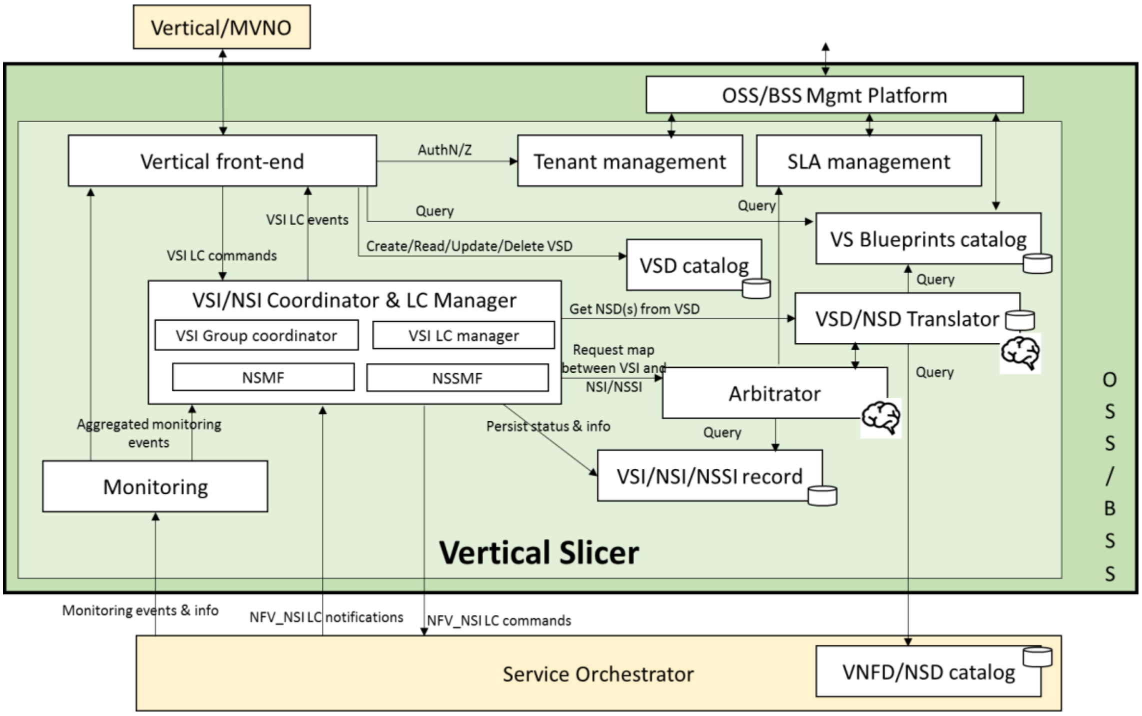


FIGURE 3: THE VERTICAL SLICER ARCHITECTURE

Looking at Figure 3, at the top we find the Vertical Front-end, which provides an entry point to receive requests from the verticals/MVNO about the service management and monitoring. Importantly, the Vertical Front-end presents the vertical with Vertical Service Blueprints (VSB), see Section 6.1, stored and handled through the related catalogue, and supports the vertical in providing the parameters therein to obtain the corresponding Vertical Service Descriptor (VSD), see Section 6.2.

The lifecycle (LC) of Vertical Service Instances (VSIs) and the corresponding Network Slice Instances (NSIs) are handled through the component called “VSI/NSI Coordinator & LC Manager”. This component, described in Section 5.6, implements the central engine of the 5GT-VS: it manages the association between VSIs and NSIs, regulating the sharing of network slices among different vertical services and their decomposition in network slice subnets. Moreover, it also handles the finite state machines for VSIs and NSIs lifecycle, coordinating commands and events associated to them. The network slice management is handled through requests for instantiation, modification and termination of the corresponding NFV-NSs, interacting with the 5GT-SO. The status and the current characteristics of VSIs and NSIs are stored persistently in the VSI/NSI records.

While the “VSI/NSI Coordinator & LC Manager” manages the lifecycle of slices and services, the complete 5GT-VS decision logic is implemented in the VSD/NSD Translator and in the Arbitrator. The VSD/NSD Translator, described in Section 5.3, selects the descriptors of the NFV network services that are able to support the requested vertical services. In particular, the VSD/NSD Translator identifies the network service deployment flavour (DF) (e.g., number of VNF instances, amount of vCPU or RAM for VNFs, bandwidth of VLs) most suitable to guarantee the performance and the characteristics defined in the VSD. The Arbitrator, described in Section 5.4, makes decisions about the sharing of network slices among different vertical services and the scaling of vertical services based on service priority and resource budget in verticals' SLAs.

Finally, the 5GT-VS Monitoring Service, described in Section 5.5, interacts with the 5GT-SO to collect monitoring data about the established NFV network services and correlates or aggregates these data in order to produce metrics and KPIs for network slices and vertical services. These metrics can be used as input for SLA verification or to make decisions about the lifecycle of a network slice, for example triggering a scale up action in case of decreasing performance.

The 5GT-VS NBI, detailed in Section 5.7, is based on a REST API and implements the Ve-Vs reference point between the 5GT-VS and the vertical/MVNO, as well as the Mgt-Vs reference point between the 5GT-VS and the OSS/BSS Management Platform (see Figure 4) In particular, the Ve-Vs reference point provides mechanisms for VS blueprint (VSB) queries and operation (e.g. instantiation, termination, queries and update) or monitoring (e.g. queries and subscriptions-notifications) of vertical services. The Mgt-Vs reference point provides management primitives, related to tenants, SLAs and VSBs.

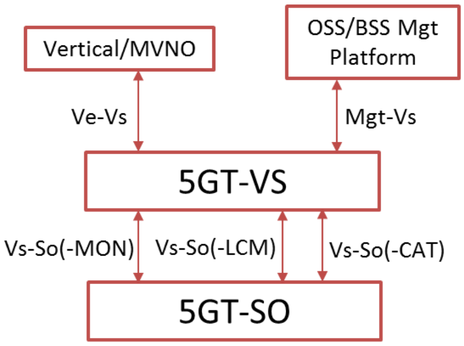


FIGURE 4: VERTICAL SLICER REFERENCE POINTS

The 5GT-VS SBI, and thereby the 5GT-SO NBI, described in Section 5.8, allows the interaction with the Service Orchestrator (5GT-SO). Its definition is based on the ETSI NFV IFA 13 [17], which defines the NBI of an NFVO. The 5GT-VS SBI implements the following reference points:

- Vs-So(-LCM), for the operation of network services. It offers primitives to instantiate, terminate, query, update and re-configure network services or receive notifications about their lifecycle;
- Vs-So(-MON), for the monitoring of network services and VNFs through queries or subscriptions and notifications about performance metrics and failures;
- Vs-So(-CAT), for the management of NFV NSDs and VNF package descriptors and storing them in catalogues, including mechanisms for their on-boarding, removal, updates and queries.

5.3 The VSD/NSD Translator Module

The VSD/NSD Translator module is an entity within the 5GT-VS that maps the vertical's requirements into technical specifications needed by the 5GT-SO to perform a NFV-NS deployment.

To define a vertical service or prepare it for deployment, the vertical first selects a VSB, which contains parts that have to be filled in with the requirements it demands. Once the vertical provided actual values for the missing parts in the VSB, it becomes a Vertical Service Descriptor (VSD): a high-level specification of a network service, based on a service logic perspective rather than a resource-based perspective.

The VSD is then mapped into a Network Service Descriptor (NSD) through the following steps: (1) the Translator queries the Virtual Network Function Descriptor (VNFD) catalogue for the VNFs referenced in the VSD; (2) the Translator queries the NSD catalogue for the NSDs referenced in the VSD; (3) the Translator fills the fields present in the deployment flavour of each VNFD and of each Virtual Link Descriptor (VLD), using the QoS restrictions of the VSD (such fields will be finalized by the Arbitrator later on).

A relevant example of VS is a video streaming service. The VSB contains fields for service type, number of users, and availability, which have been completed by the vertical in this example as follows:

- Service type: video streaming.
- No_users: 10.000.
- Time_availability: 24 h/day - 7days /week.

Upon receiving the VSD, the Translator looks for the VNFDs of firewall, load balancer and streaming servers. Such VNFDs are referenced in the NSD and connected with the VLs composing the VNFFG shown in Figure 5. Some of the VLs and VNFs are backup components to fulfil the availability requirement, while the number of streaming VNF instances is set to a suitable value to satisfy both availability and number of users to serve. With a less stringent availability requirement, the Translator would choose a VNFFG without redundant components and paths. The NSD also specifies the initial values of VNF instances and VLs deployment flavours, as well as monitoring and healing fields indicating the start and reconnect mode for VNFs, in the case some of them go down. Then this NSD is given as input to the Arbitrator, in order to finalize the CPU and bandwidth requirements specified in the aforementioned deployment flavours.

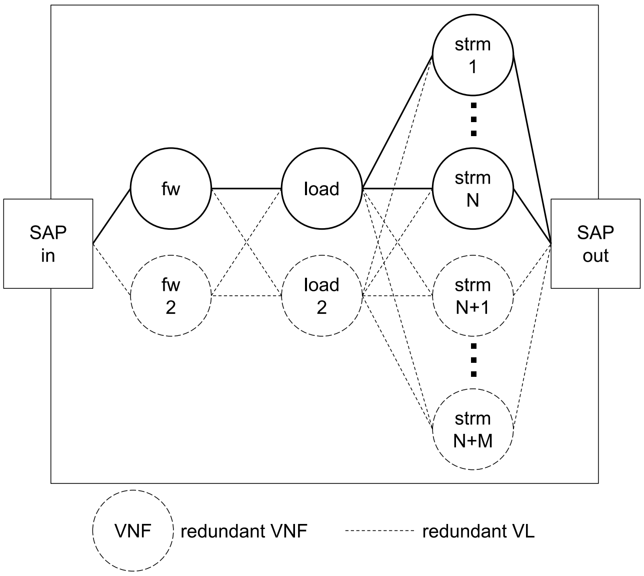


FIGURE 5: VIDEO STREAMING VNFFG

5.4 The Arbitrator Module

SLA management is a key aspect in the provisioning of services to a vertical on top of 5G networks. Any degradation of the SLA negotiated between the provider's OSS/BSS and the vertical can impact not only on the technical behaviour of the vertical service but also on the reputation or business leadership of the vertical itself. Finally, legal consequences could occur depending on the nature of the service being provided. SLAs are more than just sales agreements, they are part of a legally binding contract.

The deployment of a vertical service is a multi-dimensional problem, where various objectives, resource types (e.g., CPU, memory, storage) and autonomous infrastructure providers can be involved. Thus, there will be interdependencies among the different resources to be provisioned and configured.

The vertical, as 5G-TRANSFORMER service consumer, will specify some Service Level Objectives (SLOs) adapted to its service needs. Examples of SLOs can be, e.g., a maximum end-to-end latency of 20 ms for the Intersection Collision Avoidance (ICA) use case of an automotive vertical. Vice versa, the provider's OSS/BSS can state different service level classes, representing distinct guarantees on resource availability. The matching between the SLOs desired by the vertical and the service level classes offered by a provider will define an SLA. The contents and terms of an SLA will be inevitably used as specific directives for configuration and orchestration of services and resources in provisioned network slices. A slice life-cycle operation is effective if an acceptable trade-off is found between satisfying the terms in its SLA and satisfying the rest of the service requests in the system. It is here where arbitration mechanisms are required for managing such trade-offs.

The Arbitrator within the 5GT-VS is the entity where these mechanisms are implemented and that provides the 5GT-SO with support for service deployment. It operates based on the SLA, as well as the information on each service provided by the vertical, namely: (1) the service priority level, (2) the VNFFG representing the service, (3) the relative CPU requirements of the VNFs in the VNFFG, as well as their memory/storage requirements, and (4) the vertical's quality of service (QoS) requirements (e.g., end-to-end latency, service availability or reliability level). Note that such information is described in the NSD created by the VSD/NSD Translator for the received VSD.

The Arbitrator's tasks are twofold:

1. To decide how to map new vertical services in NSIs, allowing multiple vertical services to share one or more NSIs or NSSIs.
2. To determine the deployment flavours associated with each service, such that the vertical's QoS requirements are met, while accounting for the services priority level. Note that, upon the request of a new service instance by a vertical, the Arbitrator may need also to update the deployment flavours associated with previously allocated service instances.

These two tasks are detailed in the following subsections for the case of services requested by the same verticals (the case of services by different verticals will be addressed in future versions of the Arbitrator's logic).

5.4.1 Sharing of network slices among vertical services

During the instantiation of a vertical service, the Arbitrator is responsible for making decisions about the sharing of NSIs and NSSIs among different vertical service instances. A typical example could be the sharing of the same vEPC among different vertical service instances with similar requirements in terms of mobile access, or the sharing of a service component for the collection of vehicle messages among multiple automotive services that make use of the same messages.

The decision about slice sharing is made by the Arbitrator analysing the pair <NSD; DF> initially computed by the VSD/NSD Translator, which describes the structure, the characteristics and the cardinalities of a potential NSI able to support the requested VSI. Starting from the NSD, the Arbitrator verifies if one or more existing NSIs can be re-used to accommodate the new VSI or at least part of it, e.g. one or more nested NFV-NS that may be included in the NSD. Obviously, the decision is also affected by the constraints specified by the vertical in the VSD, in particular about the possibility to share (part of) the service components with existing ones already instantiated for the same or different verticals.

If no suitable NSIs are already available, the Arbitrator simply decides to create a NSI, based on the NSD originally computed by the VSD/NSD Translator. It will then proceed with the update of the deployment flavour initially set by the VSD/NSD Translator, as necessary (see Section 5.4.2). The “VSI/NSI Coordinator & LC Manager” is then responsible for managing the actual instantiation of the new network slice and its slice subnets, triggering the 5GT-SO to deploy the required NFV-NSs.

On the other hand, if the Arbitrator finds existing NSIs that can be shared, it must determine if and how they should be modified to accommodate the additional service. For this, it interacts with the VSD/NSD Translator, providing as input the VSDs of all the VSIs that will share resources from the given set of network slices. The VSD/NSD Translator should then combine the requirements from the various VSDs and return a suitable set of pairs <NSD; DF> defining the characteristics and the size of the target NSIs. Assuming that the elaborated solution is compliant with the vertical’s SLAs, the Arbitrator identifies the NSIs to be modified and the ones to be created entirely, according to the result provided by the VSD/NSD Translator. As in the previous case, it is up to the “VSI/NSI Coordinator & LC Manager” to manage the whole set of procedures to execute the required actions.

5.4.2 Computation of deployment flavours

As far as the deployment flavours are concerned, let us assume for now that, to support the requested vertical services, new NSIs should be deployed. We propose the following algorithm for the Arbitrator. Let us denote by C, B, M and S the total amount of, respectively, CPU, bandwidth, memory, and storage that the vertical’s services can use per SLA. First, the Arbitrator orders the service instances that the vertical wants to deploy, or it has deployed, from the highest priority level to the lowest. It then considers the highest-priority service instance, say, s, and allocates memory and storage based on the needs exhibited by the VNFs in the VNFFG representing s. A more complicated procedure is instead required for CPU and bandwidth allocation, since this depends on the VNF placement decisions made by the 5GT-SO. For clarity of presentation, let us focus on the service latency as the main performance metric, and denote by D_s the maximum latency that service s is allowed to experience. Two components contribute to the service latency: (i) the processing time, due to the execution of the VNFs in the VNFFG, and (ii) the network travel time, which is due to the time needed to transfer data from one VNF to the next in the VNFFG, when adjacent VNFs are deployed on different servers. While the former depends on the CPU allocated to the VNFs execution, the latter depends on the deployment decisions made by the 5GT-SO, and on the bandwidth associated with the VLs connecting the servers when the 5GT-SO places VNFs on different servers.

The Arbitrator considers the best and worst possible cases that may occur depending on the 5GT-SO later decisions. The best case happens when all VNFs in the VNFFG, whose set is denoted by *V*, are deployed within the same server. In this case, the bandwidth required for data transfers over VLs for service *s*, β^b, can be set to zero, while the allocated CPU, μ^b, can be computed so that:

$$\sum_{v\in V}\frac{1}{f_v\mu^b-\lambda_v}\le D_s \quad \text{and} \quad \mu^b\le C$$

where, for simplicity, we neglected the delay due to memory/storage access. In the above equation, f_v is the relative computational requirement of VNF *v*, such that; $\sum_{v\in V} f_v = 1$. Moreover,

- the first inequality imposes that the latency experienced by the service does not exceed the maximum value. The total latency is given by the sum of the latency contributions due to all VNFs in the VNFFG. Each contribution is computed by modeling the generic VNF v as a FIFO M/M/1 queue with $f_v\mu^b$ as the output rate and λ_v as the service request rate input to v;
- the second inequality imposes that the CPU allocation does not exceed the vertical's amount of CPU, *C*, that is available to the vertical.

Conversely, in the worst case, each VNF in *V* is deployed in a different server, thus implying the need for bandwidth allocation. Specifically, the allocated CPU, μ^w, and bandwidth, β^w, now should satisfy the following constraints:

$$\sum_{v\in V}\frac{1}{f_v\mu^w-\lambda_v}+\sum_{(u,v)\in E}\frac{d_{v,w}}{f_{u,v}\beta^w}\le D_s$$

$$\mu^w\le C\;;\;\;\beta^w\le B$$

where $f_{(u,v)}$ is the relative bandwidth requirement for the VL connecting VNFs $_u$ and $_v$. Also,

- the first inequality accounts for the latency due to both the VNF execution and the travel time over the VLs connecting any two adjacent VNFs (*E* denotes the set of edges in the VNFFG);
- the second and third imposes that both the total CPU and bandwidth allocations do not exceed the corresponding budget available to the vertical.

Once the Arbitrator has determined the values (μ^b,0) and (μ^w, β^w) for the tagged service, it proceeds with the second service in the list, following the same steps as above but using the remaining amount of resources available to the vertical in terms of CPU (*C*) and bandwidth (*B*).

Note that, using the above values for the CPU and bandwidth that should be allocated for a service, the Arbitrator can determine the corresponding per-VNF and per-VL values based on the relative “weight” that, respectively, VNFs and VLs have in the NSD initial setting. The Arbitrator will thus update the deployment flavour in the VLDs and the VNFDs of the NSD created by the VSD/NSD Translator. Specifically, the Arbitrator sets the instantiation-level information element in the VNFDFs, see Section 15.10, by using the worst-case values as default deployment flavour and the best-case values as optional. The updated NSD is then returned to the VSI/NSI Coordinator & LC Manager, which will send it to the 5GT-SO. After the 5GT-SO has made its deployment decisions, it will notify the 5GT-VS about the current resource allocation (e.g., in terms of characteristics of the instantiated VNFs) so that the Arbitrator can compute the new amount of resources that are available to the vertical.

Let us now consider the case where the vertical service to be processed can be supported on an existing NSI, or, using an existing NSSI. In this case, the Arbitrator should add the traffic load due to the newly requested vertical service to the load of the existing VNFs/VLs, and re-compute the necessary CPU and bandwidth, as described above. Again, the Arbitrator will use the CPU and bandwidth values obtained for the worst and the best case, to update the deployment flavour of the involved VLDs and VNFDs.

It is clear that, in case of resource shortage, some, lower-priority services, may not be accommodated, or may be terminated due to the need to re-allocate resources to higher priority services.

To summarize, the Arbitrator module allows the 5GT-SO to make deployment decisions meeting the vertical’s indications even if (i) it is unaware of higher-layer information like the SLA between the vertical and OSS/BSS, and (ii) the total amount of available resources is not sufficient to adequately deploy all requested services.

5.5 The Monitoring Service

In the 5G-TRANSFORMER framework, each architectural component (i.e. 5GT-VS, 5GT-SO, 5GT-MTP) includes a monitoring service able to provide performance metrics and failure reports targeting the logical entities managed by each component, see Section 2.4. The monitoring service at each architectural layer operates on specific resources; in particular, the 5GT-VS Monitoring Service is responsible for producing monitoring data about network slices and vertical services and for elaborating more elementary monitoring data about VNFs and NFV network services, as retrieved from the underlying 5GT-SO Monitoring Service. Figure 6 shows the different interfaces of the 5GT-VS Monitoring Service. A description of service monitoring for the complete 5G-TRANSFORMER system as well as monitoring requirements per use case is presented in [4].

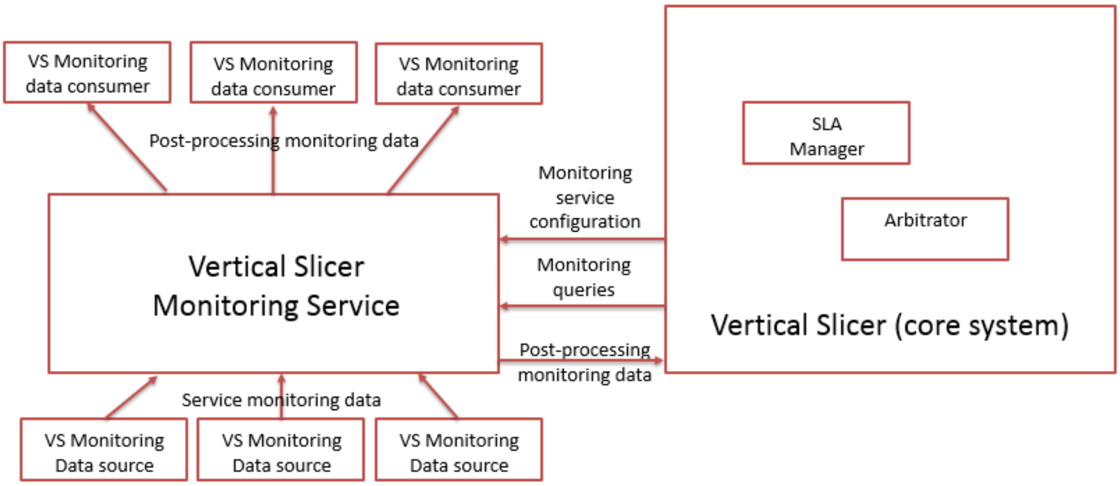


FIGURE 6: 5GT-VS MONITORING INTERFACES

The 5GT-VS Monitoring Service provides monitoring data of VSIs as input for internal decisions for SLA management at the Arbitrator. The 5GT-VS provides also monitoring data about the deployed VSIs to the verticals, to feed the internal processing of vertical applications, where needed. According to ETSI GS NFV-IFA013 [17], an NFVO exposes interfaces towards the OSS/BSS. The NFV-NS Performance Management interface and NFV-NS Fault Management interface allow the 5GT-VS to get monitoring data from the 5GT-SO, see Section 5.8 for further details on the 5GT-SO NBI. In Figure 6 the 5GT-SO is depicted abstractly as VS Monitoring Data source.

The 5GT-VS monitoring service should be able to expose the monitoring information it receives from the 5GT-SO Monitoring Service to the VS Front-end. Actually, this exposure can be part of the SLA. Thus, a field about monitoring, in addition to the one on SLA, has to exist in the VSB and VSD. This field allows a vertical specifying which monitoring data it to receive (see Table 11).

The monitoring data about VSIs provided at the 5GT-VS NBI should be aggregated and/or abstracted. 3GPP [13] defines 2 types - performance measurement and alarm - of data aggregation for network slices. Within each type there are two categories of data: network related and vertical service related. The 5GT-VS Monitoring Service receives and shares monitoring data from the 5GT-SO via the 5GT-SO-MON interface or generated internally. The VSI LC Manager and/or NS(S)MF send a request to the 5GT-VS Monitoring Service indicating the monitoring metrics to subscribe to. The 5GT-VS Monitoring Service relays the request to the 5GT-SO Monitoring Service. During service runtime, the 5GT-SO indicates the availability of monitoring data to the 5GT-VS Monitoring Service. The 5GT-VS Monitoring Service retrieves the data and informs the VSI LC Manager, NS(S)MF, and/or Arbitrator. It can also send this information directly to the 5GT-VS Front-end to provide it to the vertical, depending on the SLA and what was defined in the VSD.

The 5GT-VS Monitoring Service can supervise alarms related to an NSI and/or VSI only if it has subscribed to them. Thus, the NS(S)MF sends a corresponding request for subscription to NSI alarm notifications towards the 5GT-VS Monitoring Service, in order to receive the alarm notifications related to the NSI. As a result of this operation, whenever an alarm is raised, a notification will be sent to the 5GT-VS Monitoring Service, which will transmit it to the NS(S)MF.

