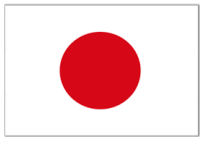

D2.5: Final report on the overall system architecture definition

Orange, Aalto, Ericsson, Eurecom, FOKUS, Hitachi, KDDI, Mandat International, NESIC, University of Tokyo, Waseda University

Document Number	D2.5
Status	Final
Work Package	WP2/T2.3
Deliverable Type	Report
Date of Delivery	15.8 2019
Responsible	Orange
Contributors	Orange, Aalto, Ericsson, Eurecom, FOKUS, Hitachi, KDDI, Mandat International, NESIC, University of Tokyo, Waseda University
Dissemination level	PU

This document has been produced by the 5GPagoda project, funded by the Horizon 2020 Programme of the European Community. The content presented in this document represents the views of the authors, and the European Commission has no liability in respect of the content.

This project has received funding from *the European Union's Horizon 2020 research and innovation programme* under grant agreement No 723172.

AUTHORS

Full Name	Affiliation
Sławomir Kukliński	Orange
Lechosław Tomaszewski	Orange
Taleb Tarik	Aalto University
Miloud Bagaa	Aalto University
Ibrahim Afolabi	Aalto University
Ilias Benkacem	Aalto University
Nicklas Beijar	Ericsson
Adlen Ksentini	Eurecom Institute
Pantelis Frangoudis	Eurecom Institute
Marius Corici	Fraunhofer FOKUS
Björn Riemer	Fraunhofer FOKUS
Eleonora Cau	Fraunhofer FOKUS
Fabian Eichhorn	Fraunhofer FOKUS
Thomas Magedanz	Fraunhofer FOKUS
Daisuke Okabe	Hitachi
Kota Kawahara	Hitachi
Hidenori Inouchi	Hitachi
Itsuro Morita	KDDI Research
Yoshinori Kitatsuji	KDDI Research
Zaw Htike	KDDI Research
Phyo May Thet	KDDI Research
Cédric Crettaz	Mandat International
Kazuto Satou	NESIC
Masato Yamazaki	NESIC
Hiroshi Takezawa	NESIC
Akihiro Nakao	The University of Tokyo
Shu Yamamoto	The University of Tokyo
Du Ping	The University of Tokyo
Yoshiaki Kiriha	The University of Tokyo
Toshitaka Tsuda	Waseda University
Takuro Sato	Waseda University

Executive summary

This deliverable provides the description of the final 5G!Pagoda architecture. The initial concept that has been outlined in D2.3. It has been updated, taking into account the experience gained during the project duration. The final architecture concept includes the outcome of work performed in other work packages as well as experience gained during implementation. The concept is also impacted by the progress of research and standardization related to network slicing.

This deliverable consists of:

- an updated (in comparison to D2.3) analysis of network slicing approaches available in the literature;
- updated network slicing related standardization status;
- updated requirements and open issues related to network slicing;
- high-level 5G!Pagoda architecture description;
- discussion about the implementation of the internal blocks of the architecture, including management and orchestration ones.

Table of Contents

List of Tables

List of Figures

Abbreviations

Throughout this document, the following acronyms, listed in Table 1, are used.

Table 1 – List of Acronyms

Abbreviation	Original term
3GPP	The Third Generation Partnership Project
5G PPP	The Fifth Generation Infrastructure Public-Private Partnership
5GMF	The Fifth Generation Mobile Communications Promotion Forum
AI	Artificial Intelligence
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
BSS	Business Support System
BSSO	Business Service Slice Orchestrator
CDN	Content Distribution Network
CN	Core Network
CPU	Central Processing Unit
DM	Domain Manager
DSSO	Domain-Specific Slice Orchestrator
E2E	End to End
EBI	East-Bound Interface
ECA	Event-Condition-Action
EM	Element Manager
eMBB	enhanced Mobile Broad Band
eMBMS	evolved Multimedia Broadcast Multicast Service
FCAPS	Fault, Configuration, Accounting, Performance, Security (management)
FG IMT-2020	The Focus Group on network aspects of IMT-2020
GPU	Graphics Processing Unit
HNS	Hardware Nodes and Subsystems
HSS	Home Subscriber System
IaaS	Infrastructure as a Service
ICN	Information-Centric Networking
IMT	International Mobile Telecommunications

Abbreviation	Original term
IoT	Internet of Things
ITU-T	International Telecommunication Union Telecommunication Standardization Sector
M2M	Machine to Machine
MANO	MANagement & Orchestration
MdO	Multi-domain Orchestrator
MDSO	Multi-Domain Slice Orchestrator
MEC	Mobile Edge Computing
MEO	Mobile Edge Orchestrator
mMTC	massive Machine-Type Communications
MQTT	Message Queue Telemetry Transport
N2N	Neighbour to Neighbour
NAS	Non-Access Stratum
NBI	North-Bound Interface
NFV	Network Function Virtualization
NFVI	Network Function Virtualization Infrastructures
NFVO	Network Function Virtualization Orchestrator
NGMN	Next Generation Mobile Network Alliance
NM	Network Manager
NMS	Network Management System
NSaaS	Network Slice as a Service
NSD	Network Service Descriptor
O&M	Operation and Maintenance
OFDM	Orthogonal Frequency-Division Multiplexing
OSS	Operation Support System
PaaS	Platform as a Service
PBM	Policy-Based Management
PNF	Physical Network Function
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network

Abbreviation	Original term
RRC	Radio Resources Control
SBC	Slice Border Control
SBI	South-Bound Interface
SDN	Software-Defined Networking
SDO	Standards Development Organization
SDR	Software-Defined Radio
SON	Self-Organizing Network
SOS	Slice Operation Support
SSF	Slice Selection Function
TMF	TM Forum (formerly: TeleManagement Forum)
TOSCA	Topology and Orchestration Specification for Cloud Applications
uRLLC	Ultra-Reliable Low Latency Communications
VIM	Virtual Infrastructure Manager
VMN	Virtual Mobile Network
VNF	Virtualized Network Function
VNFaaS	VNF as a Service
VNFM	Virtualized Network Functions Manager
WAN	Wide Area Network
WBI	West-Bound Interface
WIM	WAN Infrastructure Manager
WP 5D	Working Party 5D – IMT systems
xMBB	extreme Mobile BroadBand

1. Introduction

1.1. Objectives

The objectives of this deliverable are to describe the final 5G!Pagoda architecture and to specify the overall system from the functional and architectural standpoint, providing a description of functional entities that are specific for generic network slicing, beyond the current 3GPP scope. Part of the architecture is related to the orchestration of slices that is responsible for lifecycle management of slices and their optimization while another part is related to the effective active service implementation within the slices. The high level, reference architecture serve as the basis for the implementation of 5G!Pagoda concepts, acting as a synchronization point between the management and orchestration developments and the customized network functions ones. The final architecture describes functional entities that are proposed for the dynamic network slicing, including the design and specification coming from WP3 and WP4 as well as the implementation results from WP5.