If the 5GT-VS Monitoring Service has subscribed to performance information (if asked by the VSI LC Manager and/or NS(S)MF), then it will receive a notification when new collected performance data are available, After the notification, according to the monitoring data, it will relay it to the VSI LC Manager and/or NS(S)MF. To this end, the 5GT-VS Monitoring Service has to keep, e.g., the history of all the requests.

The NS(S)MF can request the 5GT-VS Monitoring Service to create (and delete) a threshold on a specified performance metric (for NFV-NS(s)) for which notifications will be generated when crossed. The VSI LC Manager can do the same for vertical service(s).

5.6 VSI/NSI Coordinator & LC Manager

This component coordinates the activities of the other 5GT-VS components. To do so, it uses several subcomponents and interacts with the other components and databases of the 5GT-VS. The subcomponents are the VSI Group Coordinator to handle the VSIs of one vertical, the VSI LC Manager to handle the lifecycle of single VSIs, and the NSMF and NSSMF to handle the lifecycle of NSIs and NSSIs.

5.6.1 VSI Group Coordinator

A vertical may define one or several resource budgets for its VSIs. These resource budgets define the maximum amount of e.g. storage, vCPU, memory, and bandwidth. The resource consumption of a specific VSI is checked against one such resource budget. This component maintains the relation between resource budgets and VSIs for each individual vertical. The actual checking and arbitration among different VSIs is done by the Arbitrator (see Section 5.4).

5.6.2 VSI LC Manager

The VSI LC Manager proposes vertical services to customers via the catalogue of VSBs, which may include also information on the services’ cost. Internally, the VSI LC Manager keeps a record of the requested VSIs and maintains their references to the associated supporting NSIs. It also performs accounting of vertical service resource consumption per NSI.

The 5GT-VS SBI is used to link the NSMF in the 5GT-VS with the 5GT-SO, e.g. for network slice management: discovery, allocation and LCM of NSIs, performance monitoring, fault management and accounting. Due to the monitoring and the fault management of NSIs, the VSI LC Manager is aware of the running state of each NSI and able to verify whether it matches the expected SLAs of the corresponding VSIs. If the outcome indicates that a vertical service SLA has not been fulfilled, remediation is required.

In addition, the VSI LC Manager supports life cycle management operations, pricing and charging, performance monitoring and fault reporting of VSIs. If a customer changes the requirements of a VSI, the VSI LC Manager triggers the Arbitrator to update the network slice requirements via recalculation of the NSD deployment flavour. Eventually, the VSI LC Manager asks the NSMF for a modification of the NSI capacities.

As mentioned, the VSB catalogue includes information on SLA costs. To enable the charging of the verticals for the VSIs, the VSI LC Manager has to trace their resource consumption according to the information (e.g., monitoring data, events) reported by the 5GT-SO to the 5GT-VS Monitoring Service. Thus, the VSI LC Manager subscribes to alarm notifications, using the 5GT-VS Monitoring Service. Some service related alarm data may be provided if required by the verticals. Also, the VSI LC Manager subscribes to performance measurement notifications. Again, some service related performance measurement data may be provided to the verticals according to their SLAs.

5.6.3 NSMF and NSSMF

The NSMF manages NSIs; correspondingly the NSSMF manages NSSIs. As mentioned, the VSI LC Manager relies on the NSMF and NSSMF to assess the feasibility of providing NSIs and NSSIs, resp. As the functionalities of NSMF and NSSMF are rather similar, we describe in the following the NSMF functionality only. The NSMF and NSSMF are based on [13], (see also Section 11.1.2).

Specifically, the NSMF checks which PNFs and/or VNFs are referenced in the NSDs (the NSDs are exposed by the 5GT-SO to the 5GT-VS via the NSD catalogue), then the NSMF provides the VSI LC Manager with the NSIs, to which a VSI could be deployed. The NSMF also keeps records of all the network slice requirements (e.g. number of CPU, storage) per NSI. This information can be used by the Arbitrator to recalculate the deployment flavours of the NFV-NSs. However, the NSMF is not aware of the VSIs but only of the requirements received from the Arbitrator. Due to the monitoring and the fault management of NFV-NS instances, the NSMF is aware of the running state of the latter and able to verify the expected functionalities and whether its SLAs are met or not. In case of SLA violations the NSMF can trigger corresponding alarms.

5.7 Vertical Slicer NBI

Two reference points are defined at the northbound of the 5GT-VS (see Figure 7):

- **Ve-Vs**, between a vertical and the 5GT-VS. This reference point provides the mechanisms to allow the vertical to retrieve VSBs, to manage VSDs, to request operational actions on VSIs, like instantiation, termination, modification, and to monitor performance and failures of instantiated vertical services.
- **Mgt-Vs**, between the OSS/BSS Management Platform and the 5GT-VS. This reference point provides primitives to manage tenants, SLAs and VSBs. It is used mainly for management and administrative issues and it is handled internally within the 5G-TRANSFORMER service provider.

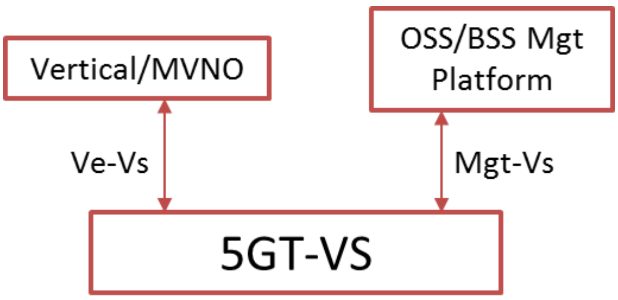


FIGURE 7: REFERENCE POINTS ON THE NORTHBOUND OF THE 5GT-VS

The 5GT-VS NBI implementing both reference points is a REST API based on the HTTP protocol and JSON messages, where the 5GT-VS acts as REST server and the verticals and the OSS/BSS Management Platform act as REST clients. The interface should also support asynchronous notifications from the 5GT-VS, for example based on web sockets. Suitable mechanisms for authentication and authorization of the entities issuing the requests should also be supported. However, this section specifies only the primitives and the content of the abstract messages exchanged at the reference points, while their encoding into specific protocol messages is left for the following implementation phase.

Moreover, it should be noted that the 5GT-VS NBI specified in this section focuses only on the provisioning of Vertical Services, while the Network Slice as a Service (NSaaS) and the NFVI as a Service cases (NFVIaaS) (see section 5.9) are left for further studies and will be addressed in the following version of the 5GT-VS architecture.

5.7.1 Ve-Vs reference point

The Ve-Vs reference point identifies the following operations:

- Query VSBs.
- Create, query, update, and delete VSDs.
- Instantiate, query, terminate, modify VSIs.
- Notifications about vertical service lifecycle events.
- Query monitoring parameters for VSIs.
- Subscriptions/notifications about vertical service monitoring parameters.
- Notifications about vertical service failures.

We provide a description of the operations including their parameters in an annex in Section 14.1. As an example of how these descriptions look like, see the Query VS blueprints operation in Section 5.7.1.1.

5.7.1.1 Query VS blueprints

This operation allows a vertical to retrieve one or more VSBs from the 5GT-VS catalogue. The blueprints are then used by the vertical to create the VSDs for the vertical services to be instantiated.

The Query VS blueprints messages are specified in Table 9.

TABLE 9: QUERY VS BLUEPRINTS MESSAGES

Message	Direction	Description	Parameters
Query VS blueprint request	Vertical → 5GT-VS	Request to retrieve one or more VSBs matching the given filter.	• Filter (e.g. VSB ID, …) Vertical ID.
Query VS blueprint response	5GT-VS → Vertical	Response including the details of the requested VSBs.	• List<VSB> (format specified in Section 6.1).

5.7.2 Mgt-Vs reference point

The Mgt-Vs reference point between the OSS/BSS Management Platform (Mgt in the following) and the 5GT-VS identifies the following operations:

- Create, query and delete tenants.

- Create, query, modify and delete SLAs.
- Create, query and delete VSBs.

We provide a description of the operations including their parameters in an annex in Section 14.2.

Beyond these operations, the OSS/BSS Management Platform has also access in read mode to the information related to all the entities managed by the 5GT-VS, i.e. VSDs, VSIs, NSIs and NSSIs, which can be retrieved from the related catalogues and records.

5.8 Service Orchestrator NBI

The 5GT-SO NBI refers to the interface between the 5GT-VS and the 5GT-SO. It is based on the ETSI NFV IFA 013 interface (reference point *Os-Ma-nfvo*) [17]. This interface is used for: (i) network service lifecycle management; (ii) VNF package management; and (iii) forwarding of NFV-NSI-related state information.

The interactions between the 5GT-VS and 5GT-SO implement three reference points (see Figure 4 in Section 5.2):

- **Vs-So (-LCM)** is used for operation on NFV-NSs. It offers primitives to select, allocate, terminate, query, and reconfigure network services or receive notification about their lifecycle. In addition, this interface should provide pricing primitive for the provided services.
- **Vs-So (-MON)** is used for the monitoring of NFV-NSs and VNFs through queries or subscriptions and notifications about performance metric. Additionally, this interface provides APIs for the fault management.
- **Vs-So (-CAT)** is used for the management of NFV NSD and VNF package descriptors, including the onboarding, removal, updates and queries.

The 5GT-SO NBI supports several interfaces such as NSD Management, Network Service Lifecycle Management (e.g. on-boarding, enable, disable, query, update NSD and PNFD), and Lifecycle Change Notification (subscription, notification) and VNF Package Management interface, among others see clauses 5.2, 5.3 and 7 of ETSI NFV IFA 013 [17] for further detail). However, the 5GT-SO NBI implements further methods to allow the 5GT-SO to handle NFV-NSs extended to include also MEC Applications. In terms of interfaces, this has an impact mostly on the management of the catalogues and on the queries of the NFV-NS instances, where both of them should support the information related to the MEC applications. In particular:

- The Vs-So(-LCM) reference point is implemented through the IFA 013 NS Lifecycle Management Interface [17]. This interface provides messages for creation and deletion of network service identifiers as well as instantiation, scaling, update, querying and termination of a network service instance;
- The Vs-So(-MON) reference point is implemented through the IFA 013 NS Performance Management Interface [17] and the IFA 013 NS Fault Management Interface [17]. The Performance Management Interface provides messages for creating, deleting and querying Performance Monitoring jobs (PM jobs) as well as subscriptions and notifications to receive monitoring reports. The Fault Management Interface provides subscriptions, notifications and queries for about alarms related to network services and VNFs;

- The Vs-So(-CAT) reference point is implemented through the IFA 013 NSD Management Interface [17], the IFA 013 VNF Package Management Interface [17] and the ETSI MEC 10-2 Application Package Management Interface [18], where the 5GT-SO implements the role of the Mobile Edge Orchestrator (MEO). Each of them provides mechanisms to on-board, enable, disable, update, delete and query NSDs (IFA 13 [17]), PNFDs (IFA 13 [17]), VNF Packages (IFA 13 [17]) and MEC Application Packages (MEC 10-2 [18]), as well as subscriptions and notifications to notify new on-boarding actions or changes in existing descriptors.

In ETSI NSD, some information elements are missing, thus it needs to be extended or be modified to fully support the services provided by the 5G-TRANSFORMER project. They are: (1) AppD, which follows the information model defined in ETSI MEC 10-2 [18]; (2) latency and location constraints. NSD and VNF packages follow the information models defined in the IFA 014 [38] and IFA 011 [39] respectively. In 5G-TRANSFORMER, the NSD is extended to include references to the AppDs, as shown in Figure 8 and Figure 9. See also Sections 6.3.1 and 6.3.2 for details on how this is integrated in NSDs.

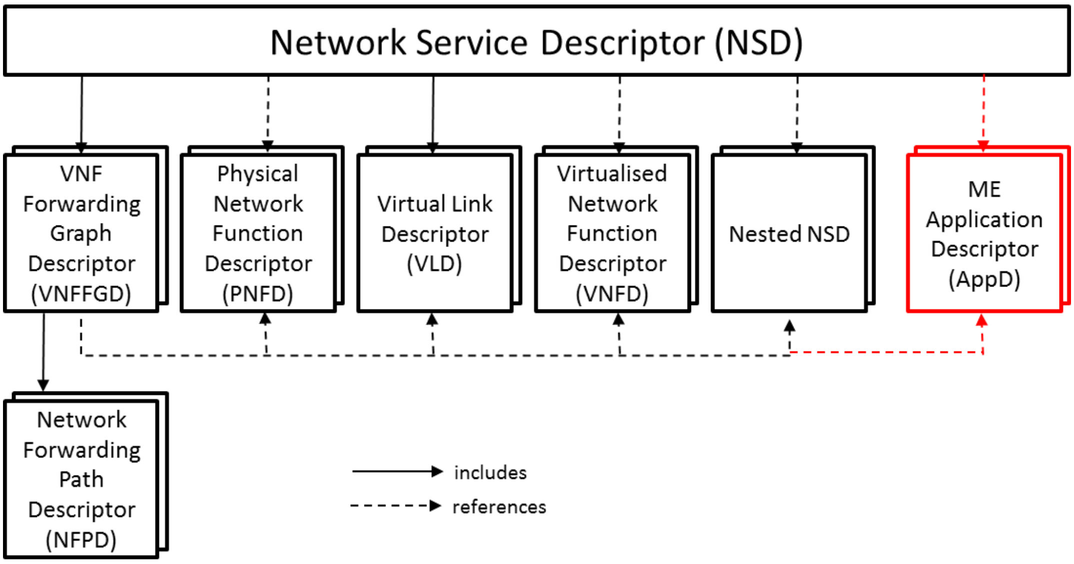


FIGURE 8: NSD EXTENDED WITH APPD REFERENCES

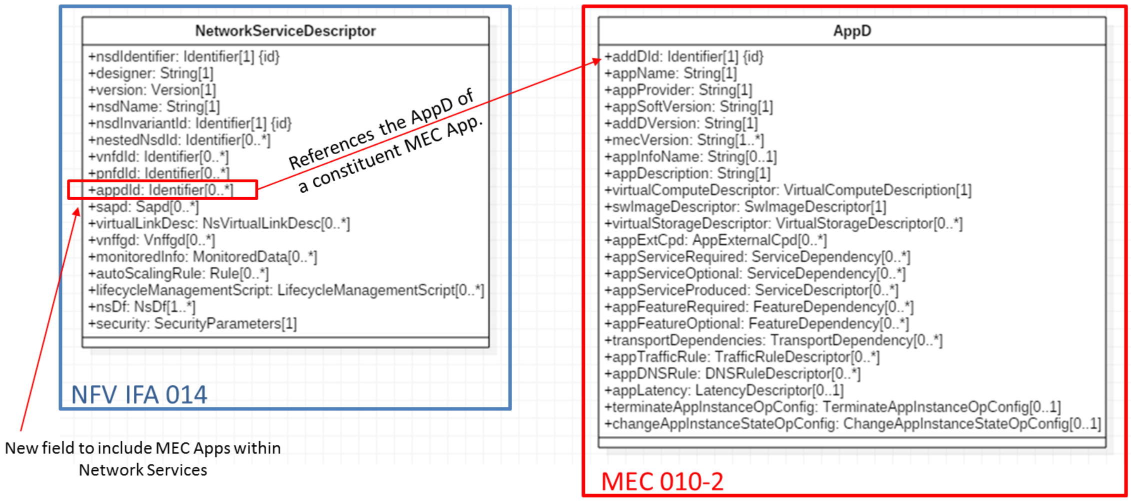


FIGURE 9: NSD AND APPD INFORMATION MODELS

Furthermore, ETSI NSD allows the integration of existing PNFs, i.e. machines deployed in a certain location, within an NSD. However, the ETSI NSD does not contain any information element to specify this location. Thus, the 5G-TRANSFORMER proposes corresponding extensions of information elements within NSDs to express location as well as latency constraints (see Section 6.3.3).

5.9 NFVIaaS and NSaaS Support

The 5GT-VS as defined before focused on the support for verticals to define and deploy vertical services, i.e. on Service as a Service (SaaS). We investigate two approaches to provide the support for the various levels of NFVIaaS and for NSaaS as defined below. In the first approach the support would be provided more directly, whereas in the second the support is provided indirectly by considering NFVIaaS and NSaaS as specific vertical services themselves.

Network slice as a service (NSaaS): Unlike vertical services, a NSaaS is to provide a network along with the services that it may support. For instance, a network provider may provide a mIoT network slice as a service, which may support several services, including sensor monitoring, collision avoidance as well as traffic management, and warehouse automation. The NSaaS customer is aware of the NSI; it can, in turn, play the role of a provider itself, and offer to its own consumers its vertical services built on top of the services of the network slice (B2B2X). Regarding the mIoT NSaaS example, the consumer may fragment its mIoT NSI capabilities per criticality level and per user-defined area (such as factory premises, cities, and venues) in vertical services offered to its subscribers. The NSaaS consumer may have to negotiate with the NSaaS provider to limit the level of exposure of:

- NSI characteristics allowing the NSaaS customer building its own services, for instance, security functions are reachable according to a prior agreement,
- NSI management features, for instance, only commissioning/decommissioning operations on an NSI are allowed.

NFVI as a Service (NFVIaaS): this business type allows customers setting up their own applications and network functions over an infrastructure provider. For instance, an

MVNO may bring its own Home Subscriber Server (HSS) that needs to be connected with the other vEPC functions provided by the MNO. This business type can be implemented as follows: The 5GT-VS provides a vEPC blueprint, wherein a field describing the HSS is filled up by the customer where the virtual machine imagine for the HSS is provided as a parameter. A MVNO can create its specific vEPC by completing this blueprint with the location of the virtual machine image for its HSS.

The direct approach allows specific customers, e.g. an MVNO, to request directly infrastructure or a NSI and operate it itself. The direct approach was used in the NSaaS example, with explicit operations for NSI management operations at the 5GT-VS NBI.

The NFVIaaS example uses the indirect approach, the infrastructure is again described as a vertical service. The blueprint has a placeholder to provide the HSS, otherwise, this is a vertical service as, e.g., one for sensor monitoring.

The second approach fits nicely, by definition, into the proposed internal architecture of the 5GT-VS. We are investigating whether all the proposed NFVIaaS and NSaaS variants can be handled this way or whether there are drawbacks in turns of performance, providing less control to the verticals as expected, etc. The outcome of these investigations will be reflected in further iterations of the internal architecture and reported in future deliverables.

6 Service Descriptions

This section focuses on the descriptors used in the 5GT-VS as such. We focus on the details of the information that shall be conveyed in each type of descriptor and the notations used. Basic examples of descriptions are provided in an annex in Section 14. To facilitate the reading of this section, in Figure 10, we show the mapping between the interfaces and the descriptors used on top of the components already shown in Figure 3.

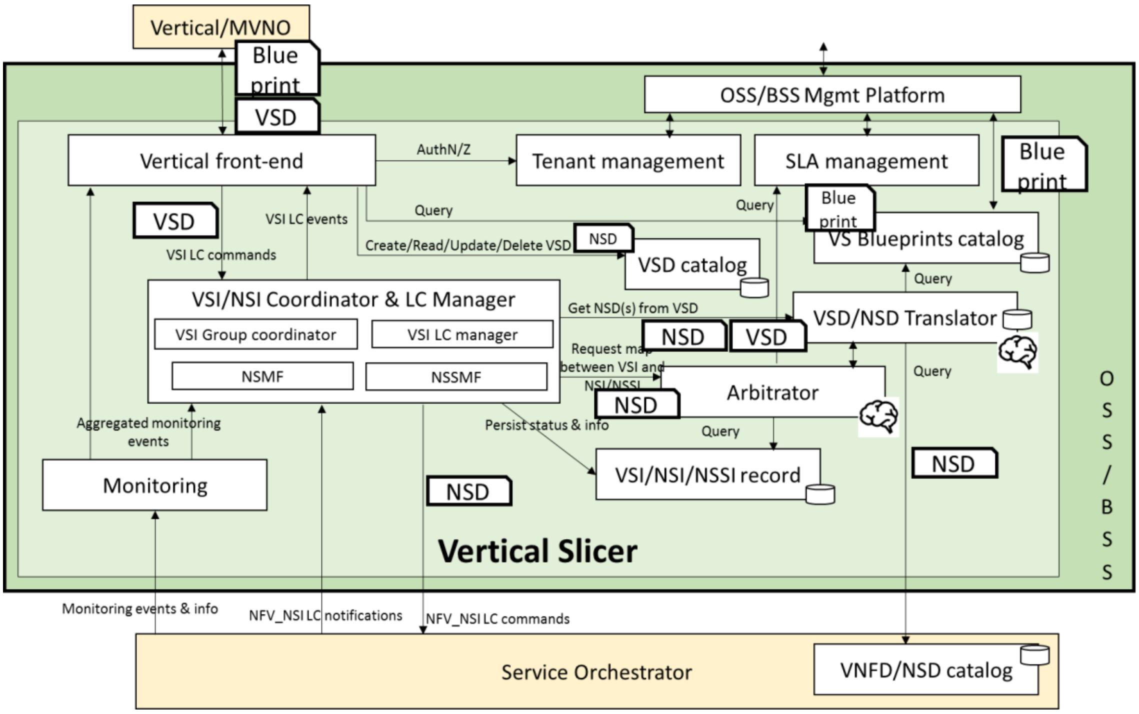


FIGURE 10: USAGE OF DESCRIPTORS IN 5GT-VS

The Vertical Service Blueprints (VSB) and Descriptors (VSD), which are main contributions of the project, enable modeling vertical services using parameters adapted to the knowledge and capabilities of the vertical (i.e. model services without detailing the infrastructure resources required). We use an extended version of ETSI NSDs [38] as a notation for network slice templates.

To instantiate a vertical service, a vertical first selects a blueprint, then it provides the missing information in the blueprint to prepare a VSD. The 5GT-VS maps this to a Network Service Descriptor (NSD), describing the network slice for this vertical service. In this section we start with presenting VSBs in Section 6.1, focusing on those fields that are different to VSDs. We consider VSBs as parameterized versions of VSDs, which we describe in Section 6.2. We describe NSDs and the extensions we propose in Section 6.3.

6.1 Vertical Service Blueprints

The 5GT-VS provides a list of VSBs to a vertical; designed to capture the service requirements specified by the vertical. Essentially, a VSB is a parameterized VSD, i.e. some aspects of a VSD are left open to be provided by a vertical.

A VSB, see Table 10, has fields to identify it and to describe the structure of the corresponding vertical service. In addition, it has fields to define SLA and other requirements. These fields are the same as those for VSDs and are shown in Table 11.

TABLE 10: VERTICAL SERVICE BLUEPRINT

Field	Description
Name	Name provided by the provider for the blueprint.
Description	Short description of the blueprint, e.g. “Sensor monitoring over 5G”.
Version	A version number.
Identity	Unique identifier for a blueprint. This is provided by the 5GT-VS when onboarding the blueprint.
Parameters	List of parameters. The list provides for each parameter its name, type, description, and the field where it is used.
Atomic functional components involved	List of atomic functional components (i.e., network functions and virtual applications in general) needed to implement the VSB requested, including dimensioning of those functions, in terms of number of users or sessions, bandwidth, parameters to be monitored, etc. The identified functional components can include functions to support the service from the provider perspective (e.g., firewall, load balancer, etc) transparent to the vertical. Specific information for the atomic components is described.
Service sequence	Description of how the atomic functional components should interact. This can be provided as a VNF forwarding graph [38]. It is also possible to refer to the service sequence of another VSD, thereby allowing to create nested VSDs.
Connectivity service	Specification of the kind of connectivity service to be provided to the vertical (e.g., L2VPN, Ethernet-VPN (EVPN), L3VPN, etc.), and properties (QoS and/or flow management, protection, restoration, etc.) that can be necessary for the service instantiation and orchestration.
External interconnection	An indication of specific external connections to be provided (i.e., Internet, Network A, etc.). This includes the specification of connection endpoints, and it may include the indication of specific identifiers of the external networks to connect (e.g., IP addresses, Autonomous Systems, etc.). Corresponding identifiers - if needed - can be retrieved from the connection endpoints after vertical service instantiation.
Internal interconnection	An indication of specific internal connections to be provided (i.e., connections to other vertical service instances or slices, from the same vertical customer or not).

It is left of the implementation of the 5GT-VS Frontend, see Figure 3, how blueprints are presented to verticals and what support is provided to verticals to provide the parameters when preparing a VSD. A wide range of parameters are possible, e.g. values for latency constraints, paths to virtual application images, types of connection services, types of traffic probes, etc., could be left open in the blueprint. It is up to the 5GT-VS Frontend whether it just offers free text fields, range limited fields, e.g. for latency values, drop-down menus, e.g. for available traffic probes, etc., or whether it provides even wizard-like functionality to prepare a VSD from a blueprint.

Examples of VSBs are defined in Section 16.

6.2 Vertical Service Descriptors

The VSDs are the descriptors obtained after providing the missing information in VSBs. VSDs drive the 5G-TRANSFORMER service customer (TSC) to express its service request. A VSD has a number of fields that identifies it, and defines the SLA requirements and other properties of the vertical service. The structural information on the vertical service is derived from the corresponding VSB.

The following table presents the structure of a generic VSD.

TABLE 11: VERTICAL SERVICE DESCRIPTOR

Field	Description
Name	Name provided by the vertical for this VSD.
Description	Short description of the VSD, e.g. "Sensor monitoring for plant B".
Version	A version number.
Blueprint	The identifier of the blueprint from which this VSD was derived.
Identity	Unique identifier for a VSD. This is provided by the 5GT-VS when onboarding the VSD.
SST	For many vertical services it is possible to capture the SLA requirements by a predefined slice/service type (SST)[9], see Table 1 and Table 2. More complex, especially nested, vertical services might require SLA requirements in separate and more detailed statements, see the SLA field.
Service constraints	List of service constraints to be taken into consideration for the service orchestration and instantiation. The needed constraints are based on the business requirements listed in Section 3. These constraints can be: • Geographical constraints on the deployment area and on the location interconnections. • Security, i.e. the level of network slice isolation (ReqVS.B21).

[9] It is possible to extend the single-value SSTs by the more fine-granular categorization as proposed by 3GPP, see Section 3.1 Table 3.

	• Priority in case of resource shortage, see Section 5.4 on arbitration. • Cost. • Synchronization. • Preferred/non-preferred, prohibited infrastructure providers (ReqVS.B19). • Lifecycle independence (ReqVS.B20). • Etc. Some of these constraints could influence decisions with respect the usage of capabilities across the federation of SO domains.
Management and control capabilities for the tenant	Different options can be considered for control and management by the tenant of the slice to be provided (i.e., the vertical customer). Following [41], either provider managed or tenant managed slices are supported.
SLA	As an alternative to specifying a single SST, service level objectives, as agreed with the vertical customer can be provided. Any of the technical requirements as defined in [1], Section 4, can be defined. E2e latency refers to the latency among the connection endpoints in the service sequence and/or interconnections and may include also processing latency at the VNF/VF. An additional SLA requirement is e.g. the required bandwidth of external and internal interconnections.
Monitoring	• List of network KPIs, e.g. throughput at CPs or SAPs, load of instantiated VNFs, etc. • Additional traffic probes to be instantiated, e.g. traffic mirroring at a defined CP.
Lifetime	Information about aging, i.e. lifetime and/or start/end dates required for the slice to realize the service request (short-lived slices vs long-lives slices).
Charging	Information about the charging aspects [42] to be taken into consideration.

The atomic functional components, the service sequence, and the external and internal interconnections can be described in the notation of an ETSI NFV NSD [38].The atomic functional components are considered as VNFs and VAs, information similar to VNF packages can be described for them, see Table 12.

TABLE 12: INFORMATION ON ATOMIC FUNCTIONAL COMPONENTS

Field	Description
Number of Application servers.	The vertical may specify the number of application servers required for the service. If left open, this number is provided by the VSD/NSD Translator, see Section 5.3.

Images of virtual applications.	The vertical specifies path to the images of the virtual applications.
Virtual application connection end points	The vertical specifies the connection end points for the application.
Lifecycle operations	Scripts for instantiating, terminating, updating, healing, where applicable
Scaling rules	Rules when to scale in/out the virtual application

Examples of VSDs are presented in detail in Section 16.

6.3 Network Service Descriptors

In this section we describe the information elements that must be present in the Network Service Descriptors (NSDs) to properly define the NFV-NS behavior. The skeleton of the list of information elements presented below is tightly related to ETSI NFV IFA 014 [38], and it tries to gather those fields necessary for the NSs that are deployed in the context of the 5G-TRANSFORMER. In this section we describe information elements of NSDs that are extended by 5G-TRANSFORMER. For other information elements we list their most relevant fields in an annex in Section 15.

6.3.1 Network Service Descriptor (NSD)

The Network Service Descriptor (NSD) entity is the root entity that references other information elements in the description of a network service. The entity allows the complete description of a service that is deployed.

TABLE 13: NSD INFORMATION ELEMENT

Field	Description
NSD_id	Unique identifier for a NSD descriptor.
nested_NSDs	List of nested NSD_ids.
VNFD_ids	List of VNFD ids.
AppD_ids	List of AppD ids, see Section 6.3.2.
PNFD_ids	List of PNFD_ids.
SAPD_ids	List of SAPD_ids.
VNFFG	ID of the VNFFG.
flavours	List of NSDF ids.
signature	Authenticity signature to protect the descriptor.

6.3.2 MEC application Descriptor (AppD)

A NFV-NS may include MEC applications, which are described using the Application Descriptor (AppD) as specified by ETSI MEC [16]. The AppD is very similar to the VNFD, however it includes specific information requirements for MEC applications. The following table summarizes the fields which indicates MEC applications requirement in terms of traffic redirection and requested latency.

TABLE 14: APPD INFORMATION ELEMENT

Field	Description
appServiceRequired	ServiceDependency: Describes services a multi-access edge application requires to run.
appTrafficRule	List of TrafficRuleDescriptor: Describes traffic rules the multi-access edge application requires.
appDNSRule	Describes DNS rules the multi-access edge application requires.
appLatency	LatencyDescriptor: Describes the maximum latency tolerated by the multi-access edge application.

6.3.3 ETSI-NSD Gaps

In this subsection we present gaps that ETSI NSD cannot cover. We envision such gaps as features to be included in our NSDs in future stages of the project. Note that we extended NSDs already with MEC AppDs, see Sections 6.3.1 and 6.3.2.

6.3.3.1 Expressing location constraints

Although we expect that a vertical is not interested where exactly its VNFs or VAs are deployed, it still wants to express certain location constraints on a VSI. E.g. the vertical might want to express that the SAP to its VSI from the public internet or from its own company network is not too far away. As a second example, the vertical might require a certain coverage at the air interface of its VSI. In the definition of VSBs and VSDs we have defined already corresponding fields, see Table 10 and Table 11.

ETSI NFV allows to define location information for PNFs and VNFs. ETSI NFV IFA014 [38] allows to define geographicalLocationInfo for PNFs as part of the PNFD. ETSI NFV IFA013 [17] allows to define VNFLocationConstraints for VNFs as part of the NsInfo information element. This allows to provide information at instantiation time of a network service. Specifying the location of VNFs and PNFs does not solve the problem, as in the case of nested service definitions, a vertical might not even be aware which VNFs and PNFs are contained in the network service. As a second problem, for PNFs the location information is described in the PNFD. This implies that for each new service instance with a different location a new service descriptor has to be created.

Therefore, we propose to extend the definition of the SapData information element of ETSI NFV IFA 013 [17] with a field to express location constraints. The SapData information element can be provided also at instantiation time of a network service instance, therefore it is not necessary to create separate NSDs for almost similar network service instances.

Also, as the SAPs are defined for the border of network services, a vertical can express its constraints on where to connect to the service without having to know the internal details of the service. For the air interface we consider defining a SAP representing the air interface. The required air interface coverage can be expressed as location constraint of this SAP. Note, a location constraint can be either a single coordinate or a set of coordinates specifying one or several regions.

6.3.3.2 Expressing latency constraints on paths

So far, latency constraints among external or internal connection points of vertical services cannot be expressed adequately within NSDs. ETSI NFV IFA014 [38] allows to define QoS attributes such as latency or packet loss ratio for deployment flavors of individual VLs. But it is actually necessary to express latency constraints across a path within the network service, not just a single VL.

Therefore, we propose to extend the definition of NSDs with optional QoS constraints. We describe QoS constraints by a redefinition of the QoS information element of ETSI NFV IFA014 [38]: It consists of zero or more PointToPointConnectionConstraints and an optional priority. A PointToPointConnectionConstraint refers to two SAPs or CPs and has the same information, except the priority field, as already defined for VL DFs, see Table 15 and Table 16.

TABLE 15: REDEFINED QOS INFORMATION ELEMENT

Attribute	Qualifier	Card.	Content	Description
p2pConstraint	M	0..N	PointToPoint ConnectionConstraint	QoS constraints among two endpoints.
priority	M	0..1	Integer	Specifies the priority level in case of congestion on the underlying physical links.

TABLE 16: POINTTOPOINTCONNECTIONCONSTRAINT INFORMATION ELEMENT

Attribute	Qualifier	Card.	Content	Description
srcCpId	M	0..1	Identifier (Reference to VnfExtCpd or to PnfExtCpd or Sapd)	References one of the CPs or SAPs among which QoS attributes hold.
dstCpId	M	0..1	Identifier (Reference to VnfExtCpd or to PnfExtCpd or Sapd)	References one of the CPs or SAPs among which QoS attributes hold.
latency	M	1	Number	Specifies the maximum latency in ms.
packetDelay Variation	M	1	Number	Specifies the maximum jitter in ms.
packetLossR atio	M	0..1	Number	Specifies the maximum packet loss ratio.

We assume that the QoS constraints are the same for different instances of the same vertical service. Therefore, we propose to provide this information as part of the NSD, not as part of the instantiation parameters of network service instances. These QoS constraints will be used by the 5GT-SO in the placement decisions of VNFs. Eventually, VNFs and VAs should be placed such that these QoS constraints are satisfied. The 5GT-VS Monitoring Service can request performance monitoring data from the 5GT-SO to supervise whether these QoS constraints are satisfied.

6.3.3.3 Expressing latency constraints on virtual links

When investigating how to express latency constraints on paths we recognized a deficiency within the QoS attributes of deployment flavors of individual links as defined in ETSI NFV IFA014 [38]. The QoS attributes such as latency are defined for the VL, even for VLs of connectivity type mesh. But the latency could be different for each pair of connection points connected by the VL. These different latencies cannot be captured by a single latency value.

Therefore, we propose to use the redefined QoS information in Table 15 also for the qos field of VL DFs.

7 5GT-VS Workflows

As described in Section 5.1, the 5GT-VS consists of several components that interact locally and with external entities of the 5G-TRANSFORMER architecture: The Vertical, OSS/BSS, and the 5GT-SO. In this section, we will describe via workflows the 5GT-VS component roles to ensure vertical service instantiation and management.

7.1 Blueprint Onboarding

Description: The workflows describes how the 5G-TRANSFORMER service provider (TSP) onboards a VSB.

Prerequisite: none.

Assumptions: none.

Workflow: VSBs of vertical services are onboarded by the TSP through its OSS. As described in the workflow in Figure 11 the TSP provides in the request to onboard the VSB the description of the VSB, descriptions of referenced vertical and network services, and packages of the referenced VNFs (01).

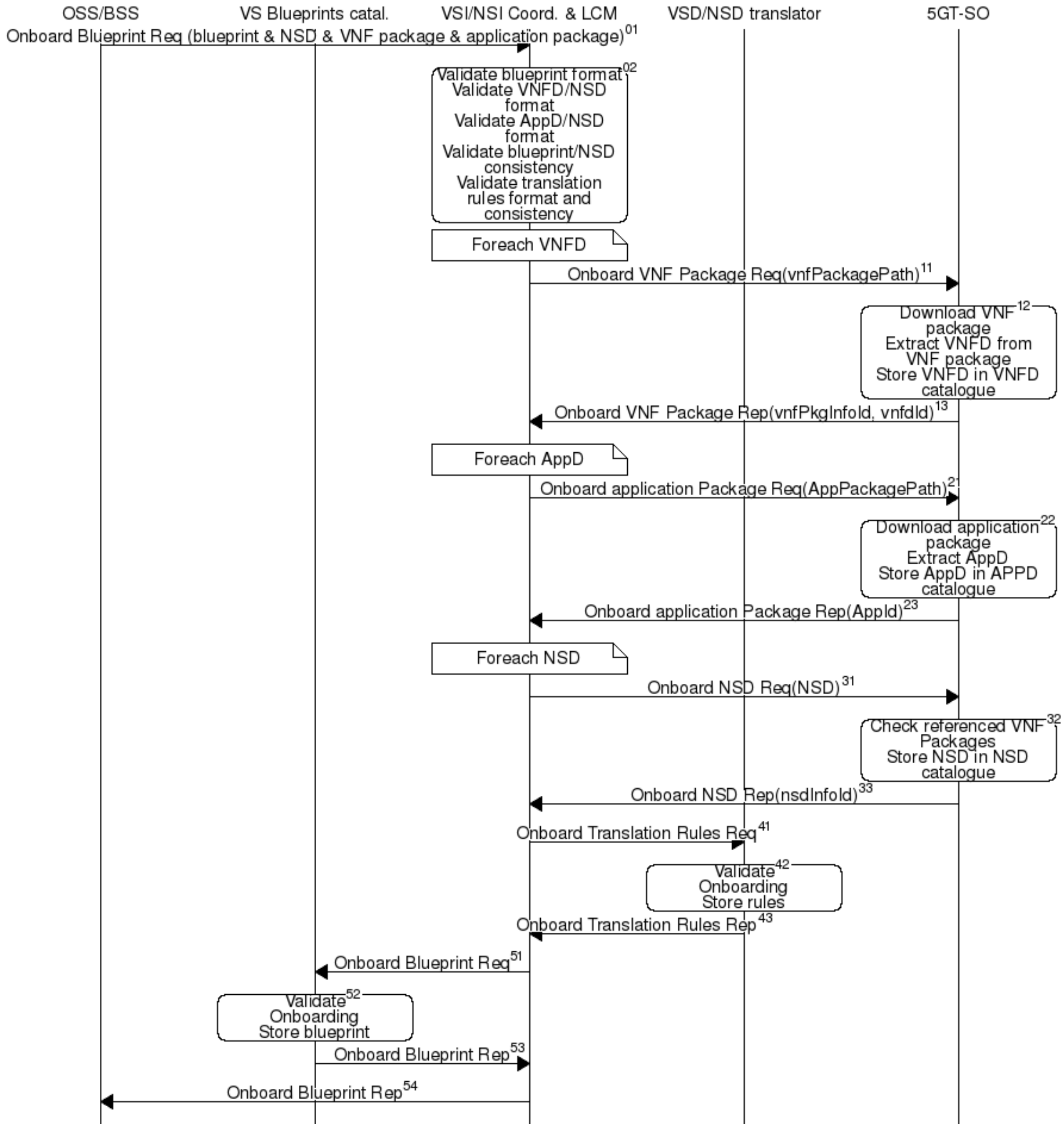


FIGURE 11: ONBOARDING BLUEPRINT WORKFLOW

The provided input is validated by the VSI/NSI Coordinator & LCM (02). If the validation is successful, the referenced VNF packages, application packages and descriptions of network services are onboarded to the 5GT-SO. VNF packages and NSDs which have been onboarded before to the 5GT-SO do not have to be onboarded again (11 - 13, 21 - 23, 31 - 33).

The translation rules from VSDs based on this blueprint to the corresponding NSDs are validated and stored in the Translator component. Onboarding the descriptors to the 5GT-SO and the translation rules to the Translator could also be done concurrently, there is no need to keep a strict order (41 - 43).

Once all the constituent parts of the blueprint have been onboarded to the respective components, the blueprint itself is validated and stored in the blueprint catalogue (51 - 53). From this point onwards, the blueprint can be used by verticals to prepare vertical services, see Section 7.2.

In case a MEC application is referenced in the blueprint, the AppD/NSD validation ensures that a MEC application could be run, by checking that the necessary related VNFD or AppD or PNFD are present in the NSD. Indeed, 5G-TRANSFORMER considers three scenarios to deploy a MEC application: (i) a MEC application requiring traffic redirection and consuming MEC services; (ii) a MEC application requiring only MEC services; (iii) MEC application requiring only traffic redirection. For the first scenario, the MEC application requires that also an EPC (MME, HSS, SPGW-C), SPGW-U, eNodeB(s) and Multi-access Edge Platform (MEP) are deployed. Specifically, the SPGW-U and the MEP should be run at the edge with the MEC application. For the second scenario, the MEC application needs eNodeB(s) and MEP. For the third scenario, the MEC application, requiring low latency access (reflected in the AppD), can be deployed without the need of other components. However, the 5GT-VS has to ensure in the VSD/NSD mapping that the traffic redirection as requested by the MEC application will be reflected in the Network Forwarding Path (NFP) included in the NSD. To summarize, the AppD/NSD validation of the 5GT-VS checks:

- Scenario 1: the presence of: VNFD or AppD of EPC elements, AppD of SPGW-U and MEP and PNFD of eNodeB(s).
- Scenario 2: the presence of: AppD of MEP and PNFD of eNodeB(s).
- Scenario 3: the presence of NFP reflecting the traffic rule described in the AppD of the MEC application.

7.2 Vertical Service Preparation

Description: The workflows described how a vertical creates the description of a new vertical service from a blueprint.

Prerequisite: Blueprints have been onboarded by the provider.

Assumptions: The vertical service does not contain VAs, therefore no VAs have to be onboarded as part of service preparation.

Workflow: A vertical *prepares* a VSD by selecting a VSB and providing the missing parameters. As described in Figure 12, there are three steps taken by the vertical.

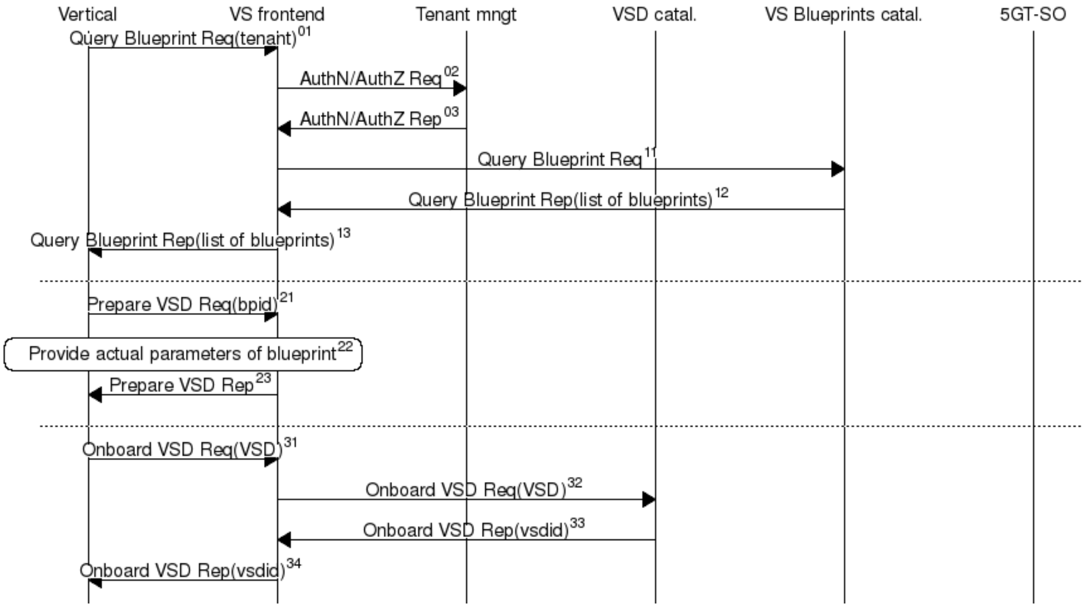


FIGURE 12: VERTICAL SERVICE PREPARATION WORKFLOW

At first, the vertical requests the list of available blueprints (01). As part of this step the vertical is authenticated and it is checked whether it is authorized to make this request (02, 03). Within the 5GT-VS, the blueprint catalogue is checked (11, 12) and the list of available blueprints is provided to the vertical (13). Depending on the authorization, the 5GT-VS may present a subset of all the blueprints to a vertical.

At second, the vertical selects a blueprint from the list of available ones and requests the VS frontend to prepare the VSD (21). As part of this step, the missing parameters for this vertical service are provided, such as requirements on latencies, amount of UEs supported, information on VM images for VAs, etc. (22).

At third, the vertical requests to onboard this specific VSD to the 5GT-VS (31), which stores this VSD in the corresponding catalogue (32, 33).

Note that no interaction with the 5GT-SO takes place, this workflow is about creating the description of the vertical service, not about its instantiation.

7.3 Vertical Service Instantiation

Description: The workflow describes the instantiation of a vertical service, triggered by the vertical.

Prerequisites: The vertical has selected a vertical service description, which is already onboarded to the vertical slicer.

Assumptions: The vertical service is a simple one, meaning it can be deployed in one NSI. We also assume that this service is deployed in a new NSI. We assume that the NSD, to which the vertical service is mapped and which describes the NSI has been onboarded to the 5GT-SO before.

Workflow: The vertical triggers the workflow, see Figure 13, by requesting the instantiation of a vertical service, identified by an identifier for the VSD (01). Before the workflow proceeds further, the vertical is authenticated and its authorization checked with the tenant management component (02, 03). Assuming this is successful, the 5GT-VS checks the existence of the VSD and retrieves it (11, 12), creates an entry for the instance of this vertical service (23) and stores it in its repositories (24). Note, this is internal bookkeeping of the 5GT-VS, the VSI is not deployed yet.

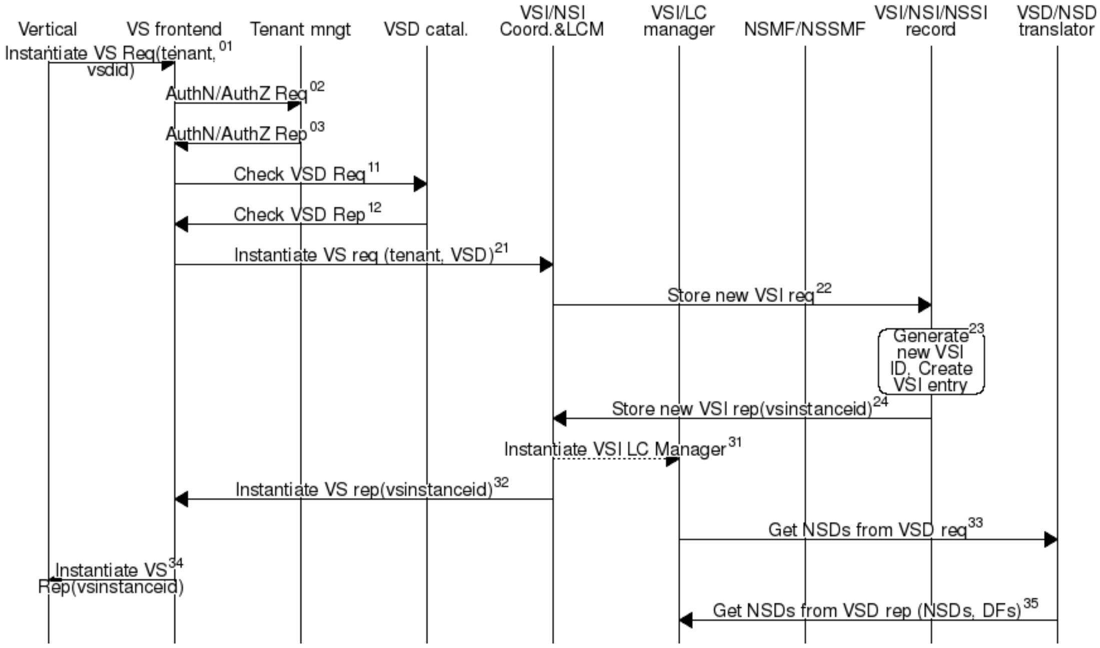


FIGURE 13: VERTICAL SERVICE INSTANTIATION WORFKLOW, PART 1

The VSI/NSI Coordinator &LCM creates a new instance of the lifecycle manager for this specific vertical service instance (31) and returns the instance id of this VSI to the Vertical frontend (32), which in turn returns it to the vertical (34). Concurrently, the VSI LC manager triggers the translation of the VSD into NSDs (33, 35).