1.2. Motivation and scope

In recent years, there have been noticeable research initiatives on the Fifth Generation of Mobile Communications System (5G System), in Europe, Japan and worldwide. However, the focus has been merely on high-level ideas and the generic directions that assume the use of software-based solutions as much as possible, while not considering the specific needs of either the orchestration mechanisms enabling a multi-slice environment neither on the actual software network functions from which the service will be composed of. A comprehensive and detailed 5G architecture is yet to be defined; leaving still space for research and standardizations activities aiming to shape the 5G system architecture, one of the core objectives of this 5G!Pagoda project.

The 5G!Pagoda architectural approach is based on the comprehensive state of the art analysis of network slicing approaches that include 5G PPP projects, 3GPP 4G and 5G slicing oriented activities, IMT-2020 Focus Group concepts and the NGMN vision of slicing. The main goal of 5G!Pagoda architecture is to provide a high-level network slicing architecture that can be instantiated to multiple networking solutions and is compliant as much as possible with all analyzed network slicing concepts, including 3GPP but is much more generic than the 3GPP approach. The concept allows also for integration of legacy systems as well as advanced, programmable data plane operations in order to provide a highly efficient data plane. Moreover, the issues of network slicing in a multi-domain environment have also been addressed. The reference architecture described in this deliverable is flexible and allows instantiation of a multi-domain network slice in many different ways.

Following this introductory chapter, the remaining part of the document is structured as follows:

- Chapter 2 provides the basic terminology used throughout this report. It includes the descriptions of abbreviations and technical terms;
- Chapter 3 describes state of the art for network slicing architecture. It provides a brief overview of existing software networks architecture concepts, on-going standardization and research projects related to slicing;
- Chapter 4 describes the requirements that were the basis for the architecture design
- Chapter 5 describes 5G!Pagoda architecture. The description includes the details of the proposed slice structure and different slice types;
- Chapter 6 consists of orchestration options and details of the orchestration architecture of 5G!Pagoda as well as a detailed description of orchestration-related functional components;
- Chapter 7 consists of a description of selected implementation related issues;
- Chapter 8 consists of the concluding remarks.

2. Terminology

Table 2 lists terms used in this document along with their definitions.

Table 2 – Definition of terms used in this document

Terminology	Definition
The Fifth Generation of Mobile Communications System (5G system)	The latest mobile telecommunications standards beyond the current 4G/IMT-Advanced standards. Rather than faster peak Internet connection speeds, 5G system planning aims at a higher capacity than the 4G system, allowing a higher number of mobile broadband users per area unit and allowing consumption of higher or unlimited data quantities in gigabyte per month and user.
Cloud computing	A type of Internet-based computing that provides shared computer processing resources and data to computers and other devices on demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources (e.g. computer networks, servers, storage, applications and services).
Software Defined Networking	A network architecture concept that allows network administrators to manage network services through abstraction of lower-level functionality. SDN is meant to address the fact that the static architecture of traditional networks does not support the dynamic, scalable computing and storage needs of more modern computing environments such as data centres.
Slice	An isolated collection of programmable resources to implement network functions and application services through software programs to accommodate individual network functions application services within each slice without interfering with the other functions and services on the other slices.
Network Function Virtualization	A network architecture concept that uses the technologies of IT virtualization to virtualize entire classes of network node functions into building blocks that may connect, or chain together, to create communication services.
Virtualized Network Function	Software implementations of network function that can be deployed on a Network Function Virtualization Infrastructure.
Working party 5D – IMT systems	A standard community responsible for the overall radio system aspects of International Mobile Telecommunications (IMT) systems, comprising the IMT-2000, IMT-Advanced and IMT for 2020 and beyond.
Network softwarization	An overall transformation trend for designing, implementing, deploying, managing and maintaining network equipment and network components by software programming, exploiting characteristics of software such as flexibility and rapidity of design, development and deployment throughout the lifecycle of network equipment and components, enabling the re-design of network and services architectures; allow optimization of costs and processes; and enable self-management.
5G!Pagoda	A research project is federating Japanese and European 5G system test-beds to explore relevant standards and align views on 5G system mobile network infrastructure supporting dynamic creation and management of network slices for different mobile services.
Slices of virtual mobile networks	A logical instantiation of a mobile network possible to create with both legacy platforms and network functions, but substantially lower barriers to using the technology, for example through increased flexibility and decreased costs.
Mobile slice	A slice of virtual mobile networks.
Resource Orchestrator	The entity responsible for domain-wide global orchestration of network services and software resource reservations in terms of network functions over the physical or virtual

Terminology	Definition
	resources the RO owns. The domain an RO oversees may consist of slices of other domain [1].
Service Orchestrator	The Network Service Orchestration is managing the lifecycle of network services. The Resource Orchestration provides an overall view of the resources present in the administrative domain to which it provides access and hides the interfaces of the VIMs present below it [1].
The Third Generation Partnership Project (3GPP)	Collaboration between groups of telecommunications associations, known as the Organizational Partners. The initial scope of 3GPP was to make a globally applicable 3rd generation (3G) mobile phone system specification based on evolved Global System for Mobile Communications (GSM) specifications within the scope of the International Mobile Telecommunications-2000 project of the International Telecommunication Union (ITU). The scope was later enlarged to include the development and maintenance of GSM and related '2G' and '2.5G' standards including GPRS and EDGE, UMTS and related '3G' standards including HSPA, '4G' and 5G standards.
Next Generation Mobile Networking Alliance (NGMN)	A mobile telecommunications association of mobile operators, vendors, manufacturers and research institutes. It was founded by major mobile operators in 2006 as an open forum to evaluate candidate technologies to develop a common view of solutions for the next evolution of wireless networks. Its objective is to ensure the successful commercial launch of future mobile broadband networks through a roadmap for technology and friendly user trials. Its office is in Frankfurt, Germany.
Internet of Things (IoT)	The internetworking of physical devices, vehicles (also referred to as 'connected devices' and 'smart devices'), buildings and other items – embedded with electronics, software, sensors, actuators and network connectivity that enable these objects to collect and exchange data. In 2013 the Global Standards Initiative on the Internet of Things (IoT-GSI) defined the IoT as 'the infrastructure of the information society'.
The Fifth Generation Infrastructure Public-Private Partnership	A group initiated by the European Commission and industry manufacturers, telecommunications operators, service providers, SMEs and researchers. It aims to deliver solutions, architectures, technologies and standards for the ubiquitous next-generation communication infrastructures of the coming decade.
The Fifth Generation Mobile Communications Promotion Forum (5GMF)	A group actively promoting 5G system study in line with trends both in Japan and abroad based on a roadmap on 5G system implementation policy published by the government of Japan.
Mobile Edge Computing	A network architecture concept that enables cloud computing capabilities and an IT service environment at the edge of the cellular network. The basic idea behind MEC is that by running applications and performing related processing tasks closer to the cellular customer, network congestion is reduced and applications perform better.
Content delivery network	A globally distributed network of proxy servers deployed in multiple data centres. The goal of a CDN is to serve content to end-users with high availability and high performance. CDNs serve a large fraction of the Internet content today, including web objects (text, graphics and scripts), downloadable objects (media files, software and documents), applications (e-commerce, portals), live streaming media, on-demand streaming media and social networks.
Quality of Experience	A measure of a customer's experiences with a service (web browsing, phone call, TV broadcast, call to a Call Centre). QoE focuses on the entire service experience and is a more holistic evaluation than the more narrowly focused user experience (focused on a software interface) and customer-support experience (support focused).