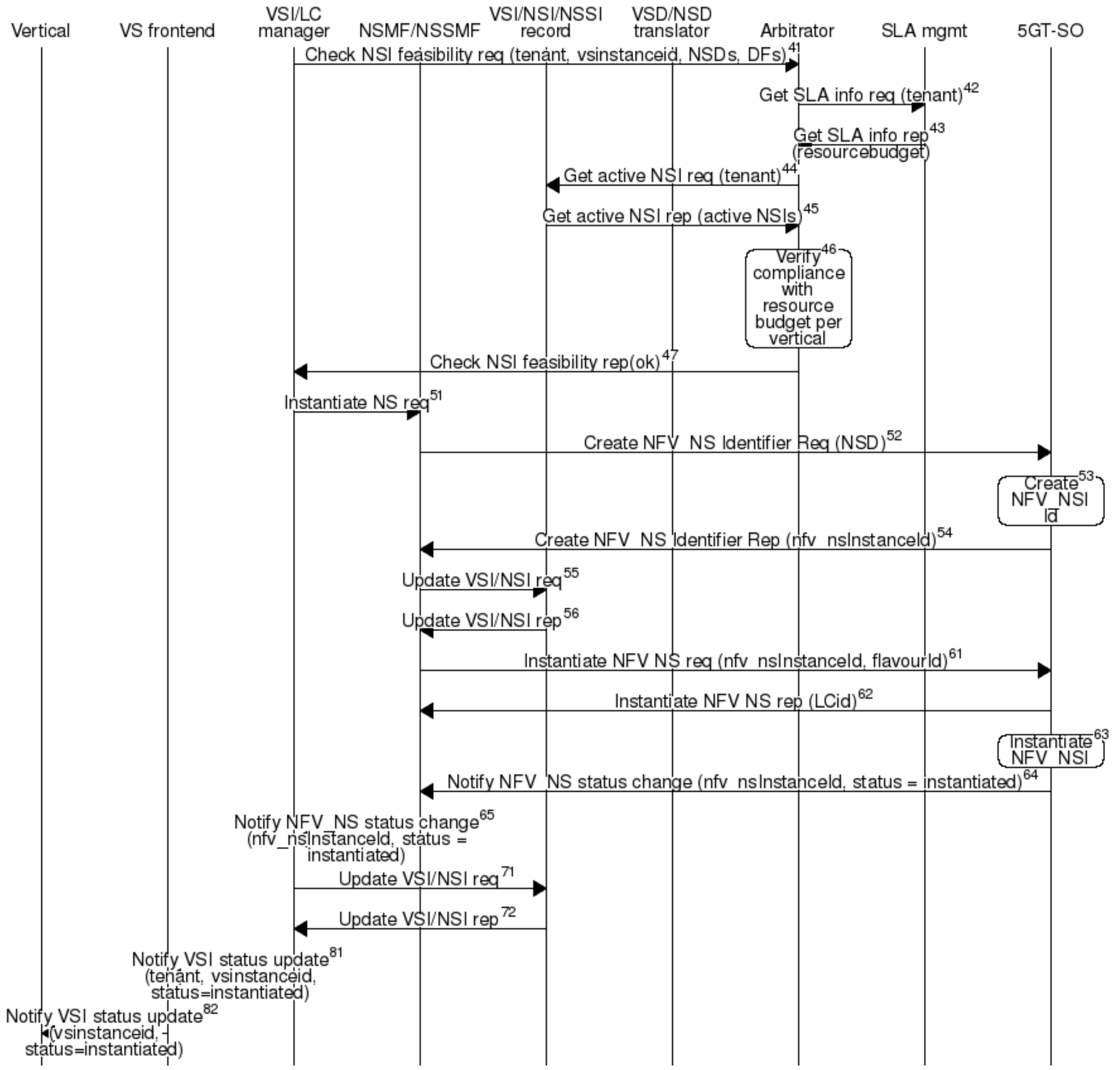


FIGURE 14: VERTICAL SERVICE INSTANTIATION WORKFLOW, PART 2

As next step, see Figure 14, the 5GT-VS checks whether this vertical service is feasible for this tenant. The VSI LC manager provides the information on the new vertical service and the NSDs, including the deployment flavours to the Arbitrator (41). The Arbitrator receives the resource budget of this vertical from the SLA manager (42, 43) and the list of active NSIs from the VSI/NSI repository (44, 45) and decides whether the requested resources still fit within the resource budget[10] of this vertical (46).

If the resource budget has not been exhausted already, the VSI LC manager triggers the NSMF (51) to request a new network service identifier from the 5GT-SO (52 - 54). The VSI/NSI record is updated with this information (55, 56). Now, the NSMF requests the actual instantiation of the network slice from the 5GT-SO (61). The NSI is described by a specific deployment flavour[11] of the NFV network service instance. The 5GT-SO makes the orchestration decisions on VNF placement and resource allocations (63).

[10] Note, this is a check against the resource budget of the vertical. This is not a check whether the infrastructure has sufficient resources.
[11] Deployment flavour as defined in ETSI NFV IFA014 [38], Section 5.3.

Once the NFV network service instance (i.e. network slice instance) has been instantiated on the actual infrastructure, the 5GT-SO notifies the NSMF (64), which in turn notifies the VSI LC manager (65), which updates the VSI/NSI record (71, 72). Eventually, the vertical is notified[12] about the instantiation of the vertical service (82).

7.4 Vertical Service Modification

Description: The workflow describes the modification of a vertical service instance, triggered by the vertical. The modification is a simple one, changing e.g. the amount of supported devices.

Prerequisites: The VSI has been instantiated.

Assumptions: The vertical service is a simple one, meaning it can be deployed in one NSI. We also assume, that this service is deployed in its own NSI, the service does not share the slice with another service.

Workflow: The workflow consists of two separate steps, see Figure 15 and Figure 16. Firstly, the VSD is changed, secondly, the VSD change is applied to a specific VSI. Note, that several instances of the same vertical service could be instantiated and only some of them are modified.

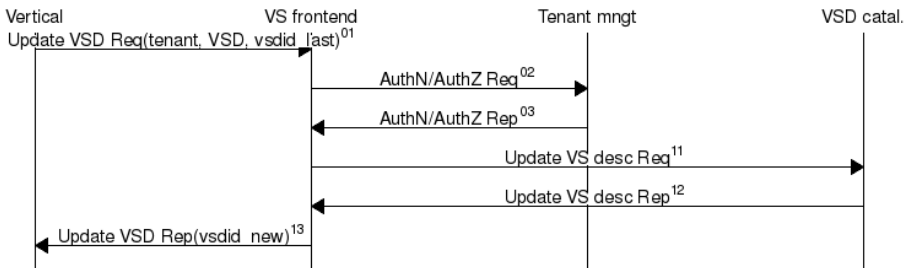


FIGURE 15: VERTICAL SERVICE MODIFICATION WORKFLOW, STEP 1

In the first step, the vertical provides the modified VSD. The relation to the other versions is kept by providing the identifier of the last version of this service descriptor as well (01). Firstly, the tenant is authenticated and its authorization is checked (02, 03). The new VSD is validated for correctness and stored in the corresponding databases (11, 12). An identifier for the new VSD is determined and provided to the vertical (13).

[12] It will be decided in the implementation phase whether notification is an operation initiated by 5GT-VS and consumed by the vertical or whether the vertical periodically polls the 5GT-VS. In case the interface between vertical and 5GT-VS is implemented as a GUI, this notification might just be a graphical indication in the GUI.

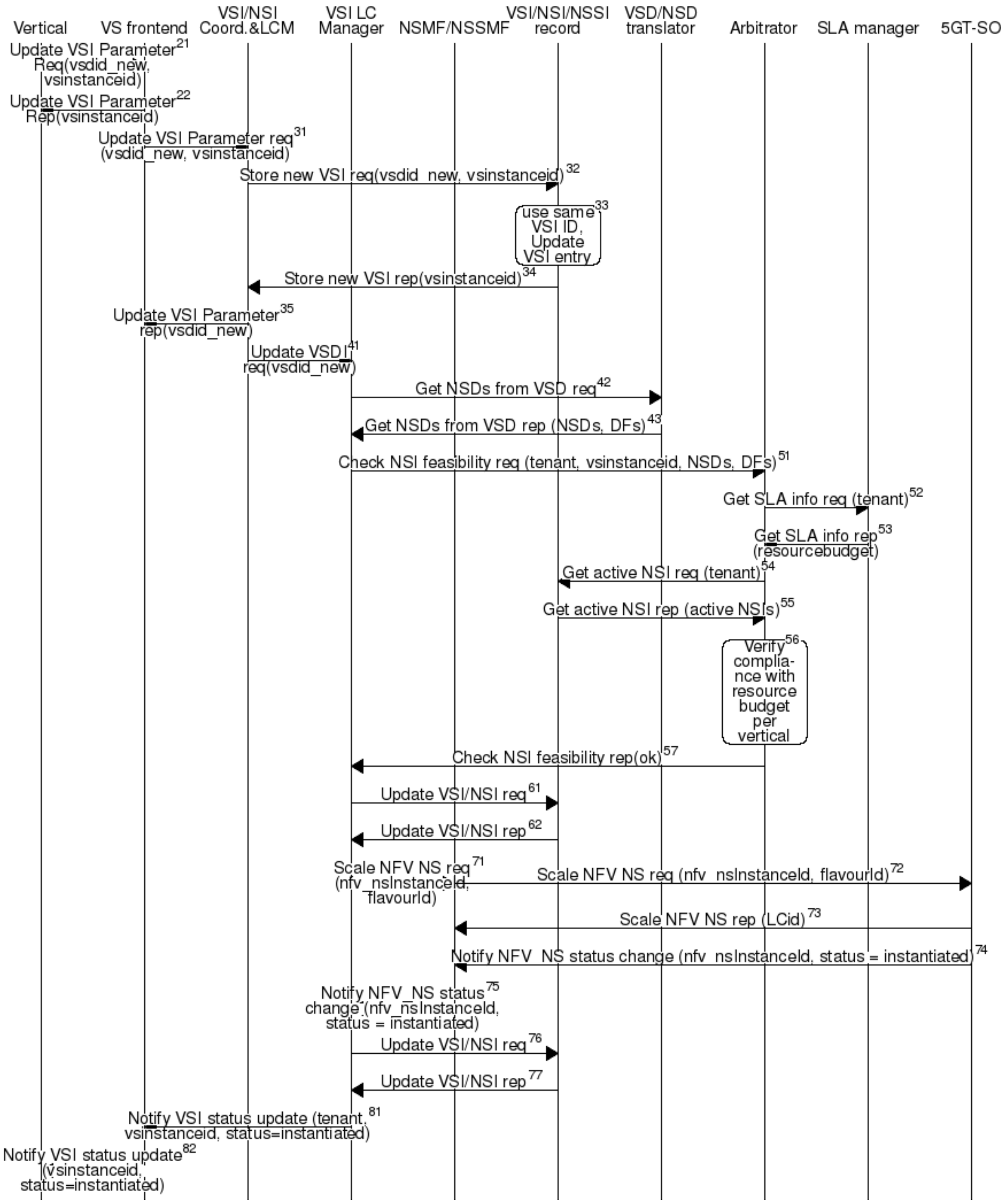


FIGURE 16: VERTICAL SERVICE MODIFICATION WORKFLOW, STEP 2

In the second step, the vertical applies the modified VSD to a VSI. The 5GT-VS can relate the vsinstanceid to the used version of the VSD (21). The operation returns immediately (22), while the actual processing continues.

The specific VSI is updated in the NSI/VSI record (31 - 35). Thereafter the VSI manager of this specific VSI triggers the remapping of the new VSD to the NSDs (42, 43) and the feasibility is checked as for newly instantiated service (51 - 57). Its state is updated in the VSI/NSI record (61, 62).

Thereafter the VSI manager informs the NSMF (71), which triggers the changes to the NSI by scaling its defining NSD at the 5GT-SO. In our simple example this could be a change of the cardinality of VNFs or a change of the deployment flavour. Note, depending on the extend of change of the VSD, the changes to the NSDs could be also more severe and might trigger more complex interaction among 5GT-VS and 5GT-SO. The NFV_NS Scale operation terminates (73), but the activities continue with applying the changes to the 5GT-MTP.Eventually the 5GT-SO notifies the 5GT-VS that the modification has finished (74, 75). In turn, the VSI LC manager updates the VSI/NSI record (76, 77) and notifies the VS frontend that the modification is finished (81). The VS frontend in turn notifies the vertical (82).

7.5 Vertical Service Termination

Description: The workflow describes the termination of a VSI, triggered by the vertical.

Prerequisites: The VSI has been instantiated.

Assumptions: This VSI is deployed in a single NSI, which is mapped to a single NFV network service.

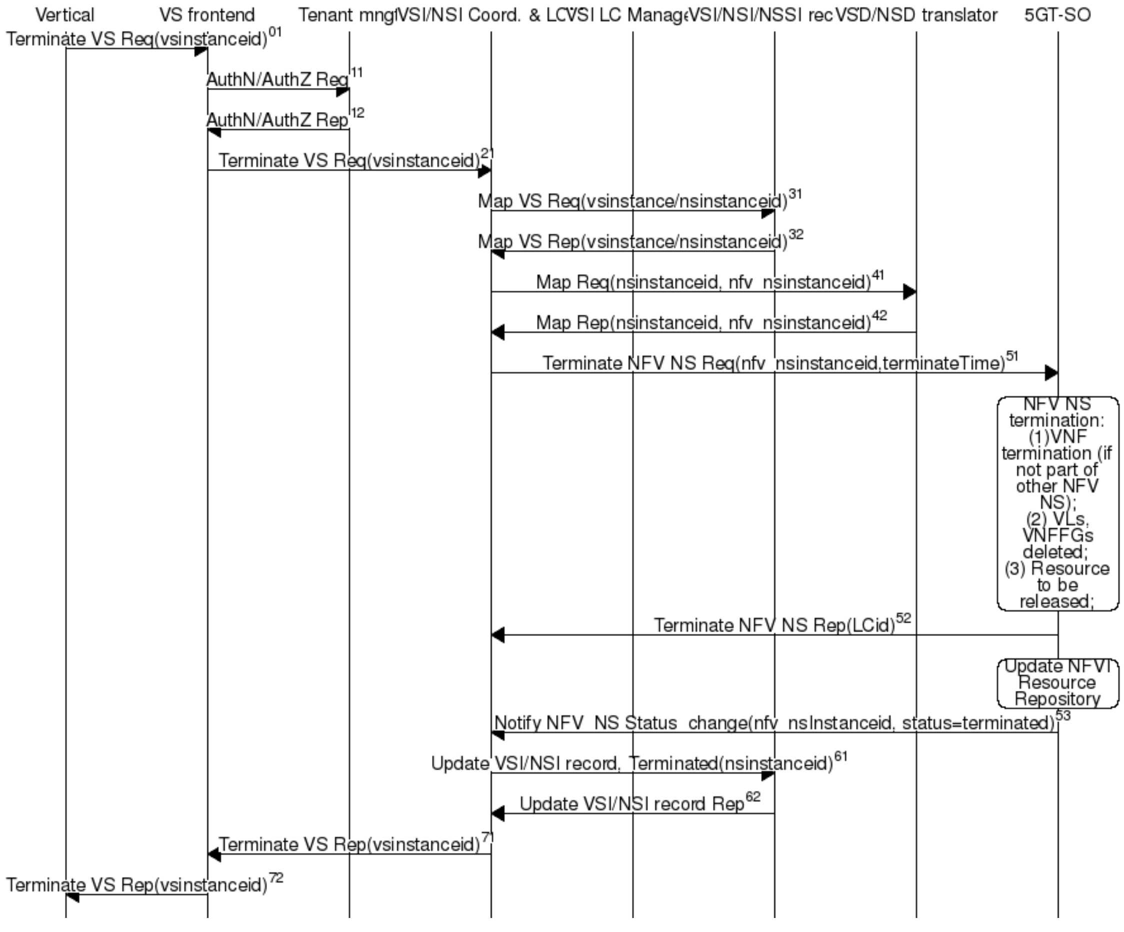


FIGURE 17: VERTICAL SERVICE TERMINATION WORKFLOW

The vertical triggers to terminate the VSI by sending the Terminate request (01) using the vsinstanceid via the VS frontend. First, the tenant is authenticated (11, 12). Then, the Terminate request is forwarded to VSI/NSI Coordinator & LC Manager (and VSI manager instance) (21), which maps the vsinstanceid to nsinstanceid, by sending a request to the VSI/NSI record (31, 32).

The VSI/NSI Coordinator & LC Manager sends a request (41, 42) to the VSI/NSD translator to obtain the corresponding nfv_nsInstanceid of the nsinstanceid. Now the VSI Coordinator/NSI Coordinator & LC Manager can send the termination request for the nfv_nsInstanceid with a specific timeout (according to ETSI NFV IFA 013 [17]). The 5GT-SO initiates VNFs termination, VL and VNFFG deletions and release of orchestrated resources. The 5GT-SO response for termination of the lifecycle for the nfv_nsInstanceId including the LCid or lifecycleOperationOccuranceId (according to ETSI NFV IFA 013 [17]). Then the 5GT-SO updates the NFVI Resource Repository concerned by the recently released resource. The 5GT-SO sends a notification message with the terminated status of the nfv_nsInstanceId (53).

The VSI/NSI coordinator & LC Manager concludes the termination of the nsInstanceId that corresponds with the terminated nfv_nsInstanceId. It first updates (61, 62) the VSI/NSI record in order to remove the nsinstanceid, and informs (71) the VS frontend. The latter (72) will in turn notify the Vertical about the success of the vertical service termination procedure.

7.6 Vertical Service Monitoring

This workflow corresponds to the use-case, where the 5GT-VS monitors a VSI and consumes the monitoring data itself.

Description: The workflow describes procedures to allow the 5GT-VS to collect monitoring data related to a network service.

Prerequisites: All the descriptors have been on-boarded, the vertical has requested the instantiation of the vertical service and the 5GT-VS has already created an nsi_Id.

Assumptions: The MonitoredData in the NSD includes only MonitoringParameters, not VnfIndicatorInfo. There is a 1:1 relationship between VSI and network service instance (no sharing or nesting).

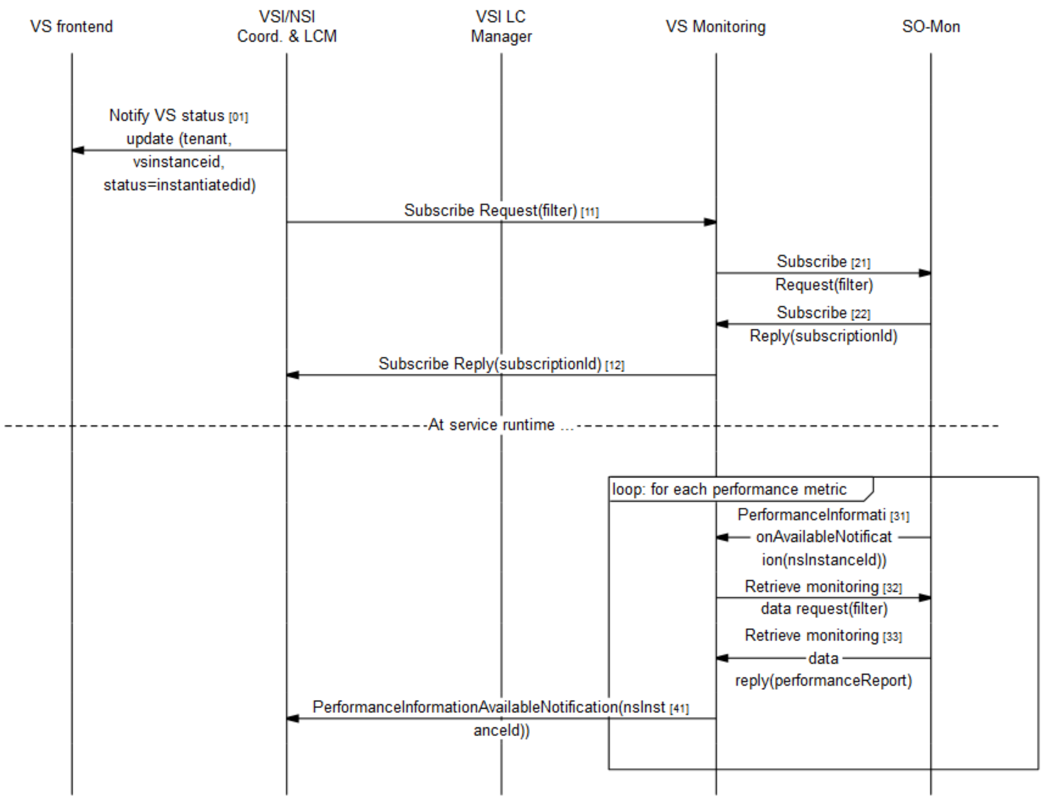


FIGURE 18: VERTICAL SERVICE MONITORING WORKFLOW

The monitoring procedure should start only after the vsinstanceid is instantiated (status=instantiated). The VSI/NSI Coordinator & LC Manager (and VSI manager instance) requests (11) to subscribe to specific monitoring metrics to the monitoring component of the 5GT-VS. The latter sends a request to the 5GT-SO Monitoring Service indicating the monitoring metrics to subscribe to (21, 22).

At service runtime, the 5GT-SO Monitoring Service indicates the availability of monitoring data to the 5GT-VS monitoring element. The latter retrieves the data (32, 33), and informs the VSI coordinator (41).

8 Conclusions

In this deliverable we have provided the requirements on the 5GT-VS, the internal architecture of the 5GT-VS, its NBI and SBI, the descriptions used by verticals and those to describe network slices, and the workflows among the components.

In this deliverable we focus on the support for verticals to deploy services by selecting and completing entries from the blueprint catalogue. We have provided example blueprints and VSDs. We plan to provide blueprints and corresponding VSDs for all the use cases investigated within this project, see [1], Section 5. Regarding the different descriptors, we have focused on the information model. We plan to investigate what would be the most suitable data model, with TOSCA, see Section 4.2, being a good candidate. We plan to investigate, how the blueprint based approach of defining vertical services can be complemented by allowing the verticals to compose their vertical services from smaller building blocks. This would allow more flexibility to the verticals to define their services.

In this deliverable we have focused on the arbitration among vertical services of one vertical. We plan to consider also the arbitration among different verticals, taking into account a classification into Gold, Silver, and Bronze verticals. We plan to investigate further the support for NFVIaaS and NSaaS, see Section 5.9. Specifically, the 5GT-VS architecture will be extended and its NBI be defined in similar detail as in Section 5.7.

Essential aspects of the 5GT-VS will be implemented [6] and demonstrated [5] in the course of the project. We aim to demonstrate that the 5GT-VS actually allows verticals to define their services in an easy way and to deploy them in a short timeframe. This will enable 5G-TRANSFORMER service providers to offer their 5G and general IT infrastructure to verticals, enabling new business cases. The demonstrations for the 5GT-VS will focus on the use cases of the vertical industries participating in this project.

9 References

[1] 5G-TRANSFORMER, D1.1, Report on vertical requirements and use cases, 2017.

[2] 5G-TRANSFORMER, D1.2, 5G-TRANSFORMER initial system design, May 2018.

[3] 5G-TRANSFORMER, D2.1, Definition of the Mobile Transport and Computing Platform, 2018.

[4] 5G-TRANSFORMER, D4.1, Definition of service orchestration and federation algorithms, service monitoring algorithms, 2018.

[5] 5G-TRANSFORMER, D5.1, Definition of vertical testbeds and initial integration plans, May 2018.

[6] 5G-TRANSFORMER, D3.3, Final design and implementation report on the Vertical Slicer, May 2019.

[7] C. Casetti et.al, Network Slices for Vertical Industries, 1st Workshop on Control and management of Vertical slicing including the Edge and Fog Systems (COMPASS), Barcelona, IEEE, 2018.

[8] C. Casetti et.al., The Vertical Slicer: Verticals' Entry Point to 5G Networks, submitted to EUCNC 2018.

[9] A. de la Oliva et.al., 5G-TRANSFORMER: Slicing and Orchestrating Transport Networks for Industry Verticals, IEEE Communications Magazine, June 2018, (accepted for publication).

[10] Xi Li et.al., Service Orchestration and Federation for Verticals, 1st Workshop on Control and management of Vertical slicing including the Edge and Fog Systems (COMPASS), Barcelona, IEEE, 2018.

[11] ETSI GS NFV-MAN 001, V1.1.1, Management and Orchestration, 2014.

[12] P. Iovanna et.al., 5G Mobile Transport and Computing Platform for verticals, 1st Workshop on Control and management of Vertical slicing including the Edge and Fog Systems (COMPASS), Barcelona, IEEE, 2018.