3. Related works

Although the definition of network slicing is still under heavy discussion, we have generally defined 'slice' as an isolated collection of programmable resources to implement network functions and application services mostly in software to accommodate individual network functions application services within each slice without interfering with the other functions and services on the other slices. It has to be noted that the network slicing is still in its infancy and many of the existing solutions address some specific issues of network slicing only and there still no universal framework that provides carrier-grade network slicing. It is expected that in the near future, many of the present approaches will converge and the final architecture will allow for E2E network slicing in a heterogeneous and multi-domain environment that includes also the slicing of RAN.

The 5G!Pagoda architecture is strongly focused on the network slicing, having two perspectives: the end-user service perspective concentrating on the components within the slices and the issues related to slice configuration in order to provide the appropriate service, and the infrastructure management perspective concentrating on the proper allocation of resources to slices and the interoperability between different slices in a multi-slice environment.

In the following sections of the chapter, a synthetic overview of activities related to network slicing is presented. This overview is a concise update to an overview of network slicing stat-of-the-art presented in deliverable D2.3 [2].

3.1. NGMN

The NGMN has published in January 2016 fundamental and often cited document entitled 'Description of Network Slicing Concept' [3]. It includes a detailed description of terminology and network slicing related concepts that are presented in Figure 1. The concept forms the basis for ETSI NFV and 3GPP network slicing activities. The NGNM network slicing architecture is comprised of three layers:

- The **Resource Layer** is composed of physical and logical resources. Physical components such as compute nodes, storage, transport and radio access network equipment belongs to a pool of physical resources. Logical resources are regarded as a virtualized pool of resources dedicated to a particular network function or shared between multiple network functions. The term Network Function refers to a '*processing function in a network*'.
- The **Network Slice Instance Layer** includes Network Slice Instances. The Network Slice Instance is '*a set of network functions and resources to run these network functions, forming a complete instantiated logical network to meet certain network characteristics required by the Service Instance(s)*'. The particular Network Slice Instance may be shared between multiple Service Instances.
- The **Service Instance Layer** represents end-user or business services, which are deployed over a network slicing platform. The particular service is described as a Service Instance.

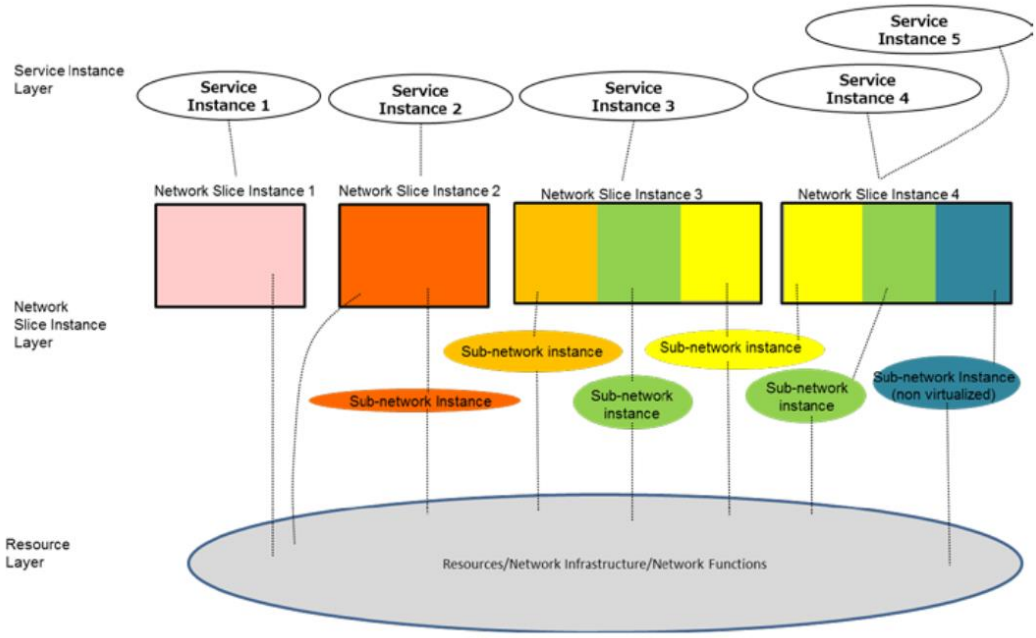

Figure 1 – NGMN network slicing concept [3]

An important functional entity is the Sub-network Instance. It acts as a subset of network functions and resources dedicated to these network functions. The Sub-network Instance may be a hardware-based subsystem. One or more Sub-network Instances may be stitched together in order to compose a Network Slice Instance. Both the Network Slice Instance and Sub-network Slice Instance are described by Network/Sub-network Slice Blueprint that characterizes '*the structure, configuration and the plans/work flows for how to instantiate and control the Network Slice Instance during its life cycle*'. The Network/Sub-network Slice Blueprint provides a description of characteristics and performance requirements of Network/Sub-network Instance that is tailored to a specific service (e.g. URLLC service) represented as Service Instance.

The NGMN vision illustrates neatly basic architectural requirements and shed lights on overall 5G architecture. It provides a high-level view that clarifies well the most important assumptions, but there is not any functional decomposition provided. Thus, the design and functionality of network entities, interfaces between them and the orchestration framework should be described in more detail. Also, multi-domain orchestration has not been taken into considerations by NGMN, and this gap has been addressed by the 5G!Pagoda architecture.