[13] 3GPP TR28.801, V15.0.0, Telecommunication management; Study on management and orchestration of network slicing for next generation network, 2017.

[14] 3GPP TS23.501, V15.0.0, System Architecture for the 5G System; Stage 2, 2017.

[15] 3GPP TS28.530, V0.4.0, Telecommunication management; Management of 5G networks and network slicing; Concepts, use cases and requirements, 2017.

[16] ETSI MEC, [Online], http://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing

[17] ETSI GS NFV-IFA-013, V2.4.1, Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification, 2018.

[18] B. Chatras, U. S. Tsang Kwong and N. Bihannic, "NFV enabling network slicing for 5G," 2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN), Paris, 2017, pp. 219-225.

[19] ETSI GS NFV-IFA-005, Management and Orchestration; Or-Vi reference point - Interface and Information Model Specification", v2.4.1, 2018.

[20] 3GPP TS 38.300, "NR; NR and NG-RAN Overall Description; Stage 2; (Release 15)", v15.0.0, December 2017.

[21] 3GPP TR 28.531, "Management and orchestration of networks and network slicing; Provisioning; Stage 1 (Release 15)", v0.3.0, 2018.

[22] ETSI GS NFV-IFA 028, Management and Orchestration; Report on architecture options to support multiple administrative domains", v3.1.1, 2018.
[23] ETSI GS NFV-IFA 010, Management and Orchestration; Functional requirements specification", v2.4.1, 2018.
[24] ETSI GR NFV 001, "Network Functions Virtualisation (NFV); Use Cases", v1.2.1, 2017.
[25] ETSI GR NFV-IFA 022, Management and Orchestration; Report on Management and Connectivity for Multi-Site Services", v0.8.2, 2018.
[26] ETSI GR MEC 017, "Mobile Edge Computing (MEC); Deployment of Mobile Edge Computing in an NFV Environment", v1.1.1, 2018.
[27] ETSI GS NFV-INF 003, Infrastructure; Compute Domain", v1.1.1, 2014.
[28] ETSI GS NFV-INF 004, Infrastructure; Hypervisor Domain", v1.1.1, 2015.
[29] ETSI GS NFV-INF 005, Infrastructure: Network Domain, V1.1.1, 2014.
[30] TeleManagement Forum. "TeleManagement Forum Information Framework (SID): GB922_Addendum_4SO_Service_Overview_R9-5_v9-7".
[31] ETSI GS NFV-IFA 008, "Ve-Vnfm reference point - Interface and Information Model Specification." V2.4.1, Release 2, Feb. 2018.
[32] 5G-PPP, "5G: Serving vertical industries", https://5g-ppp.eu/2nd-5g-vertical-workshop/
[33] ETSI GS MEC 010-2, V1.1.1, Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management, 2017.
[34] Topology and Orchestration Specification for Cloud Applications Version 1.0. 25 November 2013. OASIS Standard. [Online] http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.html
[35] TOSCA Simple Profile for Network Functions Virtualization (NFV) Version 1.0, Edited by Shitao Li and John Crandall. 11 May 2017. OASIS Committee Specification Draft 04, [Online] http://docs.oasis-open.org/tosca/tosca-nfv/v1.0/tosca-nfv-v1.0.pdf
[36] The TOSCA Cloud Service Archive (CSAR), [Online] https://www.oasis-open.org/committees/download.php/46057/CSAR%20V0-1.docx
[37] YAML, [Online], http://yaml.org/
[38] ETSI GS NFV-IFA-014, V2.4.1, Management and Orchestration; Network Service Templates Specification, 2018.
[39] ETSI GS NFV-IFA 011, V2.3.1, Management and Orchestration; VNF Packaging Specification, 2017.
[40] ETSI GR NFV-EVE 012 V3.1.1, Evolution and Ecosystem; Report on Network Slicing Support with ETSI NFV Architecture Framework, 2017.
[41] Luis M. Contreras and Diego R. López, A Network Service Provider Perspective on Network Slicing, IEEE Softwarization, Jan 2018. [Online] https://sdn.ieee.org/newsletter/january-2018/a-network-service-provider-perspective-on-network-slicing
[42] TM Forum, Impact of SDN/NFV on Charging and Billing, [Online] https://www.tmforum.org/resources/how-to-guide/gb989-impact-of-sdnnfv-on-charging-and-billing-r15-5-1/
[43] ETSI GS NFV-IFA 006, V2.3.1, Management and Orchestration; Vi-Vnfm reference point - Interface and Information Model Specification, 2017.
[44] RFC 2119, "Key words for use in RFCs to Indicate Requirement Levels", IETF, 1997.
[45] 3GPP TS23.682, V15.1.0, Architecture enhancements to facilitate communications, with packet data networks and applications, 2017.
[46] SONATA NFV, http://www.sonata-nfv.eu/

[47] ETSI GS NFV-IFA 007, V2.1.1, Or-Vnfm reference point - Interface and Information Model Specification, 2016.
[48] SONATA D2.2. Architecture Design, [Online] http://www.sonata-nfv.eu/content/d22-architecture-design-0

10 Annex I: Vertical Services

The 5G-TRANSFORMER consortium includes partners from several of the Vertical Industries identified in the market portfolio of the 5G-PPP [32], namely: automotive, entertainment, healthcare and manufacturing and also representatives from the Mobile (Virtual) Network Operator (MNO/MVNO) industry. The following sections summarize the use cases (UC) established in [1], and detail the expectations within the 5G-TRANSFORMER project in order to ease the reading of the rest of the document.

10.1 Automotive

The automotive industry is currently undergoing key technological transformations, as more and more vehicles are connected to the Internet and to each other, and advances towards higher automation levels. In order to deal with increasingly complex road situations, automated vehicles will have to rely not only on their own sensors, but also on those of other vehicles, and will need to cooperate with each other, rather than to make decisions on their own.

These trends pose significant challenges to the underlying communication system, as information must reach its destination reliably within an exceedingly short time frame - beyond what current wireless technologies can provide. 5G, the next generation of mobile communication technology, holds the promise of improved performance in terms of reduced latency, increased reliability and higher throughput under higher mobility and connectivity density.

Vehicle domain features differ across the target operative scenarios which are strongly characterized by their own peculiarities. In order to better analyse the needs of the automotive domain versus the incoming communication technology, we considered four main scenarios (urban, rural, highway and transversal) and several use cases quite different for their peculiar features outlining the key aspects that mostly impacts on 5G.

Typical automotive UCs are various and can address heterogeneous domains. In [1] more than 25 UCs from those most popular in the literature have been described; the identified UCs are grouped in 6 domains: safety, mobility, entertainment, e-road, digitalized Vehicles and automated vehicle.

In the 5G-TRANSFORMER project, we focus on the safety domain where, thanks to 5G capabilities, the vehicle can outline/foresee dangerous situations and properly react on time. In particular, two use cases have been selected and proposed for implementation:

TABLE 17: AUTOMOTIVE USE CASES

ID	Goal In context	General description
UC A.01, UC A.02	Avoid possible collision crossing intersection	The purpose of the Intersection Collision Avoidance (ICA) system is to alert drivers about the existence of any possible obstacles and eventually activate the emergency braking system. The communication infrastructure facilitates a real-time exchange of data between the involved entities.
UC A.04	Vehicles are able to see through obstacles, thanks to cooperation	Thanks to the cooperation between vehicles, streaming information is provided to all the vehicles that want/need to access to it. This

	among them achieving bilateral awareness of road conditions	information can be used to identify potential obstacles that cannot be detected through on-board sensors.

10.2 Entertainment

The Media and Entertainment (M&E) industry is one of the industries most affected by the deep changes in terms of user habits and expectations that the society has been experiencing with the explosion of the Internet. The amount of users grows daily, and the users demand progressively media-rich contents and a better quality of experience.

While all these changes provide a great economical fuel for the industry, they also impose challenges to the network infrastructures (in terms of data rates, number of connections, quality of experience, etc.) not present before. The 5G PPP already identifies the entertainment industry as one of the key interested parties. This is because one of the key objectives of 5G is to open operators' networks for new services, and this is the key enabler to support the data rates and the latency required to give an immersive experience. Furthermore, 5G also aims to provide the services with network information not available before (i.e. packet losses, signal level, etc.) to better adapt the service to the network conditions.

The 5G-TRANSFORMER project focuses on the M&E services particularly targeting sports events. The aim is to encompass these services to the FAN ENGAGEMENT trend, which envisions smarter venue services by means of providing targeted and high-quality content and following fans along the journey with contextualized information. This trend also envisions fans as content producers (i.e., to share videos, photos, emotions, opinions, comments, etc.), and captures the explosion of IoT devices by including them as additional content producers. The final goal is to give the fans a more interactive, immersive and participative experience like never before.

The following use cases are considered in the project, in order to address the needs for the different actors and scenarios identified in [1] for the Entertainment vertical industry:

TABLE 18: ENTERTAINMENT USE CASES

ID	Goal In context	General description
UC.E01	To provide a better fan experience to users attending (on-site) an event	Large-scale event sites, such as stadiums are more and more being connected in order to give a better experience to their customers (replay, choose a specific camera, language, augmented reality to bring additional information, etc.)

10.3 eHealth

The eHealth use case is one of the most critical verticals we have in the 5G-TRANSFORMER project. This industry can effectively take advantage of the future 5G networks to improve the quality of life and medical assistance of people in emergency situations. It aims to be able to detect and assist people in emergencies in the minimum possible time in order to assure the maximum probability of people passing the emergency and recover from it.

5G networks will be able to support high demands of traffic with low delay requirements. Thus it is very helpful for the eHealth use case because that allows discovering and attending emergencies in short time. Hence, there are two main targets for this use case: e-Infrastructure and eHealth application.

On the one hand, the e-Infrastructure use case focuses on how the current municipality infrastructure based on Terrestrial Trunked Radio (TETRA) can be replaced based on the 5G features. This will allow emergency alarms to be received with smaller delay and thus, be processed in a small amount of time to send an ambulance to the place of the emergency. This will also allow to access in real time the clinical history of the patient from the place of the incident to give the patient a better medical attention. In addition, to have a better e-Infrastructure, the eHealth use case will need a high-priority and low latency service in the 5G-TRANSFORMER system. To address that, the 5G-TRANSFORMER system will allow to have access to the resources of the eHealth slice in extreme cases where the network is overloaded by users like in big events.

On the other hand, the eHealth application aims to study how new technologies such as Multi-access Edge Computing (MEC) can help improving the speed of response. This application tries to reduce the response time and automate processes of communicating between the patient and the medical personnel and among the medical personnel. The idea is to have an application based on MEC for automating the collection of data from wearables, detection of problems and automatic calling the ambulance, which requires mechanisms for patient feedback (call back). If possible, it is important to provide video feed between emergency team and doctor at the hospital because the personnel in the ambulance is not specialized to deal with in some emergencies such as an urgent surgery in the most extreme case. In that case, they need to contact a doctor that monitors and guides the process over real-time 4K video.

TABLE 19: EHEALTH USE CASES

ID	Goal In context	General description
UC.H01	To provide a better medical assistance in emergency cases	Large-scale event sites where a lot of groups are deployed to cover the emergencies and have to communicate between them in real time. Emergencies that requires real time communication between the ambulances and doctors. Improvement of the current infrastructure to guarantee the real time exchange of information to detect early the emergencies.

10.4 eIndustry

The production and manufacturing industry is currently undergoing important changes mainly driven by the ongoing introduction of new emerging technologies, including mobile network, cloud computing, robotics, machine intelligence and big data. Nowadays we are facing a new industrial revolution, commonly referred to as Industry 4.0, whose aim is to provide mass customization with costs comparable to those of mass production. This can be achieved leveraging on full digitalization and automation of industrial processes.

The major ingredient to ensure full digitalization and automation is the virtualization of control, allowing to centralize all the intelligence of the operations in order to increase flexibility and facilitate the changes of the manufacturing plants. Moreover, it is essential to monitor all the elements of an industrial manufacturing plant through wireless connectivity (in order to avoid cabling that further increases complexity) and information processing (including big data and analytics technologies). These enhanced functionalities introduce strict requirements on data rates, latency, reliability, etc., all of which are addressed in the 5G mobile transport and computing platform.

The role of 5G in Industry 4.0 extends to large area logistics (i.e. in the optimization of maritime, ground, air transportations, as well as to optimize port operations and goods production processes), where there is a similar need to increase the productivity and the efficiency of the processes to cut production costs and become more and more competitive.

In [1] several use cases have been identified for the e-Industry vertical, namely monitoring in production line, cloud robotics, automated logistics, electric power generation, electric power transmission and electric power distribution. Several uses cases can coexist in different scenarios. For example, in an automated factory both monitoring application and cloud robotics solutions can be in use. All use cases presented in [1] involve enabling more efficient manufacturing and lean production which poses severe requirements on the underlying communication network making it essential that the industrial environment be equipped with 5G solutions.

Among all the e-Industry use cases, the cloud robotics has been selected as candidate for implementation in the final demonstrators of 5G-TRANSFORMER project.

TABLE 20: EINDUSTRY USE CASES

ID	Goal In context	General description
UC I.02	Highly automation of the factory plant is provided moving the control of the production processes and of the robots functionalities in cloud, exploiting wireless connectivity to minimize infrastructure, optimize processes, implement lean manufacturing.	The controlling functionality of the robots is moved to the cloud, in order to utilize its massive computing power. Huge amounts of information will have to be transferred instantaneously. With lower latency and higher bandwidth than other forms of wireless connectivity, 5G is the optimal choice.

10.5 MNO/MVNO

Increasing the capacity and the elasticity of mobile network operators' networks is one of the most important challenges foreseen in 5G networks, as it will allow opening MNOs business toward new markets and a large variety of tailored services. This evolution is especially brought through the convergence of mobile networks and cloud infrastructures, which provides the capability for mobile operators to use network function virtualization (NFV) concepts and cloud-based infrastructures in order to virtualize and decentralize their network entities. Hence, the MVNO business model emerges from this evolution through the creation of a new business model that disrupts the traditional mobile value chain. In the MVNO model, new players can participate in the mobile value chain and extract value to leverage their valuable assets.

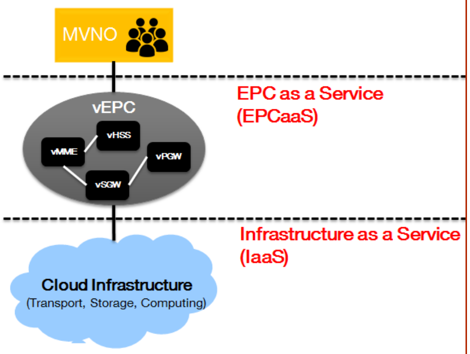
In 5G-TRANSFORMER, the MNO/MVNO industry is especially relevant and interesting because of this new role model it injects into the mobile value chain and also because of the nature of services it is offering compared to the other studied vertical industries in the project. For instance, offering the Network as a Service (NaaS) or the Infrastructure as a Service (IaaS) are types of services that are challenging for MNOs and MVNOs in order to reach real on demand and finely tailored services for their customers.

Thus, the MNO/MVNO player has a different role compared to the other verticals. In fact, the relation between the MNO/MVNO and the 5G-TRANSFORMER system depends on the chosen MVNO business model. For instance, in the case of a Full MVNO or a Mobile Virtual Network Enabler (MVNE) business model as described in [1], the role of the MVNO exceeds that of a simple vertical service provider and has almost the same role as an MNO acting as a Network service provider. Likewise, the role of the MNO hosting an MVNO is built on the offering of a Network as a Service (NaaS); for instance, the MNO would rely on network slicing combined to services like EPCaaS and IaaS in order to set up an MVNO network and provide network services like connectivity to the MVNO, In addition, verticals can be seen as customers of an MNO or an MVNO.

In [1], several use cases have been identified as relevant for the MNO/MVNO domain in 5G-TRANSFORMER. We chose to focus here on the use case UC M.01 vEPCaaS. This use case especially describes how the MNO/MVNO can offer Network as a Service (NaaS) for its customers by offering a dedicated and on-demand core network. In the same context of offering Network as a Service, we also investigate the particular case of NFVIaaS as an example of IaaS. In this particular case of IaaS, the challenge resides in the fact that the correspondent MVNO business model may imply the ownership by the MNO/MVNO of an OSS/BSS that provides him with the capability to create and configure its own network slice instances for its customers and the non-ownership of an NFV infrastructure. In this case, 5G-TRANSFORMER will offer NFVIaaS for the MNO/MVNO. One possible question in this case is whether it is possible for an MNO/MVNO to request a network slice with particular Network Functions Virtualisation Infrastructure (NFVI) resources and through which interface or service catalogue would this be possible.

TABLE 21: MNO/MVNO USE CASES

ID	Goal in context	General description
UC M.01	Build of an MVNO service through the deployment and operation of a network slice with a vEPC in “as a Service” mode.	The vEPC can be instantiated as a virtualized Control plane only or as a complete virtualized Control and User planes core network. The vEPC is supposed to provide the same implementation and performances of a real EPC that is deployed on a real infrastructure. The use of a vEPC should be totally transparent and should not

		impact services' end to end latency.

11 Annex II: Reference Architectures

A few Standard Development Organizations (SDOs) and fora are contributing to the design of management systems for 5G that have many common design principles: (i) flexibility, (ii) adaptability, and (iii) cost-efficiency. Despite the many common design objectives across these working groups (as presented below), there are various relevant architectural concepts (e.g., slicing, federation/multi-domain, edge computing) that are specific to individual groups. In this context, 5G-TRANSFORMER strives to bridge the gaps across such heterogeneous ecosystem in order to harmonically integrate these concepts under a single architecture.

This section introduces ongoing architectural work at 3GPP and ETSI NFV and MEC that fulfills two objectives:

- Serve as inspiration to define the 5G-TRANSFORMER architecture;
- Set the framework in which 5G- TRANSFORMER must be integrated to maximize its impact, i.e., by seeking as much as possible compliance with what is already defined.

Despite the fact that there are other organizations discussing about slicing and architectural concepts related with 5G-TRANSFORMER, we focus on the ones below because they are the ones with a more complete definition of their architecture and building blocks, and so, they go well beyond requirements and high-level concepts.

11.1 3GPP

The most relevant working groups inside 3GPP related to 5G-TRANSFORMER are SA2 (Architecture) and SA5 (Telecom management).

11.1.1 3GPP SA2

The 3GPP SA2 Working Group (WG), responsible for overall system architecture, is currently working on specifying the 5G Core (5GC) architecture with network slicing being a main feature of 5GC. Technical Specification (TS) 23.501 [14] defines Stage-2 system architecture for the 5G system which includes network slicing. Figure 19depicts an example of network slicing from 3GPP's perspective.

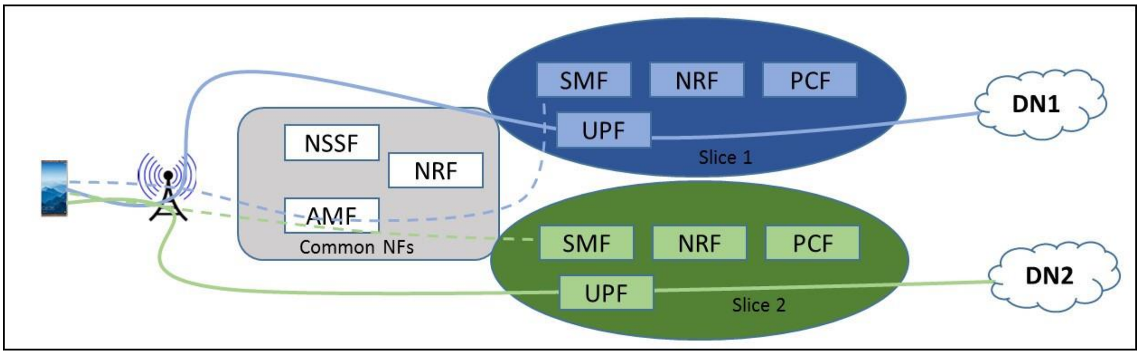


FIGURE 19: EXAMPLE OF NETWORK SLICES FROM 3GPP SA2 PERSPECTIVE

A network slice is viewed as a logical end-to-end network that can be dynamically created. A given User Equipment (UE) may access to multiple slices over the same Access Network (e.g. over the same radio interface). Each slice may serve a particular service type with agreed upon SLA. In the following, we provide highlights of 3GPP network slicing as being defined in TS 23.501 [14] in SA2.

A network slice is defined within a Public Land Mobile Network (PLMN) and includes the Core Network Control Plane and User Plane Network Functions as well as the 5G Access Network (AN). The 5G Access Network may be a Next Generation (NG) Radio Access Network described in 3GPP TS 38.300 [20] or a non-3GPP Access Network.

TS 23.501 [14] defines Network Function, Slice, and Slice Instance as follows:

- Network Function: A 3GPP adopted or 3GPP defined processing function in a network, which has defined functional behavior and 3GPP defined interfaces. (Note: A network function can be implemented either as a network element on a dedicated hardware, as a software instance running on a dedicated hardware, or as a virtualized function instantiated on an appropriate platform, e.g. on a cloud infrastructure.);
- Network Slice: A logical network that provides specific network capabilities and network characteristics;
- Network Slice instance: A set of network function instances and the required resources (e.g. compute, storage and networking resources) which form a deployed network slice.

11.1.2 3GPP SA5

3GPP SA5 Working Group (WG) is the 3GPP telecom management working group. 3GPP SA5 specifies the requirements, architecture and solutions for provisioning and management of the network, including Radio Access Network (RAN) and Core Network (CN) and its services.

SA5 has completed a study on management and orchestration on network slicing (3GPP 28.801 [13]) and started the normative specification work for release 15 based on this study. It is expected to be completed by the second quarter of 2018, including:

- Network slice concepts, use cases and requirements (3GPP 28.530 [15]);
- Provisioning of network slicing for 5G networks and services (3GPP 28.531[21]);
- Assurance data and performance management for 5G networks and network slicing;
- Fault supervision for 5G network and network slicing.

The following description highlights management and orchestration aspects of network slicing in 3GPP 28.801 [13]. However, these may be updated in the SA5 normative specifications based on the ongoing development of the SA2 technical specifications.

- General management and orchestration aspects of network slicing defined in 3GPP 28.801 [13]. Based on 3GPP 23.501 [14], SA5 has defined different management aspects for network slices in 3GPP 28.801 [13] as listed below:
    - Managing a complete Network Slice Instance (NSI) is not only managing all the functionalities but also the resources necessary to support certain set of communication services.
    - An NSI not only contains Network Functions (NFs), e.g belonging to AN and CN, but also the connectivity between the NFs. If the NFs are interconnected, the 3GPP management system contains the information relevant to connections between these NFs such as topology of

connections, individual link requirements (e.g. QoS attributes), etc. For the part of the Transport Network (TN) supporting connectivity between the NFs, the 3GPP management system provides link requirements to the management system that handles the part of the TN supporting connectivity between the NFs.
    - NSI can be composed of network slice subnets of physical network functions and/or virtualized network functions.
- Network Slice Instance lifecycle management. 3GPP 28.801 [13] has introduced the network slice instance lifecycle management as depicted below in Figure 20, considering it independent of the network service instance which is using the network slice instance. Typically, a network slice instance is designed (preparation phase), then it is instantiated (Instantiation, Configuration and Activation phase), then it is operated (Run Time phase) and finally it may be decommissioned when the slice is no longer needed (Decommissioning phase). 3GPP 28.801 [13] introduces 3 management logical functions:
    - Communication Service Management Function (CSMF): Responsible for translating the communication service related requirement to network slice related requirements.
    - Network Slice Management Function (NSMF): Responsible for management and orchestration of NSI and derive network slice subnet related requirements from network slice related requirements.
    - Network Slice Subnet Management Function (NSSMF): Responsible for management and orchestration of network slice subnet instances (NSSI).

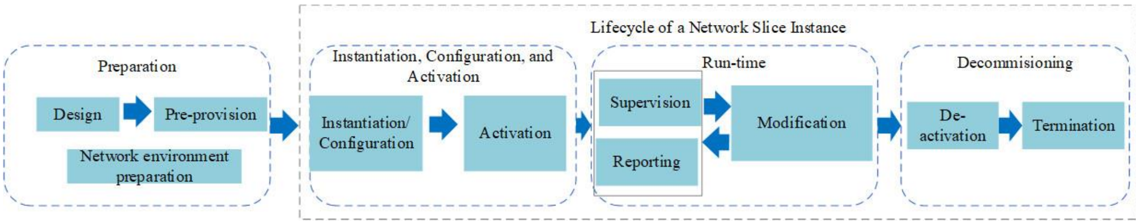


FIGURE 20: 3GPP VIEW ON NETWORK SLICE INSTANCE LIFECYCLE

11.2 ETSI

11.2.1 ETSI NFV

Work is also ongoing inside ETSI NFV on how the NFV architecture in general, but more specifically, the ETSI MANO components can support network slicing.