3.2. 5G PPP architecture and network slicing

The 5G PPP architecture draft [4] defines network slicing as one of the most important parts of the overall 5G architecture. **The 5G PPP organization defines network slicing as *'multiple logical networks as independent business operations on a common physical infrastructure', composed of* both physical and virtual network functions, as well as edge-cloud and central-cloud deployments.**

The 5G PPP introduces overall network softwarization and programmability framework, which is presented in Figure 2. It is split up into planes that are not completely independent from each

other. Thus, interactions between them are realized by reference points/interfaces. Figure 2 covers the entire ecosystem of 5G softwarized networks. *The forwarding/data plane* is the collection of resources across all network devices capable of forwarding the control and data plane traffic. The common resources comprise the network infrastructure, including Fixed and Radio Access Networks, Aggregation and Core Networks and Network Clouds. The Infrastructure Control Plane is a set of functions that are responsible for controlling network devices, elements and data processing units. This control plane is separated from the control and enforcement functions, which are network segment-specific and integrated with data plane devices. Thus, the control plane is not fully centralized, and data plane contains control agents, called Common Control & Enforcement entities. The centralized control entity – Infrastructure Control of (Virtual) Network Functions – is responsible for managing network softwarization functions, orchestration functions, mobility control functions, cloud functions, mobile edge computing functions as well as includes adaptors to different enforcement functions.

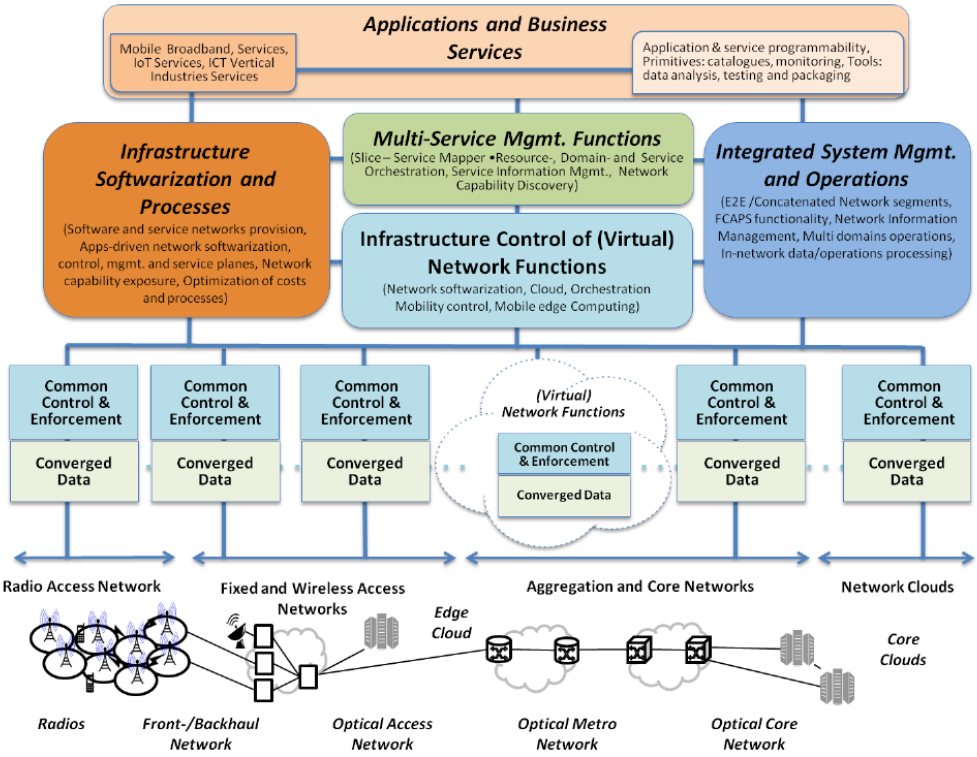

Figure 2 – Overall network softwarization and programmability framework [4]

The Infrastructure Softwarization Plane enables the provisioning and operation of software and service networks. It includes software for designing, implementing, deploying, managing and maintaining network components and/or services by the programmable interface. The key responsibility of this plane is the deployment of new network and management services, which may result in E2E slice provisioning. This plane acts as an Application Programming Interface to a software network platform and enables programmability of E2E network services. The Integrated Network Management and Operations Plane enables the creation, operation and control of dedicated management functions operating on the top of 5G E2E infrastructure. It provides management capabilities (FCAPS, Monitoring, Network Information Management) for each network slice. Furthermore, the functions of this plane coordinate multi-domain operations. The

functions of the Multi-Service Management Plane are able to create, install, configure and manage a group of network functions and/or nodes. This plane includes Slice-Service Mapper functions, Resource, Domain and Service Orchestration functions Service Information Management functions, and Network Capability Discovery functions. It can be perceived as Operations Support System (OSS) with extensions that allow dynamically creating, operating and controlling multiple logically isolated network services running on the top of network (virtual) infrastructure. The Application and Business Plane defines service deployments from the business point of view and provides a business interface for parties that want to launch a network service on the softwarized network platform.

The 5G PPP draft defines also the Service & Infrastructure Management and Orchestration architecture, which in fact is a reference architecture of the E2E orchestration framework.

A single-domain multi-service control and management platform consists of three planes: the Multi-Service Management Plane, the Integrated Management and Operation Plane and the Application and Business Service Plane. The purpose of the orchestration framework architecture is the description of service and resource orchestration mechanisms across the 5G network. To provide flexibility and programmability 5G PPP has designed key functionalities, which are: the Service Development Kit to let developers quickly implement novel network services, the Management System to take care of reliable operation and the Service Platform including customizable Service (Domain) Orchestrator, Resource Orchestrator, a Service Information Base and various enablers.

The 5G PPP describes the multi-domain orchestration architecture, but the 'multi-domain' term is defined loosely, and it refers to as a *'multi-technology (orchestrating resources and/or services using multiple domain orchestrators) or multi-operator (orchestrating resources and/or services using domain orchestrators belonging to multiple administrative domains)'*. In general, it means that the multi-domain orchestrator provides resources and services using multiple technology-specific domain orchestrators that can belong to various operators and/or vendors.

3.3. 5G PPP projects related to slicing

There is a wide scope of projects related to 5G architecture, participating in the 5G PPP initiative and funded within the EU program 'Horizon 2020'. These 5G PPP projects span many topics related to the 5G system, including several projects that aim at designing the new 5G network architecture in terms of network slicing. In this subsection, we will describe only those which are closely linked or overlapped with 5G!Pagoda.