In this respect the Evolution and Ecosystem (EVE) working group has carried out activities that map NFV and 3GPP network slicing concepts (see EVE012 [40]). On the one hand, ETSI NFV EVE012 [40] establishes the correspondence between a network slice (3GPP) and a network service (ETSI NFV). There, ETSI describes that an NFV Network Service (NFV-NS) can be regarded as a resource-centric view of a network slice, for the cases where a NSI would contain at least one virtualized network function. According to 3GPP 28.801 [13], an NSSI can be shared by multiple NSIs. The virtualized resources for the slice subnet and their connectivity to physical resources can thus be represented by the nested network service concept defined in ETSI NFV-IFA 014 [38] (right hand side of Figure 21), or one or more VNFs and PNFs directly attached to the

Network Service used by the network slice. Figure 21 illustrates the relationship between 3GPP's network slice and ETSI NFV network service.

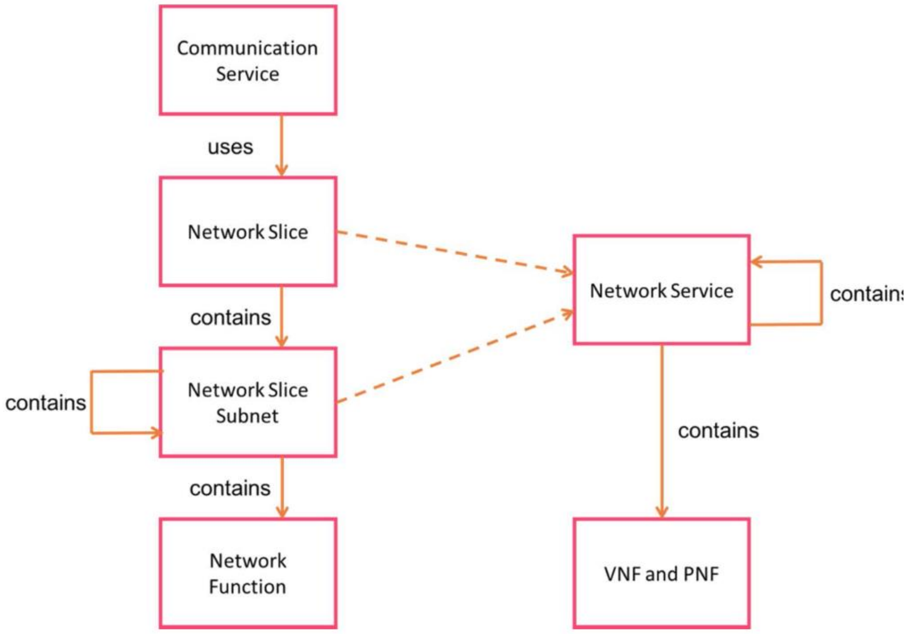


FIGURE 21: RELATION BETWEEN 3GPP AND ETSI INFORMATION MODELS (FROM [40])

As mentioned before, 3GPP 28.801 [13] identifies three management functions related to network slicing management: Communication Service Management Function (CSMF), Network Slice Management Function (NSMF), and Network Slice Subnet Management Function (NSSMF).

As shown in Figure 22, the Os-Ma-nfvo reference point can be used for the interaction between 3GPP slicing related management functions and NFV MANO. To properly interface with NFV MANO, the NSMF and/or NSSMF need to determine the type of network service or set of network services, VNF and PNF that can support the resource requirements for a NSI or NSSI, and whether new instances of these network services, VNFs and the connectivity to the PNFs need to be created or existing instances can be reused.

From a resource management viewpoint, NSI can be mapped to an instance of a simple or composite network service instance or to a concatenation of such network service instances. From a resource management viewpoint, different NSIs can use instances of the same type of network service (i.e. they are instantiated from the same Network Service Descriptor or NSD) with the same or different deployment flavours. Alternatively, different NSIs can use instances of different types of network services. The first approach can be used if the NSIs share the same types of network functions (or a large common subset) but differ in terms of the performance expected from these network functions (and from the virtual links connecting them) and/or the number of instances to be deployed for each of them. If slices differ more significantly, mapping to different network services, each with its own NSD can be considered. The same mapping principles might apply to NSSIs.

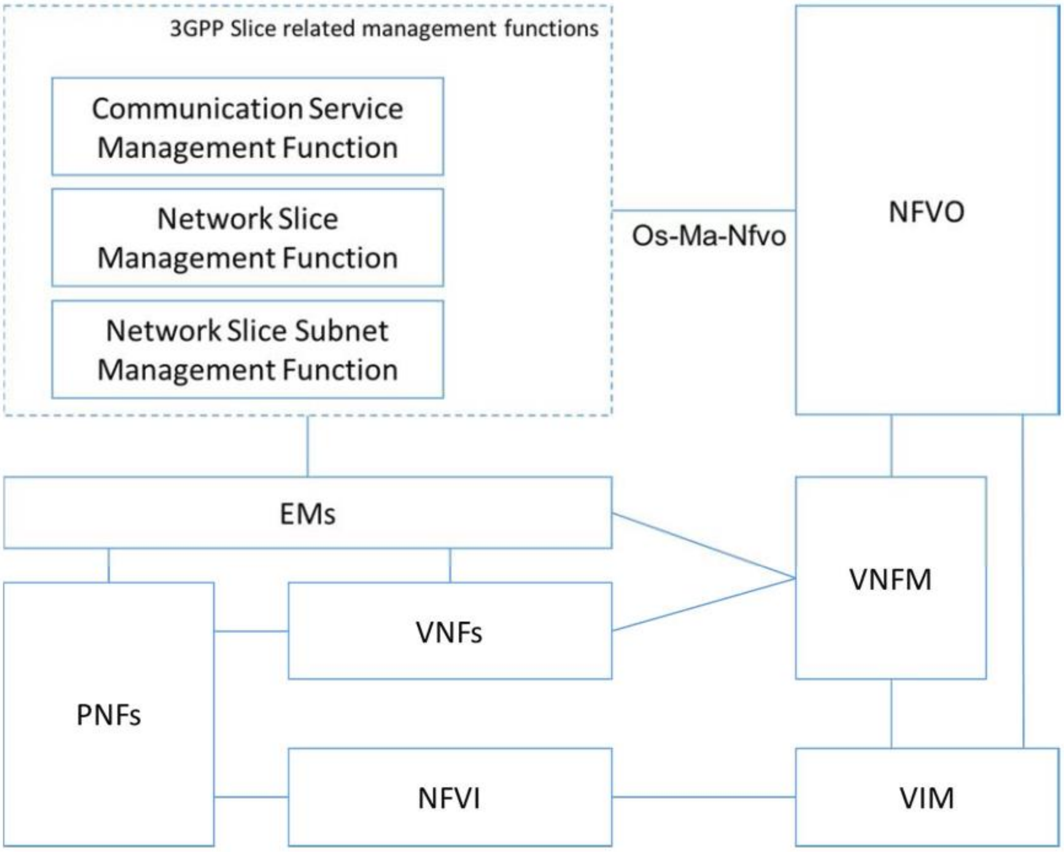


FIGURE 22: NETWORK SLICE MANAGEMENT IN AN NFV FRAMEWORK (FROM [40])

Also, as described before, 3GPP 28.801 [13] describes the lifecycle of a network slice, which is comprised of the four following phases: (i) Preparation; (ii) Instantiation, configuration and activation; (iii) Run-time; and (iv) Decommissioning.

The preparation phase includes the creation and verification of Network Slice Template(s) (NST(s)). From an NFV perspective, the resource requirement for a NST can be realized by one or more existing NSDs that have been previously on-boarded on the NFVO. The creation of a new NST can lead to requiring update of an existing NSD or generation of a new NSD followed by on-boarding the new NSD if the slice requirements do not map to an already on-boarded NSD. Indeed, the NFV-NS for the multiple NSIs may be instantiated with the same NSD, in order to deliver exactly the same optimizations and features but dedicated to different enterprise customers. On the other hand, a network slice intended to support totally new customer facing services is likely to require a new NS and thus the generation of a new NSD. The network slice instantiation step in the second phase triggers the instantiation of the underlying NSs. NFV-MANO functions are only involved in the network slice configuration phase if the configuration of virtualisation-related parameters is required on one or more of the constituent VNF instances. Configuration of the network applications embedded in the constituent network functions involves the NSMF or NSSMF and/or other parts of the OSS/BSS, and the element managers (if any) associated to these functions. NFV-MANO functions can be triggered during the network slice activation step. If explicit activation of VNFs is required, the NSMF or the NSSMF can change the operational state of those VNFs through an Update NFV-NS operation defined in ETSI NFV-IFA 013 [17]. The involvement of NFV-MANO in the run-time phase is limited to the operations related to the performance management, fault management, and lifecycle management of virtualised resources (e.g. scaling an underlying NFV-NS to expand a NSI). The decommissioning phase triggers the termination of the underlying network service instances.

Additionally, and given the multiple administrative boundaries of the 5G-TRANSFORMER architecture, the Interfaces and Architecture (IFA) working group is of particular interest for our project. ETSI NFV IFA028 [22] reports on potential architecture options to support the offering of NFV MANO services across multiple administrative domain. NFV-MANO services can be offered and consumed by different organizations, e.g. by different network operators or by different departments within the same network operator. Administrative domains as defined in ETSI NFV IFA010 [23] can be mapped to such different organizations. Examples of use cases for NFV-MANO service offerings across multiple administrative domains are described in ETSI NFV 001 [24]. Furthermore, ETSI NFV IFA022 [25] reports on the functional architecture needed to provision and manage multi-site network services. To this end, a set of multi-site use cases are studied.

Furthermore, compliance with widely accepted standards of the 5G-TRANSFORMER architecture is also relevant to maximize its impact. Therefore, in a more general architectural context than that defined by the previous documents (which focus on specific issues) the interfaces already defined in ETSI NFV MANO are also relevant:

- IFA013 [17] defines the interfaces supported over the Os-Ma-nfvo reference point of the NFV MANO architectural framework as well as the information elements exchanged over those interfaces;
- IFA005 [19] defines the interfaces supported over the Or-Vi reference point of the NFV MANO architectural framework as well as the information elements exchanged over those interfaces;
- IFA006 [43] defines the interfaces supported over the Vi-Vnfm reference point of the NFV MANO architectural framework as well as the information elements exchanged over those interfaces;

11.2.2 ETSI MEC

Multi-Access Edge Computing (MEC) is one of the key concepts for fulfilling some of the requirements of vertical services, and therefore its integration in the 5G-TRANSFORMER architecture is nexus in its design. MEC and its integration in an NFV context was studied in ETSI MEC017 [26] document and a reference architecture is provided with the following key observations:

- The mobile edge platform is deployed as a VNF and therefore the procedures defined by ETSI NFV for this means are used;
- ETSI NFV MANO sees mobile edge applications as regular VNFs allowing for reuse of ETSI MANO functionality (with perhaps some extensions);
- The virtualization infrastructure is deployed as a NFVI and its virtualized resources are managed by the VIM. For this purpose, the procedures defined by ETSI NFV infrastructure specifications, i.e. ETSI NFV INF003 [27], ETSI NFV INF004 [28] and ETSI NFV INF005 [29] can be used.

12 Annex III: Notation for Requirements

In this deliverable we follow - with slight adaptions - the notation for requirements used already in [1]. For each requirement, the following fields should be provided:

ID	Requirement	Ref	F/NF
ReqX.XX	e.g. The vehicle shall be connected to a 5G router	XX.XX	F/NF

The meanings of the fields are as follows:

- **ID**: is the identifier of the requirement (written in the form ReqX.XX).
- **Requirement**: a complete sentence explaining the requirement.
- **Ref**: Reference to the source of the requirement, e.g. a use case, a statement in a standard, etc.
- **F/NF**: if the requirement is Functional (F) or Non Functional (NF).

NOTE: The requirement field is written following the approach followed by IETF documents, included next. The key words “must”, “must not”, “required”, “shall”, “shall not”, “should”, “should not”, “recommended”, “may” and “optional” in this document are to be interpreted as described in [44].

1. **MUST** This word, or the terms “**REQUIRED**” or “**SHALL**”, mean that the definition is an absolute requirement of the specification.
2. **MUST NOT** This phrase, or the phrase “**SHALL NOT**”, mean that the definition is an absolute prohibition of the specification.
3. **SHOULD** This word, or the adjective “**RECOMMENDED**”, mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighted before choosing a different course.
4. **SHOULD NOT** This phrase, or the phrase “**NOT RECOMMENDED**” mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
5. **MAY** This word, or the adjective “**OPTIONAL**”, mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein, an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option (except, of course, for the feature the option provides.).

13 Annex IV: Generic Requirements on Vertical Service Management

This annex describes the generic requirements of network services defined in ETSI NFV IFA 013 [17], adapted to the 5GT-VS.

TABLE 22: VSD MANAGEMENT REQUIREMENTS, [17], SECTION 5.3.1

ID	Requirement	Ref	F/NF
ReqVS.Vsd.01	The 5GT-VS shall support onboarding an VSD[13].	[17], Os-Ma-nfvo.Nsd.001	F
ReqVS.Vsd.02	void.	[17], Os-Ma-nfvo.Nsd.002	F
ReqVS.Vsd.03	void.	[17], Os-Ma-nfvo.Nsd.003	F
ReqVS.Vsd.04	The 5GT-VS shall support updating an VSD.	[17], Os-Ma-nfvo.Nsd.004	F
ReqVS.Vsd.05	The 5GT-VS shall support querying an VSD.	[17], Os-Ma-nfvo.Nsd.005	F
ReqVS.Vsd.06	The 5GT-VS shall support deleting an VSD.	[17], Os-Ma-nfvo.Nsd.006	F
ReqVS.Vsd.07	The 5GT-VS shall support providing notifications about the onboarding of VSDs.	[17], Os-Ma-nfvo.Nsd.007	F
ReqVS.Vsd.08	The 5GT-VS shall support providing notifications as a result of changes on VSD states.	[17], Os-Ma-nfvo.Nsd.008	F
ReqVS.Vsd.09	void.	[17], Os-Ma-nfvo.Nsd.009	F
ReqVS.Vsd.10	void.	[17], Os-Ma-nfvo.Nsd.010	F
ReqVS.Vsd.11	void.	[17], Os-Ma-nfvo.Nsd.011	F
ReqVS.Vsd.12	void.	[17], Os-Ma-nfvo.Nsd.012	F
ReqVS.Vsd.13	The 5GT-VS shall support subscribing to notifications related to NSD management changes.	[17], Os-Ma-nfvo.Nsd.013	F

[13] A corresponding requirement for VSBs is defined in Section 3.2.1, ReqVS.Di.13.

TABLE 23: VS LIFECYCLE MANAGEMENT REQUIREMENTS, [17], SECTION 5.3.2

ID	Requirement	Ref	F/NF
ReqVS.Lcm.01	The 5GT-VS shall support instantiating a VSI.	[17], Os-Ma-nfvo.NsLcm.001	F
ReqVS.Lcm.02	The 5GT-VS shall support terminating a VSI.	[17], Os-Ma-nfvo.NsLcm.002	F
ReqVS.Lcm.03	The 5GT-VS shall support querying a VSI.	[17], Os-Ma-nfvo.NsLcm.003	F
ReqVS.Lcm.04	The 5GT-VS shall support scaling a VSI.	[17], Os-Ma-nfvo.NsLcm.004	F
ReqVS.Lcm.05	The 5GT-VS shall support updating a VSI.	[17], Os-Ma-nfvo.NsLcm.005	F
ReqVS.Lcm.06	The 5GT-VS shall support creating a classification and selection rule for the existing NFP instance.	[17], Os-Ma-nfvo.NsLcm.006	F
ReqVS.Lcm.07	The 5GT-VS system shall support updating the classification and selection rule for the existing NFP instance.	[17], Os-Ma-nfvo.NsLcm.007	F
ReqVS.Lcm.08	The 5GT-VS shall support instantiating a VSI which includes existing VA instance(s).	[17], Os-Ma-nfvo.NsLcm.008	F
ReqVS.Lcm.09	The 5GT-VS shall support updating a VSI which includes existing VNF and VA instance(s).	[17], Os-Ma-nfvo.NsLcm.009	F
ReqVS.Lcm.10	The 5GT-VS shall support healing a VSI.	[17], Os-Ma-nfvo.NsLcm.010	F
ReqVS.Lcm.11	The 5GT-VS shall support sharing a VNF, VA, or PNF instance or a nested NFV-NS instance between multiple VSIs.	[17], Os-Ma-nfvo.NsLcm.011	F
ReqVS.Lcm.12	The 5GT-VS shall support instantiating a VNF or VA instance explicitly as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.012	F
ReqVS.Lcm.13	The 5GT-VS shall support adding/removing an existing VNF or VA instance to/from a VSI as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.013	F
ReqVS.Lcm.14	The 5GT-VS shall support scaling a VNF or VA instance explicitly as part of the scaling of a VS.	[17], Os-Ma-nfvo.NsLcm.014	F
ReqVS.Lcm.15	The 5GT-VS shall support querying information about a VNF or VA instance as part of the query of a VS.	[17], Os-Ma-nfvo.NsLcm.015	F
ReqVS.Lcm.16	The 5GT-VS shall support healing a VNF or VA instance explicitly as part of the healing of a NFV-NS.	[17], Os-Ma-nfvo.NsLcm.016	F
ReqVS.Lcm.17	The 5GT-VS shall support changing the state of a VNF or VA instance explicitly as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.017	F
ReqVS.Lcm.18	The 5GT-VS shall support changing the deployment flavour (DF) of a VNF or VA instance explicitly as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.018	F

ReqVS.Lcm.19	The 5GT-VS shall support modifying information and/or the configuration parameters of a VNF or VA instance explicitly as part of the update of an VS.	[17], Os-Ma-nfvo.NsLcm.019	F
ReqVS.Lcm.20	void		
ReqVS.Lcm.21	The 5GT-VS shall support providing additional affinity or anti-affinity rules when instantiating a VS.	[17], Os-Ma-nfvo.NsLcm.021	F
ReqVS.Lcm.22	void	[17], Os-Ma-nfvo.NsLcm.022	
ReqVS.Lcm.23	The 5GT-VS shall support adding/removing an existing nested VSI to/from an VSI explicitly as part of the update of a VS	[17], Os-Ma-nfvo.NsLcm.023	F
ReqVS.Lcm.24	The 5GT-VS shall support adding new SAP to a VS and removing existing SAP from a VS explicitly as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.024	F
ReqVS.Lcm.25	The 5GT-VS shall support associating a new VSD version to an existing VSI explicitly as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.025	F
ReqVS.Lcm.26	void	[17], Os-Ma-nfvo.NsLcm.026	
ReqVS.Lcm.27	The 5GT-VS shall support adding a new VNFFG to a VSI, remove existing VNFFG and updating a VNFFG from a VSI explicitly as part of the update of a VS.	[17], Os-Ma-nfvo.NsLcm.027	F
ReqVS.Lcm.28	The 5GT-VS shall support querying the status of a VS LCM operation.	[17], Os-Ma-nfvo.NsLcm.028	F
ReqVS.Lcm.29	The 5GT-VS shall support providing to the vertical and the OSS/BSS notifications about changes of a VSI that are related to VS LCM operations.	[17], Os-Ma-nfvo.NsLcm.029	F
ReqVS.Lcm.30	The 5GT-VS shall provide notifications which contain information about the type of the VS lifecycle change, the addition/deletion/modification of VNFs, VAs and/or PNFs) about change in the connectivity between elements of the VS.	[17], Os-Ma-nfvo.NsLcm.030	F
ReqVS.Lcm.31	The 5GT-VS shall provide notifications which contain information about the VLs and VNFFGs that are added/modified/deleted as part of the VS lifecycle operation.	[17], Os-Ma-nfvo.NsLcm.031	F
ReqVS.Lcm.32	The 5GT-VS shall support notifying the result (successful or failed) of VS instantiation with indicating the VS instance Id.	[17], Os-Ma-nfvo.NsLcm.032	F
ReqVS.Lcm.33	The 5GT-VS shall support providing to the vertical and the OSS/BSS notifications about creation and deletion of an VS instance identifier and the associated	[17], Os-Ma-nfvo.NsLcm.033	F

	instance of a VS information element, further referred to as VS identifier creation/deletion notifications.		
ReqVS.Lcm.34	The 5GT-VS shall support subscribing to VS lifecycle change notifications and to VS identifier creation/deletion notifications.	[17], Os-Ma-nfvo.NsLcm.034	F
ReqVS.Lcm.35	The 5GT-VS shall support creating a VS instance identifier and the associated instance of an VS information element.	[17], Os-Ma-nfvo.NsLcm.035	F
ReqVS.Lcm.36	The 5GT-VS shall support changing the external connectivity of a VNF or VA instance explicitly as part of the update of a VSI.	[17], Os-Ma-nfvo.NsLcm.036	F
ReqVS.Lcm.37	The 5GT-VS shall support the capability to invoke VS error handling operation(s) after the VS life cycle operation occurrence fails.	[17], Os-Ma-nfvo.NsLcm.037	F

TABLE 24: VS PERFORMANCE MANAGEMENT REQUIREMENTS, [17], SECTION 5.3.4

ID	Requirement	Ref	F/NF
ReqVS.Pm.01	The 5GT-VS shall enable the vertical and the OSS/BSS to control the collection and reporting of performance information for VSs.	[17], Os-Ma-nfvo.Pm.001	F
ReqVS.Pm.02	The 5GT-VS shall support the capability to notify the availability of performance information.	[17], Os-Ma-nfvo.Pm.002	F
ReqVS.Pm.03	The 5GT-VS shall expose the type of performance information that the 5GT-SO can collect from the VSs.	[17], Os-Ma-nfvo.Pm.003	F
ReqVS.Pm.04	The 5GT-VS shall support enable the vertical and the OSS/BSS to create a PM job specifying the type of resource(s) and performance information that the vertical or the OSS/BSS requires.	[17], Os-Ma-nfvo.Pm.004	F
ReqVS.Pm.05	The 5GT-VS shall enable the vertical and the OSS/BSS to create a PM job specifying the granularity for collection and reporting of performance information on VSs.	[17], Os-Ma-nfvo.Pm.005	F
ReqVS.Pm.06	The 5GT-VS shall enable the vertical and the OSS/BSS to delete one or more explicitly identified PM job(s).	[17], Os-Ma-nfvo.Pm.006	F
ReqVS.Pm.07	The 5GT-VS shall support periodic collection of performance information (bounded or unbounded).	[17], Os-Ma-nfvo.Pm.007	F
ReqVS.Pm.08	The 5GT-VS shall support the grouping of measurements.	[17], Os-Ma-nfvo.Pm.008	F
ReqVS.Pm.09	The 5GT-VS shall enable the vertical and the OSS/BSS to manage the thresholds on the performance information collected by the 5GT-SO.	[17], Os-Ma-nfvo.Pm.009	F

ReqVS.Pm.10	The 5GT-VS shall support the capability to notify about a threshold defined for a specified metric of VSs being crossed.	[17], Os-Ma-nfvo.Pm.010	F
ReqVS.Pm.11	The 5GT-VS shall enable the vertical and the OSS/BSS to receive notifications related to threshold crossing.	[17], Os-Ma-nfvo.Pm.011	F
ReqVS.Pm.12	The 5GT-VS should support querying the list of active PM jobs and defined threshold conditions by the consumer entity that created them.	[17], Os-Ma-nfvo.Pm.012	F
ReqVS.Pm.13	The 5GT-VS shall support the deletion of threshold conditions on the performance information collected by the 5GT-SO.	[17], Os-Ma-nfvo.Pm.013	F

TABLE 25: VS FAULT MANAGEMENT REQUIREMENTS, [17], SECTION 5.3.5

ID	Requirement	Ref	F/NF
ReqVS.Fm.01	The 5GT-VS shall support collecting VSs fault information.	[17], Os-Ma-nfvo.Fm.001	F
ReqVS.Fm.02	The 5GT-VS shall support providing alarm notifications related to faults on VSs to the vertical and the OSS/BSS.	[17], Os-Ma-nfvo.Fm.002	F
ReqVS.Fm.03	The 5GT-VS shall support providing a notification when there is a change in the alarm information on VS.	[17], Os-Ma-nfvo.Fm.003	F
ReqVS.Fm.04	The 5GT-VS shall support the sending of notification to the vertical and the OSS/BSS when an alarm on a VS has been created.	[17], Os-Ma-nfvo.Fm.004	F
ReqVS.Fm.05	The 5GT-VS shall support the sending of notification to the vertical and the OSS/BSS when an alarm on an VS has been cleared.	[17], Os-Ma-nfvo.Fm.005	F
ReqVS.Fm.06	The 5GT-VS shall allow unambiguous identification of the alarm on a VS sent to the vertical or the OSS/BSS.	[17], Os-Ma-nfvo.Fm.006	F
ReqVS.Fm.07	The 5GT-VS shall allow unambiguous identification of the VS causing the alarm.	[17], Os-Ma-nfvo.Fm.007	F
ReqVS.Fm.08	The 5GT-VS shall allow unambiguous identification of the alarm cause.	[17], Os-Ma-nfvo.Fm.008	F

TABLE 26: VNF PACKAGE MANAGEMENT REQUIREMENTS, [17], SECTION 5.3.6

ID	Requirement	Ref	F/NF
ReqVS.VP.01	The 5GT-VS shall support on-boarding a VNF Package.	[17], Os-Ma-nfvo.VnfPkgm.001	F
ReqVS.VP.02	The 5GT-VS shall support disabling a VNF Package.	[17], Os-Ma-nfvo.VnfPkgm.002	F
ReqVS.VP.03	The 5GT-VS shall support enabling a VNF Package.	[17], Os-Ma-nfvo.VnfPkgm.003	F
ReqVS.VP.04	The 5GT-VS shall support querying VNF Package information.	[17], Os-Ma-nfvo.VnfPkgm.004	F

ReqVS.VP.05	The 5GT-VS shall support deleting a VNF Package.	[17], Os-Ma-nfvo.VnfPkgm.005	F
ReqVS.VP06	The 5GT-VS shall support providing notifications about the on-boarding of VNF Packages.	[17], Os-Ma-nfvo.VnfPkgm.006	F
ReqVS.VP.07	The 5GT-VS shall support providing notifications as a result of changes on VNF Package states.	[17], Os-Ma-nfvo.VnfPkgm.007	F
ReqVS.VP.08	The 5GT-VS shall support fetching a VNF Package, or selected artifacts contained in a package.	[17], Os-Ma-nfvo.VnfPkgm.008	F
ReqVS.VP.09	The 5GT-VS shall support aborting the pending deletion of a VNF Package as long as a VNF instance exists that is based on this package.	[17], Os-Ma-nfvo.VnfPkgm.009	F

14 Annex V: Vertical Slicer NBI Operations

The operations at the 5GT-VS NBI, see Section 5.7, are listed in the following subsections together with their parameters.

14.1 Ve-Vs reference point

14.1.1 Query VS blueprints

This operation is described in Section 5.7.1.1, see Table 9.

14.1.2 Create VSD

This operation allows a vertical to create a new VSD that is added to the 5GT-VS catalogue. The VSD is then used by the vertical to request the instantiation of a new vertical service.

The Create VSD messages are specified in Table 27.

TABLE 27: CREATE VSD MESSAGES

Message	Direction	Description	Parameters
Create VSD request	Vertical → 5GT-VS	Request to create a new VSD.	• VSD (format specified in Section 6.2). • Vertical ID. • Is Public (set to true if the VSD is visible to other verticals).
Create VSD response	5GT-VS → Vertical	Response with the result of the operation.	• Result Code. • VSD ID.

14.1.3 Query VSD

This operation allows a vertical to retrieve one or more VSDs from the 5GT-VS catalogue, based on the given filter. A vertical can retrieve only the VSDs that have been created by the vertical itself or the VSDs that are declared as public.

The Query VSD messages are specified in Table 28.

TABLE 28: QUERY VSD MESSAGES

Message	Direction	Description	Parameters
Query VSD request	Vertical → 5GT-VS	Request to retrieve one or more VSDs matching the given filter.	• Filter (e.g. VSD ID). • Vertical ID.
Query VSD response	5GT-VS → Vertical	Response including the details of the requested VSDs.	• Result Code. • List<VSD> (format specified in Section 6.2.

14.1.4 Update VSD

This operation allows a vertical to update a VSD from the 5GT-VS catalogue, providing their IDs. A vertical is allowed to update only the VSDs that have been created by the vertical itself.

The Update VSD messages are specified in Table 29. The updated VSD has a new VSD ID, the previous version is still accessible with the old VSD ID.

TABLE 29: UPDATE VSD MESSAGES

Message	Direction	Description	Parameters
Update VSD request	Vertical → 5GT-VS	Request to update a VSD	• VSD (format specified in Section 6.2). • Vertical ID. • VSD ID. • Is Public (set to true if the VSD is visible to other verticals).
Update VSD response	5GT-VS → Vertical	Response with the result of the operation.	• Result Code. • VSD ID.

14.1.5 Delete VSD

This operation allows a vertical to delete one or more VSDs from the 5GT-VS catalogue, providing their IDs. A vertical is allowed to delete only the VSDs that have been created by the vertical itself.

The Delete VSD messages are specified in Table 30.

TABLE 30: DELETE VSD MESSAGES

Message	Direction	Description	Parameters
Delete VSD request	Vertical → 5GT-VS	Request to delete one or more VSDs with the given IDs.	• List<VSD ID>. • Vertical ID.
Delete VSD response	5GT-VS → Vertical	Response with the result of the operation.	• Result Code.

14.1.6 Instantiate VSI

This operation allows a vertical to instantiate a new VSI, given its VSD. Upon the reception of an Instantiate VSI request, the 5GT-VS creates a new entry in its repository, generating a unique VSI ID that is immediately returned to the vertical. This VSI ID can then be used by the vertical to retrieve information about the status and the characteristics of the VSI, using the Query VSI primitive.

The procedures to actually instantiate the VSI are then handled by the 5GT-VS in an asynchronous manner and notifications about the instantiation result, as well as any other relevant lifecycle event, are sent later to the vertical, if a suitable URL was provided in the instantiation request. The corresponding workflow is described in Section 7.3.

The Instantiate VSI messages are specified in Table 31.

TABLE 31: INSTANTIATE VSI MESSAGES

Message	Direction	Description	Parameters
Instantiate VSI request	Vertical → 5GT-VS	Request to instantiate a new VSI based on the specified VSD.	• VSI name. • VSI description. • VSD ID. • Vertical ID. • Notification URL (URL where the vertical wants to receive notifications about the lifecycle or failure events of the VSI). • Additional parameters (e.g. configuration parameters provided by the vertical).
Instantiate VSI response	5GT-VS → Vertical	Response with the preliminary result of the operation and, if successful, the VSI ID.	• Result Code. • VSI ID.

14.1.7 Query VSI

This operation allows a vertical to retrieve the information about an existing VSI previously requested by the vertical itself. The target VSI is identified through its unique ID, as returned by the 5GT-VS in the Instantiate VSI Response. This method can be used by the vertical to get information about the current status of the VSI, its characteristics (through a reference to the current VSD) and the value assigned by the system to the output parameters, when they become available.

The Query VSI messages are specified in Table 32.

TABLE 32: QUERY VSI MESSAGES

Message	Direction	Description	Parameters
Query VSI request	Vertical → 5GT-VS	Request to retrieve information about the VSI matching the given ID.	• VSI ID. • Vertical ID.
Query VSI response	5GT-VS → Vertical	Response including the details of the requested VSI.	• Result Code. • VSI ID. • VSI name. • VSI description. • VSD ID. • VSI status (i.e. instantiating, instantiated, under modification, terminating, terminated, failed).

			• External interconnections (SAPs including the values assigned by the system to IP addresses, VLAN IDs, etc.). • Internal interconnections (CPs including the values assigned by the system to IP addresses, VLAN IDs, etc.).

14.1.8 Terminate VSI

This operation allows a vertical to terminate an existing VSI previously requested by the vertical itself. The target VSI is identified through its unique ID, as returned by the 5GT-VS in the Instantiate VSI Response. The 5GT-VS returns synchronously a preliminary result of the operation, but the procedure to actually terminate the VSI is handled by the 5GT-VS in an asynchronous manner. A notification about the termination result is sent to the vertical when the termination process is finished, if a suitable URL was provided by the vertical in the instantiation request. The corresponding workflow is described in Section 7.5.

The Terminate VSI messages are specified in Table 33.

TABLE 33: TERMINATE VSI MESSAGES

Message	Direction	Description	Parameters
Terminate VSI request	Vertical → 5GT-VS	Request to terminate a VSI matching the given ID.	• VSI ID. • Vertical ID.
Terminate VSI response	5GT-VS → Vertical	Response with the preliminary result of the operation.	• Result Code.

14.1.9 Modify VSI

This operation allows a vertical to modify an already instantiated VSI, providing a new VSD. The procedure to actually modify the VSI is handled by the 5GT-VS in an asynchronous manner and, when the procedure is completed, a notification about the modification result is sent to the vertical, if a suitable URL was provided by the vertical in the instantiation request. The corresponding workflow is described in Section 7.4.

The Modify VSI messages are specified in Table 34.

TABLE 34: MODIFY VSI MESSAGES

Message	Direction	Description	Parameters
Modify VSI request	Vertical → 5GT-VS	Request to modify a VSI based on a new VSD.	• VSI ID. • VSD ID (reference to the new VSD). • Vertical ID.

Modify VSI response	5GT-VS → Vertical	Response with the preliminary result of the operation.	• Result Code.

14.1.10 Notify VSI lifecycle event

This operation allows the 5GT-VS to inform the vertical about lifecycle events related to VSIs owned by the vertical itself. These notifications are asynchronous and can report the results of actions triggered by the vertical (e.g. the result of the instantiation, termination or modification procedures) or events triggered by an autonomous decision of the 5GT-VS (e.g. scaling in and out of the VSI as consequence of arbitrator's decision about a contention among vertical services with different priorities).

The Notify VSI lifecycle message is specified in Table 35.

TABLE 35: NOTIFY VSI LIFECYCLE EVENT MESSAGES

Message	Direction	Description	Parameters
Notify VSI lifecycle event	5GT-VS → Vertical	Notification of a lifecycle event.	• VSI ID. • LC notification type (e.g. instantiation result, termination result, modification result, scaling result). • Result Code. • VSD (only for notifications about scaling results, it indicates the new VSD currently associated to the VSI).

14.1.11 Query VSI monitoring parameter

This operation allows a vertical to retrieve monitoring parameters about an existing VSI previously requested by the vertical itself. The requested monitoring parameters should be included in the ones specified in the original VSD.

The Query VSI monitoring parameter messages are specified in Table 36.

TABLE 36: QUERY VSI MONITORING PARAMETER MESSAGES

Message	Direction	Description	Parameters
Query VSI monitoring parameter request	Vertical → 5GT-VS	Request to retrieve monitoring parameters about the VSI matching the given ID.	• VSI ID. • Vertical ID. • List<Monitoring Parameter>.
Query VSI monitoring parameter response	5GT-VS → Vertical	Response including the details of the requested VSI.	• Map<Monitoring Parameter; Value>. • Timestamp.

14.1.12 Subscription/Notification about VSI monitoring parameters

This operation allows a vertical to subscribe with the 5GT-VS to receive notifications about monitoring parameters, e.g. providing rules to describe thresholds for which notifications are generated when crossed or a time interval for periodical notifications.

The Subscription/Notification messages about VSI monitoring parameters are specified in Table 37.

TABLE 37: SUBSCRIBE/NOTIFY VSI MONITORING PARAMETERS MESSAGES

Message	Direction	Description	Parameters
Create VSI monitoring subscription request	Vertical → 5GT-VS	Request to create a new subscription for receiving notifications about monitoring parameters.	• VSI ID. • Vertical ID. • Map<Monitoring Parameter; Threshold>. • Time interval.
Create VSI monitoring subscription response	5GT-VS → Vertical	Response including the ID of the subscription.	• Result Code. • Subscription ID.
Delete VSI monitoring subscription request	Vertical → 5GT-VS	Request to delete a previous subscription.	• Subscription ID.
Delete VSI monitoring subscription response	5GT-VS → Vertical	Response with the result of the operation.	• Result Code.
Notify VSI monitoring parameter	5GT-VS → Vertical	Notification about a new monitoring parameter or threshold crossed event	• VSI ID. • Map<Monitoring Parameter; Value> • Timestamp.

14.1.13 Notify VSI failure event

This operation allows the 5GT-VS to inform the vertical about failures related to VSIs owned by the vertical itself. These notifications are generated asynchronously by the 5GT-VS and they are sent to the URL provided by the vertical in the instantiation request.

The Notify VSI failure event message is specified in Table 38.

TABLE 38: NOTIFY VSI FAILURE MESSAGES

Message	Direction	Description	Parameters
Notify VSI failure event	5GT-VS → Vertical	Notification of a failure event.	• VSI ID. • LC notification type (e.g. instantiation result, termination result, modification result, scaling result). • Result Code.

			• VSD (only for notifications about scaling results, it indicates the new VSD currently associated to the VSI).

14.2 Mgt-Vs reference point

14.2.1 Create tenant

This operation allows the system administrator to create a new tenant in the 5GT-VS repository. A tenant corresponds to a TSC (i.e. a vertical or an M(V)NO) who has established a business relationship with the TSP owning the 5GT-VS. Each tenant will be able to request VSIs in the limit of the SLAs signed between customer and provider; for this reason, the tenant information stored in the 5GT-VS repository will be used to validate and authorize the requests received on the Ve-Vs reference point for management of VSDs and VSIs. Tenant groups are used to define the sharing of services on a given slice, e.g. there could be a group of ‘car manufacturers’.

The Create tenant messages are specified in Table 39.

TABLE 39: CREATE TENANT MESSAGES

Message	Direction	Description	Parameters
Create tenant request	Mgt → 5GT-VS	Request to create a new tenant.	• Tenant ID. • Tenant name. • Tenant credential. • List<Group ID>.
Create tenant response	5GT-VS → Mgt	Response with the result of the operation.	• Result Code.

14.2.2 Query tenant

This operation allows the system administrator to retrieve all the information associated with an existing tenant, including the IDs of the VSDs created by and the VSIs instantiated for the tenant, the IDs of the active SLAs, and the total amount of resources currently allocated for the VSIs belonging to the tenant. Detailed information about each VSD, VSI and SLA can then be obtained through the related query methods.

The Query tenant messages are specified in Table 40.

TABLE 40: QUERY TENANT MESSAGES

Message	Direction	Description	Parameters
Query tenant request	Mgt → 5GT-VS	Request to retrieve the details of a given tenant.	• Tenant ID.
Query tenant response	5GT-VS → Mgt	Response with the information associated to the given tenant.	• Tenant ID. • Tenant name. • Tenant credential. • List<Group ID>. • List<SLA ID>.

			• List<VSD ID>. • List<VSI ID>. • Total allocated resources.

14.2.3 Delete tenant

This operation allows the system administrator to delete an existing tenant. A tenant can be removed from the system only when all its VSIs have been terminated.

The Delete tenant messages are specified in Table 41.

TABLE 41: DELETE TENANT MESSAGES

Message	Direction	Description	Parameters
Delete tenant request	Mgt → 5GT-VS	Request to delete a tenant with the given ID.	• Tenant ID.
Delete tenant response	5GT-VS → Mgt	Response with the result of the operation.	• Result Code.

14.2.4 Create SLA

This operation allows the system administrator to create a new SLA for an existing tenant in the 5GT-VS repository. A non-exhaustive list of information that can be included in an SLA includes the maximum amount of resources (e.g. computing vCPUs or RAM or disk, storage space, optionally distinguished between cloud and MEC resources) that can be allocated for a given tenant or the maximum number of VSIs that can be requested by the given tenant. The SLA information is used by the 5GT-VS to validate the requests from the tenants and, in particular, by the Arbitrator to identify contentions among vertical services belonging to the same customer.

The Create SLA messages are specified in Table 42.

TABLE 42: CREATE SLA MESSAGES

Message	Direction	Description	Parameters
Create SLA request	Mgt → 5GT-VS	Request to create a new SLA for the given tenant.	• Tenant ID. • List<SLA constraint[14]>. • SLA Status (enabled, disabled).
Create SLA response	5GT-VS → Mgt	Response with the result of the operation and the SLA ID.	• Result Code. • SLA ID.

14.2.5 Query SLA

This operation allows the system administrator to retrieve the information about an SLA, given its ID.

The Query SLA messages are specified in Table 43.

[14] The format of an SLA constraint is left for further specification during the implementation phase.

TABLE 43: QUERY SLA MESSAGES

Message	Direction	Description	Parameters
Query SLA request	Mgt → 5GT-VS	Request to retrieve the details of a given SLA.	• SLA ID.
Query SLA response	5GT-VS → Mgt	Response with the information associated to the given SLA.	• SLA ID. • Tenant ID. • List<SLA constraint>. • SLA Status (enabled, disabled).

14.2.6 Delete SLA

This operation allows the system administrator to delete an existing SLA.

The Delete SLA messages are specified in Table 44.

TABLE 44: DELETE SLA MESSAGES

Message	Direction	Description	Parameters
Delete SLA request	Mgt → 5GT-VS	Request to delete an SLA with the given ID.	• SLA ID.
Delete SLA response	5GT-VS → Mgt	Response with the result of the operation.	• Result Code.

14.2.7 Modify SLA

This operation allows the system administrator to modify an existing SLA, e.g. to change its status enabling or disabling it or to update its content (amount of resources, etc.).

The Modify SLA messages are specified in Table 45.

TABLE 45: MODIFY SLA MESSAGES

Message	Direction	Description	Parameters
Modify SLA request	Mgt → 5GT-VS	Request to modify an SLA with the given ID.	• SLA ID. • List<SLA constraint>. • SLA Status (enabled, disabled).
Modify SLA response	5GT-VS → Mgt	Response with the result of the operation.	• Result Code.

14.2.8 Create VS blueprint

This operation allows the system administrator to on-board a new VSB in the 5GT-VS catalogue, so that it can be used as a baseline by the verticals to create their own VSDs. A VSB is typically designed (off-line) together with the NSDs and the VNF packages or MEC application descriptors (AppDs) that describe the NFV network services able to implement the vertical service defined in the blueprint itself. This design phase includes also the definition of suitable “translation rules” between VSDs that can be derived from the blueprint and the corresponding NSD deployment flavours, i.e. translation rules between QoS-like parameters and resource-like parameters. For this reason, a VSB is typically on-boarded together with the corresponding NSDs, VNF packages, AppDs and translation rules.

The Create VS blueprint messages are specified in Table 46.

TABLE 46: CREATE VS BLUEPRINT MESSAGES

Message	Direction	Description	Parameters
Create VS blueprint request	Mgt → 5GT-VS	Request to create a new VSB.	• VSB (format specified in Section 6.1). • List<NSD> (format specified in Section 6.3). • List<VNF Package>. • List<AppD>. • List<Translation Rule[15]>.
Create VS blueprint response	5GT-VS → Mgt	Response with the result of the operation and the VSB ID generated by the 5GT-VS.	• Result Code. • VSB ID.

14.2.9 Query VS blueprint

This operation allows the system administrator to retrieve the information about one or more VSBs, based on a filter.

The Query VS blueprint messages are specified in Table 47.

TABLE 47: QUERY VS BLUEPRINT MESSAGES

Message	Direction	Description	Parameters
Query VS blueprint request	Mgt → 5GT-VS	Request to retrieve one or more VSBs matching the given filter.	• Filter (e.g. VSB ID, …).
Query VS blueprint response	5GT-VS → Mgt	Response including the details of the requested VSBs.	• List<VS Blueprint> (format specified in Section 6.1).

14.2.10 Delete VS blueprint

This operation allows the system administrator to delete an existing VSB.

The Delete VS blueprint messages are specified in Table 48.

TABLE 48: DELETE VS BLUEPRINT MESSAGES

Message	Direction	Description	Parameters
Delete VS blueprint request	Mgt → 5GT-VS	Request to delete a VSB with the given ID.	• VSB ID.
Delete VS blueprint response	5GT-VS → Mgt	Response with the result of the operation.	• Result Code.

[15] The format of a translation rule is left for further specification during the implementation phase.

15 Annex VI: Network service descriptors

In this annex we describe the most relevant fields of NSD information elements, which have not been modified by 5G-TRANSFORMER.

15.1 Virtual Network Function Descriptor (VNFD)

This descriptor encapsulates all the information necessary to specify the characteristics of a VNF such as which software image is going to use, the resources it will use and other properties that help in the deployment of the VNF.

TABLE 49: VNFD INFORMATION ELEMENT

Field	Description
vnfId	Unique identifier for the VNFD.
image	Software image that contain the VNF to be virtually deployed.
virtualCPU	CPU resources used by the VNF.
virtualDisk	Disk resources for the VNF.
virtualMemory	RAM resources used by the VNF.
cpds	Connection point descriptors used to connect with the VNF.
LFCMS	Scripts used for the life cycle management of the VNF. This is a list of pairs <eventType, script> to specify how to deal with different life cycle events.
indicators	List of monitoring indicators used for this VNF, such as CPU usage, memory pressure, etc.

15.2 Physical Network Function Descriptor (PNFD)

A NFV-NS can make use of already existing NFs that are deployed in bare metal infrastructure. Information related to those NFs is not specified by the NSD, hence the PNFD must provide information on how that PNF can be connected in the VNFFG and what is its functionality.

TABLE 50: PNFD INFORMATION ELEMENT

Field	Description
PNFD_id	Unique identifier for a PNFD descriptor.
CPDs	List of connection points descriptors that can be used to attach to the PNF.
description	A string describing the PNF functionality.

15.3 Connection Point Descriptors (CPDs)

Every PNF and VNF have connection points that are interconnected with Virtual Links (VLs), these connection points are entities at its own that are part of the NSD.

TABLE 51: CPD INFORMATION ELEMENT

Field	Description
CPD_id	Unique identifier for a CPD.
protocol	The communication protocols used in the link: Ethernet, IPv4, IPv6, etc.
address	L2 or L3 address used to reach the CP.
description	A string describing the use of the CP.

15.4 Service Access Point Descriptor (SAPD)

It is a CPD used to provide access to a NFV-NS. It can be considered as an ingress/egress port for the NFV-NS.

TABLE 52: SAPD INFORMATION ELEMENT

Field	Description
SAPD_id	Unique identifier for a SAPD descriptor.
VLD_id	Identifier of the VL used by the NS to connect with a VNF/PNF of the NS.
assoc_CPD_id	Identifier of the CPD attached to the VNF/PNF of the NS. Note that this CP is the one the VLD gives access to.
(Inherited_fields)	Fields inherited from the CPD entity.

15.5 Virtual Network Functions Forwarding Graph Descriptor (VNFFGD)

The NSD must define how the interconnection is between the present NFs (i.e. the NFs graph), and which are the NFs present in the service. As well it must specify how the traffic flows in the service graph/sequence.

TABLE 53: VNFFGD INFORMATION ELEMENT

Field	Description
VNFFGD_id	Unique identifier for a VNFFGD descriptor.
VNFD_ids	List of VNFD_id present in the NFV-NS. These are nodes of the VNFFG.
AppD_ids	List of AppD_id present in the NFV-NS. These are nodes of the VNFFG

PNFD_ids	List of PNFD_id present in the NFV-NS. These are nodes of the VNFFG.
VLD_ids	List of PNFD_id present in the NFV-NS. These are edges of the VNFFG.
SAPs	List of service access points present in the NFV-NS to access the NS.
CPDs	References to connection points attached to the constituent VNFs and PNFs.

15.6 Virtual Link Descriptor (VLD)

NFV-NSs use virtual links to connect the set of NFs present in the network service. Every VL should satisfy parameters to specify QoS, connectivity, security and deployment flavours. Here we present the fields we consider:

TABLE 54: VLD INFORMATION ELEMENT

Field	Description
VLD_id	Unique identifier for a VL descriptor.
protocol	The communication protocols used in the link: Ethernet, IPv4, IPv6, etc.
VL_flavour	Latency, packet delay, packet loss ratio, traffic priority and availability of the VL.
description	A string specifying the purpose of the VL.
signature	A security signature to provide authenticity of the VLD.