3.3.1. 5G-Crosshaul

The 5G-Crosshaul project [5], [6], [7] is dedicated to a 5G backhaul and fronthaul transport development. Based on SDN and NFV as key principles it aims to provide unified (both at data and control planes' levels) multi-technology (mmWave, µWave, optical, copper) E2E packet-based transport domain and support recursive and homogenous multi-tenancy of underlying heterogeneous resources giving tenants freedom and flexibility of, controlling, reconfiguring and

operating their networks without influencing other coexisting tenants' networks. Dealing with various transport technologies forming specific technological domains, the project faces the question of multi-domain slices orchestration.

It applies the principles of decoupling of data and control planes, logically centralized control and exposure of abstract resources and their states to external applications. In the control plane, the ETSI NFV architecture with MANO (NFVO/VNFM/VIM with plugins for controlling SDN, storage and computing resources via 5G-Crosshaul or Non-5G-Crosshaul/legacy interface) is applied. The architecture vision assumes a 'domain-based approach', and the transport domain (5G-Crosshaul domain) control plane interacts with 5G core and 5G access. The NBI of the control plane provides programming and monitoring the underlying data plane by a common set of core services and primitives, in fact, constitutes the E2E business orchestrator. The vision of 5G-Crosshaul architecture assumes the existence of autonomous legacy control functions, e.g. like MPLS/GMPLS.

In the data plane, the transport network functions will be accomplished with 5G-Crosshaul forwarding/processing elements (as VNFs) or non-5G-Crosshaul NEs (legacy infrastructure, e.g. radio links) controlled by XCI via its SBI. From the functional point of view, the data plane (see: Figure 3) will be composed of three layers:

- the lowest one, corresponding to overlay of all infrastructure layers, is composed of:
  - *interconnection plane*, formed by networking infrastructure providing the higher planes with connectivity with heterogeneous links; composed of packet forwarding elements creating a unified packed-based network;
  - *general processing plane*, serving for VNFs (e.g. BBUs, CDNs) located in processing units;
  - *the end-point plane*, formed by 5G network architecture elements or functions, utilizing the 5G-Crosshaul transport (e.g. RRHs, eNodeBs, core network functions in operators' POPs
- the middle layer visualizing the common packet network based on technology abstraction, unified framing and common data and the control and management planes; the unified abstract view covers different technologies underneath (fronthaul and backhaul) and enables exposure of the integrated transport network;
- the upper layer referring to functional features requested from the services of transport network infrastructure: (1) *reconfigurability* – dynamic allocation of capacity or rerouting of traffic, reallocation of resources (both networking and processing as well, e.g. dynamic NFV relocation) between busy and idle areas in order to satisfy demands changing in time or provide inter-domain optimization (e.g. RAN and transport) or preserve SLA obligations; (2) *energy efficiency* – drowsing, de-activating or decommissioning dynamically the underused network parts in order to reduce the energy consumption by transport network elements, may be combined with the joint optimization of RAN and transport network resources; (3) *multi-tenancy/network slicing support* – enabling recursive multi-tenancy (especially recursive XCI architecture) for homogeneous way of sharing of heterogeneous underlying infrastructure, the concept of introduction of 'partitioner/slicer' component at each forwarding element is considered; these features will be exposed via NBI API to higher level applications.

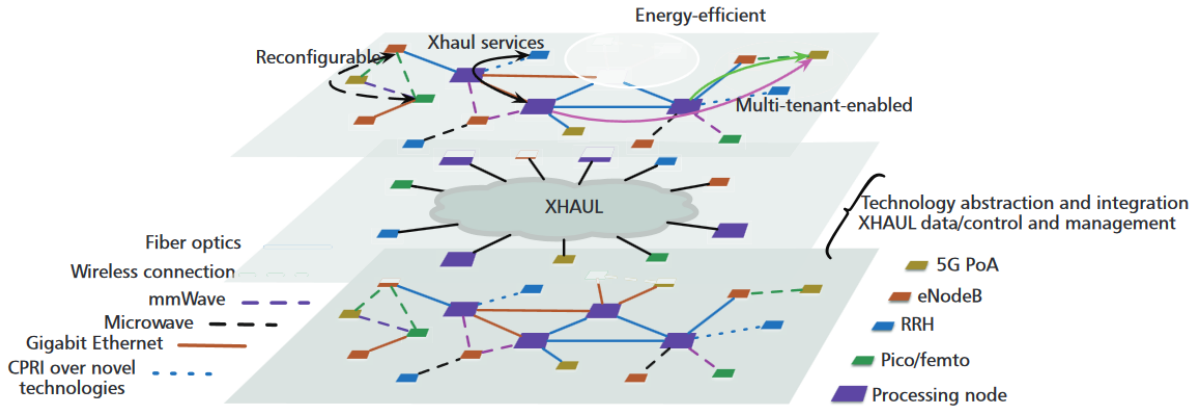

Figure 3 – 5G PPP 5G-Crosshaul functional structure [5]

3.3.2. 5G Exchange

The 5G Exchange (5GEx) project aims to enable E2E and cross-domain orchestration of services (multiple technology domains or administration/ownership domains as well) in multi-vendor environments. Such orchestration tool should provide provisioning of services in an automated way and able to interact with resources exposed by various operators to one common market of services of interconnected heterogeneous infrastructure delivered by infrastructural providers and commercially used by service providers. This automation is expected to allow E2E feasibility check, provisioning and validation in a timeframe of tens of minutes instead of the current timeframe of tens of days.

The project has defined a series of use cases grouped in three categories: connectivity, VNFaaS and NSaaS. The mid one contains the generic case of network slicing, i.e. delivering separated logical virtual partitions allowing homogeneous control independently of the ownership of the underlying heterogeneous networks (NFV/SDN-based, but also legacy ones). The last category defines three cases of various degree of complexity: 'IaaS' (VMs + associated storage + connectivity), 'GiLAN/roaming' (network slicing + remotely controlled VNF placement) and 'My Cloud Anywhere' (user location changes entailing dynamic rearrangement or even spatial migration of provisioned network following user's trace).

The proposed 5G-Ex high-level framework is shown in Figure 4. The entire reference framework containing all involved components and interworking interfaces consists of resource domains and their respective (Single-/Multi-) Domain Orchestrators performing resource and/or service orchestration; Multi-provider Multi-domain Orchestrator (MdO) coordinating resource and/or service orchestration at multi-domain level (multi-technology orchestration of resources and/or services using multiple Domain Orchestrators or multi-operator orchestration of resources and/or services using Domain Orchestrators belonging to multiple administrative domains). There are three 5G-Ex APIs defined: ① for interactions between business customers and MdOs (Business-to-Customer, B2C) in order to specify the business customers' requirements for services; ② for interactions between different MdOs (business-to-business, B2B) in order to request and orchestrate resources and services across different administrative domains; ③ for interactions of MdOs with Domain Orchestrators to orchestrate resources and services within the same administrative domains. The MdO may also belong to 3rd party service providers not having their

own resource domains but operating multi-domain orchestrators (brokers/resellers of resources and services owned by other providers).

Separation of contexts of APIs ② and ③ (multi-domain context and local context), even if the technical definition of both interfaces may be exactly the same, provides manageability of resources/services exposed to the 'wholesale market' (selectiveness of what can be drawn out to other MdO operators) and confidentiality of local infrastructure details still ensuring flexibility for every actor of the picture. In case the definition of APIs ② and ③ is exactly the same, the architecture model has inherent recursiveness (refer to the case of 3rd party MdO). Indeed, as the services exposed at the levels ② and ③ are CFSes (according to TMF methodology) and the 'know-how' (mapping to RFSes and their decomposition to Rs) is accomplished at the level of domain orchestrator, these CFSes (defined as commonly understood abstract 'black boxes') can be recursively bundled or unbundled subsequently within the MdOs hierarchy.

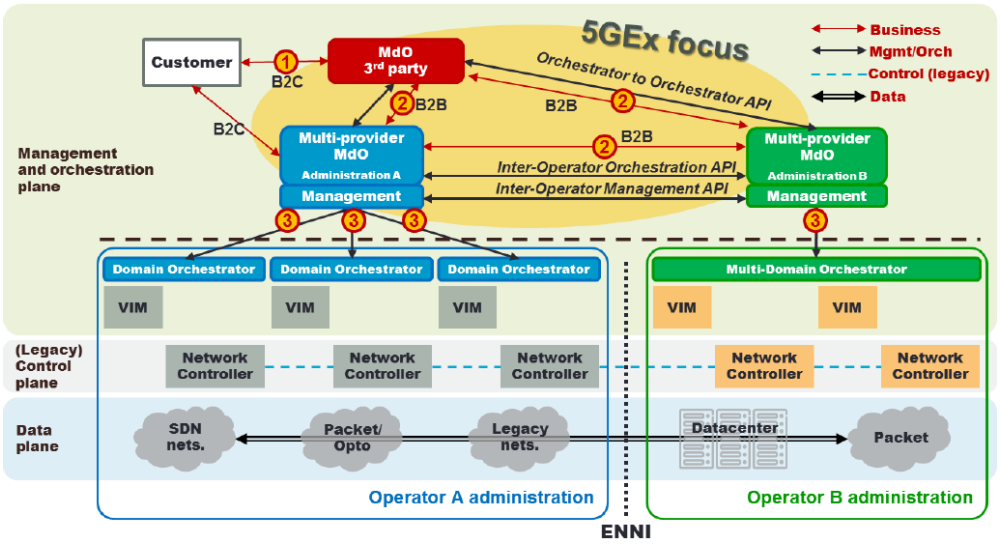

Figure 4 – 5G PPP 5GEx reference architectural framework [8]

The foundation of 5G-Ex orchestration is ETSI NFV-MANO. However, it is extended in order to enable multi-domain and multi-operator orchestration. According to 5G-Ex approach, the network service orchestration part covers OSS-BSS, VNF and EM area. Thus, service orchestration means the configuration of parameters within VNFs and provisioning their proper interactions. The resource orchestration focuses on NFVO and VIM part. Building a multi-domain architecture means setting new inter-operator interfaces and defining interactions between Multi-provider MdO and distant: Inter-provider NFVO (implementing multi-provider service decomposition), VNF Manager, Element Managers (responsible for FCAPS of VNFs), SLA Manager (responsible for reporting on the performance of its own part of the service), Service Catalogue, Topology Distribution (exchanging topology information with peer MdOs, to be included in the Resource Topology Repository) and MD-PCE (Multi-domain Path Computation Element as defined by IETF). The functional model of the 5G-Ex concept of multi-domain orchestration, explaining both E2E and N2N relations is shown in Figure 5. The basic relation is N2N and publically available information about the basic inter-provider topology, and optionally service catalogue information may be distributed hop-by-hop to all or predefined group of providers. This information supports initial mapping of the multi-provider NFVO orchestration process and depending on provider policies may also include more

detailed provider internal topology, IT resource capability/location and access information of orchestration interfaces. The extended, bilateral communication in E2E customer-provider relationship model is intended to provide extended bilateral business information exchange, which is invisible for 3rd parties.

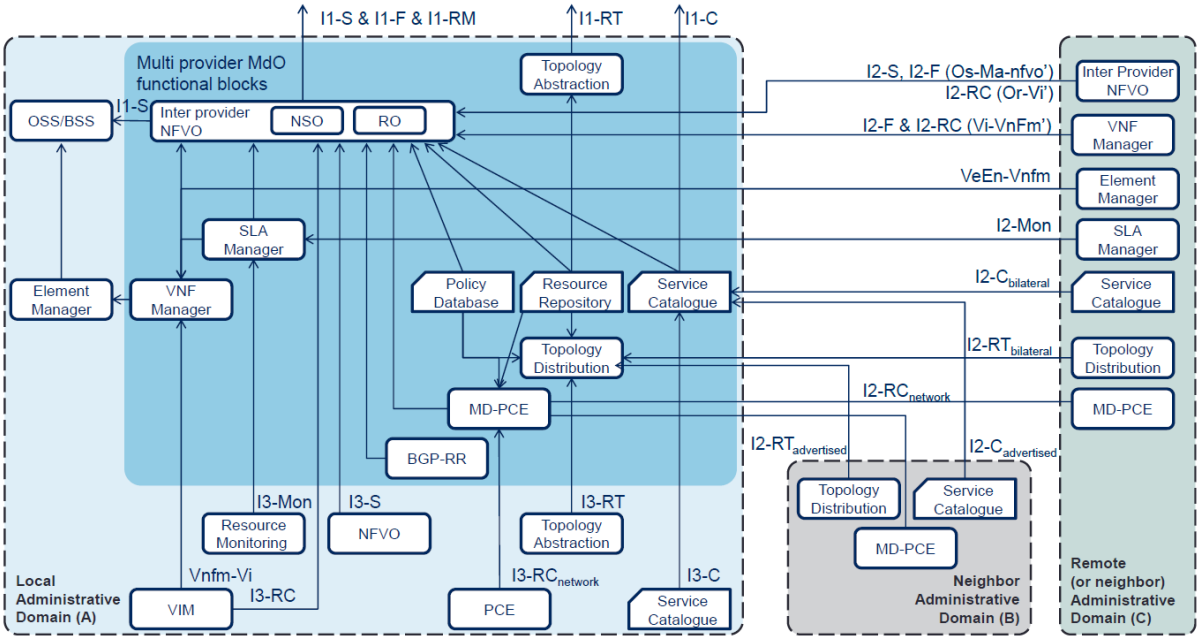

Figure 5 – 5G PPP 5GEx functional model of multi-domain orchestration [8]