15.7 Monitoring Parameter

The monitoring parameters are a set of metrics to control the behaviour of resources used by the deployed NFV-NS.

TABLE 55: MONITORING PARAMETER INFORMATION ELEMENT

Field	Description
monitoring_param_id	Unique identifier for a monitoring parameter.
name	Explanatory name of the monitoring parameter.
metric	Specify the metric used to monitor a certain resource, e.g., CPU load measured as percentage.

15.8 Life Cycle Management Operation (LCMO)

This information element specifies which script should be executed to deal with every event that can trigger the LCM.

TABLE 56: LCMO INFORMATION ELEMENT

Field	Description
LCMO_id	Unique identifier for a LCM entity.
event	Type of event that triggers the LCM: periodic check, traffic peaks, over-usage of resources, etc.
management_type	Specify the type of management action that is triggered upon the appearance of the related event: scale, healing, resources increase/decrease, update, etc.
assoc_NF	ID of the VNF or PNF associated to the LCM event.
assoc_VL	ID of the VL associated to the LCM event.
script	Program to be executed upon the occurance of the event and to perform the management_type action with the associated VL or NF.

15.9 Network Service Deployment Flavour (NSDF)

This information element details the specific flavour of the NFV-NS to be deployed, and references the flavours of the components that are part of the NFV-NS.

TABLE 57: NSDF INFORMATION ELEMENT

Field	Description
NSDF_id	Unique identifier for a NSDF entity.
vnfDfs	IDs of the VNF deployment flavors for the VNFs present in the NS.
pnfDfs	IDs of the PNF deployment flavors for the PNFs present in the NS.
virtualLinkDfs	IDs of the VL deployment flavors for the VLs present in the NS.
NsDfs	IDs of the NS deployment flavors for the embed NSs present in the NS.
dependencies	List of IDs specifying the deployment order of the constituent VNFs and nested NSs.

15.10 Virtual Network Function Deployment Flavour (VNFDF)

Information element used to specify deployment details of a certain VNF.

TABLE 58: VNFDF INFORMATION ELEMENT

Field	Description
VNFDF_id	Unique identifier for a VNFDF entity.

vnfID	IDs of the VNF related to this flavor.
num_instances	Number of VNF instances to be deployed.
instantiationLevel	Levels (1…N) of resources that can be used to instantiate the VNF using this flavour.
affinity_rule	Set of affinity rules related to the flavour. This is a list of restrictions such as location, resource-specific deployments, etc.
CPDs	List of connection point descriptor IDs that will be attached to the VNF in this deployment flavor. A matrix must be provided to specify the list of CPDs of every instance.
VLDs	Virtual link descriptor IDs used for the ingress/egress traffic that the VNF will process. A matrix must be provided to specify the list of VLDs of every instance.
monitoring_params	List or matrix of monitoring parameter IDs for every VNF instance to be deployed.
LCMOs	List/matrix of Life Cycle Management Operation IDs related to each instance specified in the flavor.

15.11Virtual Link Deployment Flavour (VLDF)

This information element enhances the specification of behavioural requirements used by the VL to be deployed. It is necessary to meet the requirements that the link will have to hold within the NFV-NS that it belongs to.

TABLE 59: VLDF INFORMATION ELEMENT

Field	Description
vldf_id	Unique identifier for a VLDF entity.
latency	Link latency.
jitter	Specifies the packet delay variation.
packetLoss	Determines the maximum packet losses of the VL.

16 Annex VII: Vertical Service Examples

16.1 Collecting Sensor Data

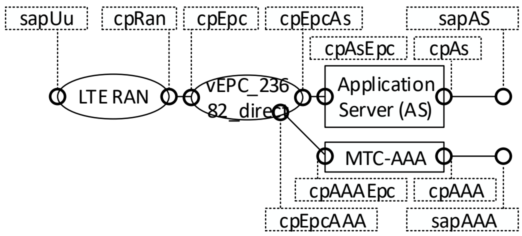
This example defines a vertical service to collect monitoring data via an LTE network. It is based on the 3GPP reference architecture in [45]. This architecture is already provided in the VSB, leaving it to the vertical to provide the amount of sensors and their message rates and the virtual machine images of an application server and of a AAA server. A TSP could also provide a more complete VSB to verticals, where the AAA server and the application server according to a specific IoT platform are already included. In such a case the vertical would get the ability to connect to these servers to register sensors and to retrieve pre-processed monitoring data, resp.

16.1.1 Vertical Service Blueprint

An example blueprint for such a vertical service is presented in Table 60. The information to be provided by a vertical is marked as **<<boldface text>>**.

TABLE 60: VERTICAL SERVICE BLUEPRINT SENSOR MONITORING

Field	Description
Name	LTE Sensor Monitoring
Description	Sensor data can be collected via the LTE air interface and aggregated in an application server. Sensors are authorized via a AAA server. The network scenario corresponds to the reference architecture in 3GPP 23.682, indirect mode. The application server and AAA server are provided by the vertical.
Version	1.0
Identity	Xyz4711_bp
Parameters	<coverageArea, Coordinates, “LTE coverage area needed”, Service Constraints/Geographical area> <sapLocation, Coordinate, “Location of SAP”, Service Constraints/sapAS Location> <deviceAmount, Integer, “amount of sensors”, SLA/sapUu> <msgRate, Integer, “message rate of a sensor”, SLA/sapUu> <msgSize, Integer, “average message size of a sensor>, SLA/sapUu> <aggregatedBw, Integer, “bitrate of aggregated monitoring data”, SLA/cpAs> <asVM, URL, “location of AS VM”, functional component/as/image”>

	<aaaVM, URL “location of MTC-AAA VM”, functional components/aaa/image>
Atomic functional components involved	vEPC_23682_indirect (see Table 61) 4GRAN (see Table 62) MTC-AAA (see Table 63) AS (see Table 64)
Service sequence	A VNFFG in textual notation (ETSI NFV IFA 014 [38]) of 
Connectivity service	sapAS - cpAs, sapAAA - cpAAA, cpAsEpc - cpEpcAs, cpAAAEpc - cpEpcAAA: L3VPN sapUu: 4G
External interconnection	sapAS, sapAAA, sapUu
Internal interconnection	n/a (no other services, except those listed already are needed)
SST	n/a (see the field SLA instead)
Service constraints	Geographical area: **<<Coordinates describing the plant boundary>>** sapAs location: **<<MetroArea of this sap>>** Security: low Priority: medium Cost: n/a Synchronization: low Etc.
Management and control capabilities for the tenant	Provider managed
SLA	sapUu: **<<N>>** devices with **<<rate>>**msg/min of **<<size>>**B. cpAs: **<<bwAs>>**bps cpAAA: 10Mbps latency sapUu - cpAsEpc: 50ms latency sapUu - cpAAAEpc: 50ms

Monitoring	sapUu bandwidth
Lifetime	On-demand, without limitation
Charging	To be defined

The information on the functional components is described in Table 61, Table 62, Table 63, and Table 64. Note that the functional blocks for the AAA server, see Table 63, and the AS, see Table 64, contain placeholders for information to be provided by the vertical.

TABLE 61: ATOMIC FUNCTIONAL COMPONENTS VEPC_23682_INDIRECT

Field	Description
Number of Application servers.	n/a (provided by Translator)
Images of virtual applications.	SGW: http://abc.com/sgw SCS: http://abc.com/scs, etc.
Virtual application connection end points	cpEpc, cpEpcAs, cpEpcAAA
Lifecycle operations	To be defined
Scaling rules	SCS, HSS, etc.: Scale out: 80% load, Scale in: 60% load

TABLE 62: ATOMIC FUNCTIONAL COMPONENT 4GRAN

Field	Description
Number of Application servers.	n/a (provided by Translator)
Images of virtual applications.	n/a, eNbs are PNFs
Virtual application connection end points	sapUu, cpRan
Lifecycle operations	To be defined
Scaling rules	n/a

TABLE 63: ATOMIC FUNCTIONAL COMPONENT MTC-AAA

Field	Description
Number of Application servers.	1
Images of virtual applications.	vmAAA: **<<urlAAA>>**
Virtual application connection end points	cpAAAEpc, cpAAA
Lifecycle operations	To be defined
Scaling rules	Scale out: 80% load, Scale in: 60% load

TABLE 64: ATOMIC FUNCTIONAL COMPONENT AS

Field	Description
Number of Application servers.	n/a (provided by Translator)
Images of virtual applications.	vmAS: **<<urlAS>>**
Virtual application connection end points	**cpAsEpc, cpAs**
Lifecycle operations	To be defined
Scaling rules	Scale out: 80% load, Scale in: 60% load

16.1.2 Vertical Service Descriptor

In this example of a VSD, the information, which was still missing in the blueprint has been provided, written in boldface. In some cases, symbolic values or an explanation has been written instead of actual values for the sake of clarity.

TABLE 65: VERTICAL SERVICE DESCRIPTOR SENSOR MONITORING

Field	Description
Name	LorryMovement_ConstructionSite_Ulm
Description	The position of lorries on a big construction site in Ulm are monitored
Version	1.1
Blueprint	Xyz4711_bp
Identity	Abc0815_vsd
SST	n/a (There are several SAPs with different requirements, a single SST is not sufficient)
Service constraints	Geographical area: **city area of Ulm and surroundings** sapAs location: **Region_Ulm** Security: low Priority: medium Cost: n/a Synchronization: low Etc.
Management and control capabilities for the tenant	Provider managed
SLA	sapUu: **500** devices with **1**msg/min of **200**B. cpAs: **1G**bps

	cpAAA: 10Mbps latency sapUu - cpAsEpc: 50ms latency sapUu - cpAAAEpc: 50ms
Monitoring	sapUu bandwidth
Lifetime	On-demand, without limitation
Charging	To be defined

TABLE 66: ATOMIC FUNCTIONAL COMPONENT MTC-AAA

Field	Description
Number of Application servers.	1
Images of virtual applications.	vmAAA: **http://mole.com/aaa**
Virtual application connection end points	cpAAAEpc, cpAAA
Lifecycle operations	To be defined
Scaling rules	Scale out: 80% load, Scale in: 60% load

TABLE 67: ATOMIC FUNCTIONAL COMPONENT AS

Field	Description
Number of Application servers.	n/a (provided by Translator)
Images of virtual applications.	vmAS: **http//mole.com/as_location**
Virtual application connection end points	**cpAsEpc, cpAs**
Lifecycle operations	To be defined
Scaling rules	Scale out: 80% load, Scale in: 60% load

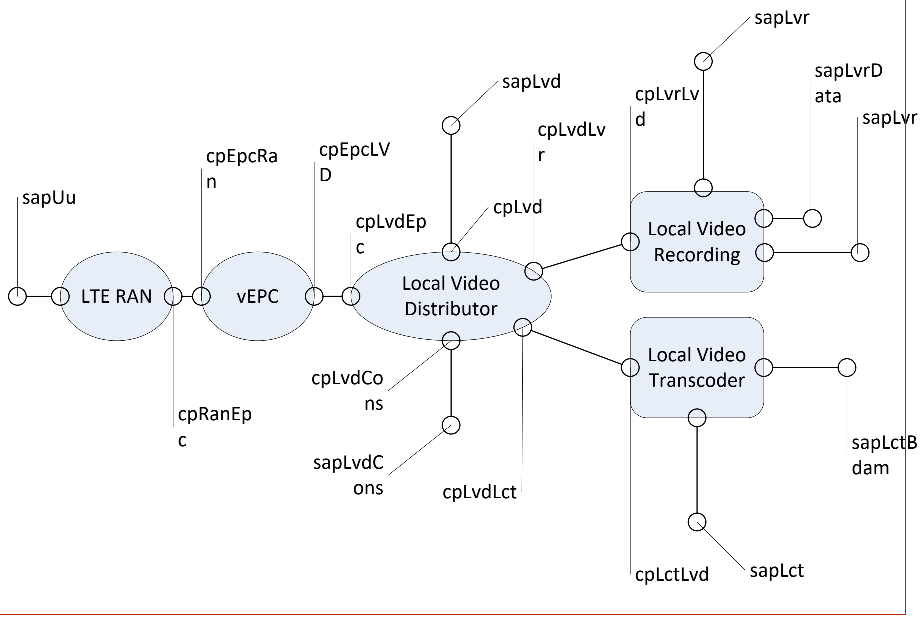
16.2 Entertainment

The following sections provide an illustrative example of the information required in the descriptors of the 5G-TRANSFORMER project, using the example of the On site live event experience (OLE) use case described in [1]. The numbers used in the example are just for illustrative purposes and do not aim to represent the numbers expected in real scenarios. We plan to extend the service definition during the evolution of the project.

16.2.1 Vertical Service Blueprint

TABLE 68: VERTICAL SERVICE BLUEPRINT ENTERTAINMENT

Field	Description

Name	5g_transformer.ole_bp
Description	OLE Blueprint example for fan engagement. The local video distributor determines if the request should be redirected to the live video feeds which are handled by the local video transcoder, or to the local cache of videos which is represented by the local video recording. The service has also access points to connect to the local broadcast dam of the venue, to external content sources and finally an access point to connect to other service instances in the same venue.
Identity	Xyz4712_bp
Parameters	<coverageArea, Coordinates, “LTE coverage area needed”, Service Constraints/Geographical area> <sapLocation, Coordinate, “Location of SAP”, Service Constraints/sapAS Location> <security Level, HIGH|MEDIUM|LOW, “Security Level of the service”, Service Constraints/Security> <Priority, HIGH|MEDIUM|LOW, “Priority of the service”, Service Constraints/Priority> <userAmount, Integer, “amount of users”, SLA/sapUu> <videoRate, Integer, “bandwidth to allocate per stream”, SLA/sapUu>
Atomic functional components involved	Local Video distributor Local Video Recording Local Video Transcoder vEPC
Service sequence	

Connectivity service	sapLvd - cpLvd, sapLvr-cpLvr, sapLct - cpLct, cpLvdEpc - cpEpcLvd, sapLctBdam-cpLctBdam, sapLvdCons-cpLvdCons: L3VPN cpLvd-cpLvr, cpLvd-cpLvt: GRE sapUu: 4G
External interconnection	sapLvr, sapLvd, sapLct, sapLvr, sapLvdData, sapLctBdam
Internal interconnection	sapLvd
SST	n/a
Service constraints	Geographical area: **<<Coordinates describing the venue boundary>>** sapAs location: **<<MetroArea of this sap>>** Security: **<<Security Level>>** Priority: **<<Priority>>**
Management and control capabilities for the tenant	Provider managed
SLA	latency sapUu - cpLvrEpc: 10ms sapUu: **<<N>>** devices with **<<rate>>**.
Monitoring	sapUu bandwidth sapUu-cpLvdEpc latency Local Video Recorder: cpu, memory, storage Local Video Distributor: sessions, cpu, memory Local Video Transcoding: active feeds, cpu, memory
Lifetime	On-demand, without limitation
Charging	To be defined

16.2.2 Vertical Service Descriptor

TABLE 69: VERTICAL SERVICE DESCRIPTOR ENTERTAINMENT

Field	Description
Name	5g_transformer.OLE.venue_a
Description	Vertical service descriptor of the ole service with the input parameters required in venue_a
Version	1.1

Blueprint	Xyz4712_bp
Identity	Abc0816_vsd
SST	n/a (There are several SAPs with different requirements, a single SST is not sufficient)
Service constraints	Geographical area: area **surrounding venue 'A'** sapAs location: **venue A** Security: **High** Priority: **Medium** Etc.
Management and control capabilities for the tenant	Provider managed
SLA	sapUu: **10** devices with **100 mbps**. cpLvd: **1** Gbps cpLvrData: 10Gbps cpLvtBdam: 10Gbps latency sapUu - cpLvdEpc: 5ms latency sapUu - cpLvdLvr: 5ms latency sapUu - cpLvdLvt: 5ms
Monitoring	sapUu bandwidth
Lifetime	On-demand, without limitation
Charging	To be defined

Here, the number of application servers for the local video distributor is fixed already in the blueprint.

TABLE 70: ATOMIC FUNCTIONAL COMPONENT LOCAL VIDEO DISTRIBUTOR

Field	Description
Number of Application servers.	1
Images of virtual applications.	vmLvd: **http://ole.com/urlLvd**
Virtual application connection end points	cpLvdEpc, cpLvd
Lifecycle operations	To be defined
Scaling rules	cpu: determined by translator Ram: determined by translator

TABLE 71: ATOMIC FUNCTIONAL COMPONENT LOCAL VIDEO RECORDER

Field	Description
Number of Application servers.	Determined by translator
Images of virtual applications.	vmLvr: **http://ole.com/urlLvr**
Virtual application connection end points	cpLvrLvd, cpLvrData, cpLvr,
Lifecycle operations	To be defined
Scaling rules	cpu: determined by translator when trigger reached Ram: determined by translator when trigger reached Storage: determined by translator when trigger reached

TABLE 72: ATOMIC FUNCTIONAL COMPONENT LOCAL VIDEO TRANSCODER

Field	Description
Number of Application servers.	Determined by translator
Images of virtual applications.	vmLvt: **http://ole.com/urlLvt**
Virtual application connection end points	cpLvtLvd, cpLvtBdam, cpLvt,
Lifecycle operations	To be defined
Scaling rules	cpu: determined by translator when trigger reached Ram: determined by translator when trigger reached Storage: determined by translator when trigger reached

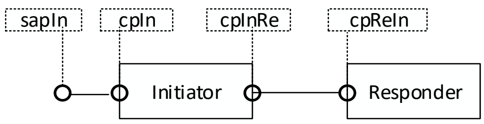
16.3 Latency probe

This example of a vertical service is not intended as a vertical service for production use. But it can be used to test the actual deployment mechanisms and to determine the latency among two data centers. The vertical service provides two servers, able to ping each other and external access to one of the servers to actually trigger the ping.

16.3.1 Vertical Service Blueprint

The information that is still open in the VSB in is the geographical location of some connection points, allowing to specify indirectly in which datacenter the servers will be deployed. In contrast to the example on Collecting Sensor Data in Section 16.1, there is no virtual machine image to be provided by the vertical and the lifetime of vertical service instances are limited to one hour.

TABLE 73: VERTICAL SERVICE BLUEPRINT LATENCY PROBE

Field	Description
Name	Latency Probe
Description	An Initiator and Responder Linux Server are connected to each other. The vertical can login to the Initiator and trigger Pings to the Responder.
Version	2.0
Identity	Xyz4712_bp
Parameters	<loc1, Coordinate, "location of cpInRe", Service Constraints/cpInRe location> <loc2, Coordinate, "location of cpReIn", Service Constraints/cpReIn location>
Atomic functional components involved	stdLinux (see Table 74)
Service sequence	VNFFG in textual notation (ETSI NFV IFA 014) of 
Connectivity service	sapIn - cpIn, cpInRe - cpReIn: L3VPN
External interconnection	sapIn
Internal interconnection	n/a
SST	n/a (see the field SLA instead)
Service constraints	Geographical area: n/a cpInRe location: **<<MetroArea of this sap>>** cpReIn location: **<<MetroArea of this sap>>** Security: low Priority: medium Cost: n/a Synchronization: low Etc.
Management and control capabilities for the tenant	Provider managed
SLA	n/a

Monitoring	n/a
Lifetime	On-demand, 1h
Charging	To be defined

TABLE 74: ATOMIC FUNCTIONAL COMPONENTS STDLINUX

Field	Description
Number of Application servers.	n/a (provided by Translator)
Images of virtual applications.	Initiator, Responder: http://abc.com/stdLinux
Virtual application connection end points	cpIn, cpInRe, cpReIn
Lifecycle operations	To be defined
Scaling rules	n/a

16.3.2 Vertical Service Descriptor

This example VSD describes a vertical service to probe latency among two test bed sites of the 5G-TRANSFORMER project.

TABLE 75: VERTICAL SERVICE DESCRIPTOR FOR LATENCY PROBE

Field	Description
Name	Latency CTTC SSSA
Description	Measure the latency between CTTC (Barcelona) and SSSA (Pisa)
Version	2.0
Blueprint	Xyz4712_bp
Identity	Abc0816_vsd
SST	n/a
Service constraints	Geographical area: n/a cpInRe location: **Region_Barcelona** cpReIn location: **Region_Pisa** Security: low Priority: medium Cost: n/a Synchronization: low Etc.

Management and control capabilities for the tenant	Provider managed
SLA	n/a
Monitoring	n/a
Lifetime	On-demand, 1h
Charging	To be defined

16.4 Cloud/Edge robotics

This VSB defines a cloud and edge robotics service for industrial automation e.g. warehouse automation. The robots interact with their environment using sensors and actuators. The data generated by the sensors is transmitted to a cloud data center using a packet data network e.g. LTE or 5G. The cloud data center subsequently processes the raw data to determine the action of the actuators and conveys the results to the robot via a return or feedback path. In addition to the sensor data service described in Section 16.1, this service also considers robotic control via remote actuation. The data conveyed between the robot and the data center can be grouped into three categories.

- Robot management data used to provide actuator instructions for the robot, e.g. motion, rotation, etc.
- Low bandwidth data generated by sensors on board the robot e.g. temperature, humidity, luminosity, etc.
- High bandwidth data generated by high resolution cameras on board the robot.

Depending on the application each type of data can processed locally, at an edge data center or a distant cloud data center.

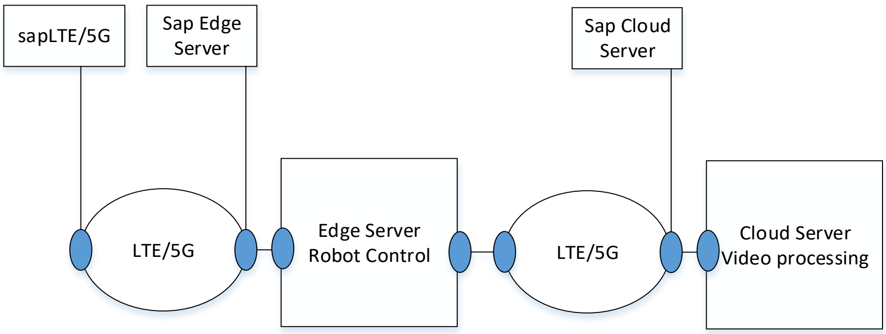
In this scenario, the vertical would require connectivity between the robots and specific edge and cloud servers.

16.4.1 Vertical Service Blueprint

An example VSB for such a vertical service is presented below.

TABLE 76: VERTICAL SERVICE BLUEPRINT CLOUD/EDGE ROBOTIC

Field	Description
Name	Warehouse automation
Description	Cloud and edge robotics for warehouse automation via LTE or 5G. The cloud and edge servers are provided by the vertical.
Version	1.0
Identity	CZuat#725_ttyu
Parameters	• Number of robots. • Number of Edge server(s), i.e. corresponding VNF instances. • Number Cloud server (s), i.e. corresponding VNF instances. • Location of warehouse (s).

	• Location of Edge server (s). • Location of Cloud server (s). • Average data rate per robot. • Image of robot control. • Image of video processing.
Atomic functional components involved	Robot control. Video processing.
Service sequence	
Connectivity service	sapLTE/5G - sap Edge Server, sapEdgeServer - sapCloudServer
External interconnection	n/a
Internal interconnection	n/a
SST	n/a
Service constraints	Warehouse geographical location Security: medium
Management and control capabilities for the tenant	Provider managed
SLA	sapLTE/5G <<N>> devices with <<rate>> bps. Capacity sapLTE/5G - sap Edge Server: 1000 Mbps. Capacity sapEdgeServer - sapCloudServer: 1000 Mbps. latency sapLTE/5G - sapEdgeServer: 10ms latency sapEdgeServer - sapCloudServer: 1000ms
Monitoring	sapLTE/5G bandwidth

	latency sapLTE/5G - sapEdgeServer latency sapEdgeServer - sapCloudServer
Lifetime	1 year
Charging	n/a

TABLE 77: ATOMIC FUNCTIONAL COMPONENT ROBOT CONTROL OR VIDEO PROCESSING

Field	Description
Number of Application servers.	n/a (provided by Translator)
Images of virtual applications.	Robot control: http://robotcontrolimage.com Video processing: http://videoprocimage.com
Virtual application connection end points	sapLTE/5G, sapEdgeServer, sapCloudServer
Lifecycle operations	To be defined
Scaling rules	n/a