3.3.3. 5G NORMA

The project 5G NORMA is aimed at the development of a customizable 5G mobile network architecture with adaptability to handling portfolios of services and patterns of traffic changing in time and with adaptability to future trends (including growing expectations about performance, security and cost or energy efficiency), especially focusing on 3GPP roadmap adoption. The 5G NORMA architecture principles are:

- Network customization by the adaptive allocation of network functions and service specificity-oriented network architecture;
- Service-aware resource sharing with network slicing for network multi-tenancy;
- Network programmability for flexible network control;
- The separation between dedicated and common NFs both in CP and UP;
- Full integration of PNFs and VNFs within the overall architecture;
- Split of Data/Control/MANO/Service layers;
- Hierarchical orchestration – ETSI NFV MANO-based, extended to accommodate network slicing needs and inter-slice E2E service orchestration (5G-NORMA-MANO);
- Joint optimization of mobile access and core network functions.

These principles of 5G NORMA mobile network architecture entail specific concept of the underlying environment of shared infrastructure, multi-tenancy management and slice/E2E service orchestration (see Figure 6). The actual mobile network is composed of control and data layers (CP and DP) composed of one set of common functions (referred as a 'common slice') and multiple sets of dedicated functions (called 'dedicated slices'). Common slices may include both PNFs and VNFs. Except for 'canonical' CP-NFs both in common and dedicated slices, there are defined Software-Defined Mobile network (SDM) Controller functions (called SDM-C for the dedicated functions and SDM-X for the common functions). These controllers deal with MANO layer via their NBIs and with their own NFs (CP and UP) via their SBIs. Their role will be explained later.

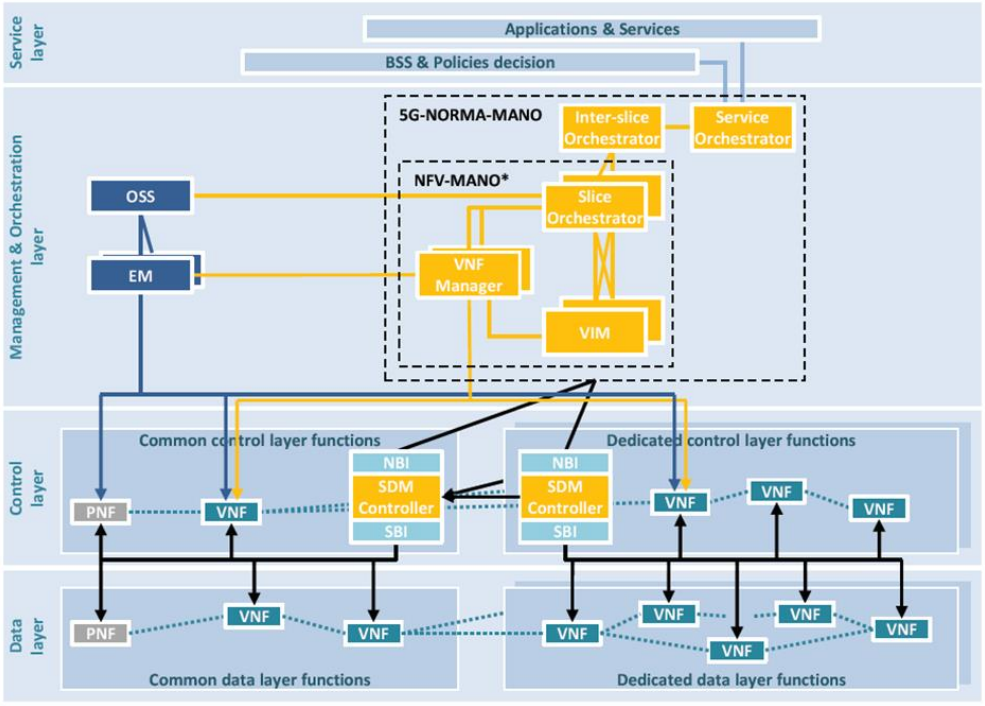

Figure 6 – 5G NORMA functional reference architecture [9]

The MANO layer is composed of (1) EMs (performing FCAPS) and their 'overlay' OSS (legacy functions, but VNF-aware) managing the actual 5G network and (2) 5G-NORMA-MANO domain for E2E network orchestration and lifecycle management. The idea of 5G-NORMA-MANO incorporates a constellation of ETSI NFVI-MANO objects instances: NVF-Os called 'Slice Orchestrators' (each dedicated for a single CP+UP slice, either common one or dedicated ones), VNF-Ms (one per vendor) and VIMs (one per cloud). These instances are owned and operated by the respective owners of slice service/infrastructure. Additionally, the E2E slice instances are instantiated by the Service Orchestrator (owned and operated either by slice tenant or slice service provider). Its tasks are service function chaining (decomposing a service request to a set of network services and their related SLAs/KQIs) and choosing the way of implementation (re-parameterization of existing slices, their reconfiguration by amending service chains or creating of new slices) based on business and policy decisions.

The gateway for E2E slice Service Orchestrators is the Inter-Slice Orchestrator (or Orchestrator Manager) operated by the slice service provider. It is not only a dispatcher. Having a comprehensive view of underlying resources (aware of a 'principle of communicating vessels' within the underlying

infrastructure) and service requests' requirements is able to handle the process of dynamic provisioning of the slices and resource sharing management, coordinating globally the resource allocation to slices/tenants in order to avoid requirements conflicts between slices (especially slices belonging to different tenants). Additional Inter-Slice Orchestrator can be considered on top of the one belonging to slice service provider for the tenant who wants to have autonomy of optimization of resources among slices and/or has several slice service providers.

The SDM Controller is a key function in 5G NORMA concept. Every single slice (both common one and dedicated ones) has its own SDM Controller instance, translating decisions of the higher-level control applications into commands to PNFs (in case of common slice only) or VNFs within the specific slice and controlling these NFs' performances. The NBI exposed to MANO layer serves for 'VNF insert/VNF reconfigure' functions and management of resources assigned to the network slice, especially in order to satisfy QoE/QoS targets (if they cannot be met, the SDM-C may request re-orchestration). The SDM-X (controller-coordinator of the common slice) is additionally responsible for the coordination of access of all the controllers of dedicated network slices that use common NFs the E2E chain and resolving potential conflicting requests.

It should be noted, that despite the fact, that 5G NORMA is nominally dedicated to entire mobile network architecture (AN+CN), the applications of SDM-C/SDM-X listed in [10] focus on RAN part of 5G network and affect the following NFs: SON, RAN Paging, eMBMS Control, NAS Control, RRC Slice, eICIC (in case of SDM-C) and Multi-tenancy Scheduling (RRC and ICIC schemes), mMTC RAN Congestion Control, QoS Control of radio stack (in case of SDM-X).

3.3.4. SliceNet

The goal of the SliceNet project is to develop the QoE-oriented E2E multi-technology (UE, RAN, edge, core, backbone) slicing framework for verticals, which will support slice provisioning, control, management and orchestration. The framework exposes an integrated API for verticals and applies cognitive network management and orchestration paradigm. The architectural concept (see Figure 7) consists of Data, Control and Management planes, which are interconnected by the SliceNet Service Bus [11]. The centralized Management Plane deals with service, slice and resource layers. The management tasks are grouped into sub-planes for monitoring, orchestration, cognition and management information. The Control Plane of the framework is responsible for slicing control execution according to Management Plane decisions. The Service Bus-based architecture supports various use cases and management scenarios (slice life-cycle operations, including cognition and autonomous control closed-loops).

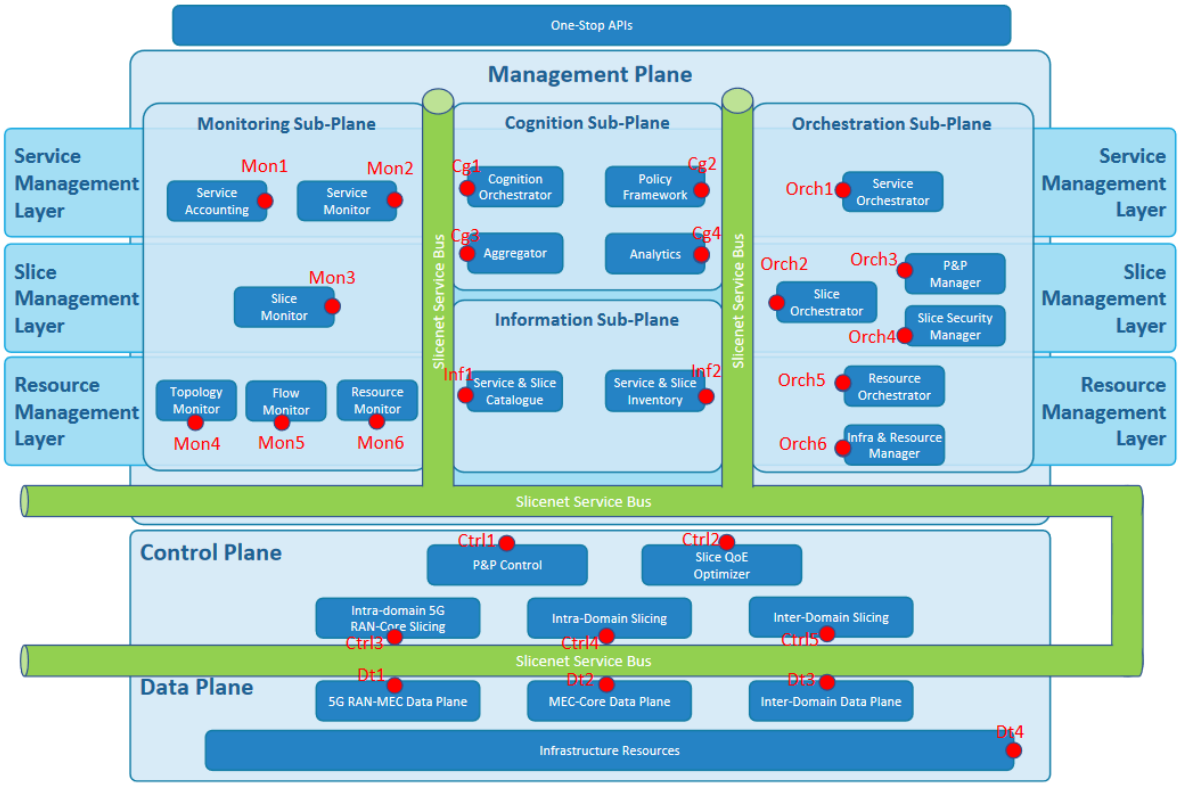

Figure 7 – SliceNet logical architecture and reference points [11]

3.3.5. 5GTANGO

The 5GTango project develops network slicing features as part of the '5GTANGO Service Platform' (SP)'. The SP is able to create Network Slice Templates, as well as instantiate multiple NSIs and perform other lifecycle management actions. The SP extends the ETSI NFV MANO framework by addition of several components: Gatekeeper (SP API and GUI), SLA Manager, Policy Manager, Slice Manager, Monitoring and Infrastructure Adapter [12]. The Slice Manager is the component responsible for the management of network slices. This component has two main sub-components: Slice2NS Mapper and Slice Lifecycle Manager. The Slice2NS Mapper is responsible for the mapping between the NSIs and the underlying Network Services, while the Slice Lifecycle Manager is responsible for the NSI LCM. The Slice Manager will reuse the SP's existing NFVO APIs to manage the underlying Network Services. The NSTs are be stored in the Catalogue, and the NSIR (NSI Repository) stores the records of the running NSIs and historical data in the Repository.

3.3.6. MATILDA

MATILDA aims at the development of intelligent and unified orchestration mechanisms for the automated placement of the 5G-ready applications and for the lifecycle support of the required network slices. High-level Services Provider objectives are translated into deployment and runtime policies using a set of optimization and adaptation mechanisms which support runtime adaptation of the application components and/or network functions.

MATILDA reshapes the ETSI NFV MANO architecture in order to adapt it to the needs of multi-site cloud/edge computing environment for vertical industries [13]. The typical configuration of OSS/BSS + NFVO + MEO is enriched by an additional element – 'Computing Slice Broker' (CSB), which main role is to expose a VIM-like interface to the Vertical Application Orchestrator (VAO, operated by the vertical), to enable uploading images of components and direct management of their life-cycles. Thus, the vertical has its own 'virtualized' computing infrastructure (for hosting application components) exposed in an abstract way, without details of infrastructure PoP(s) actually providing them. Hence, the CSP plays the role of a 'proxy IaaS' interface. The high-level service objectives and policies are enforced by the VAO through the OSS/BSS path, resulting in delivery of virtual computing resources implemented on multiple POPs.

3.4. ITU-T activities on network slicing

There are network slicing activities at ITU-T, Study Group 13 (especially within Questions 20 and 21). The term 'network slice' (a complete logical network to provide certain network characteristics) has been initially defined in the context of the architecture of mobile core networks [14], following the 3GPP concepts of dedicated core networks (see further). Later on, the approach has been generalized to the entire communication network architecture. The ITU-T has already published recommendations concerning network softwarization and slicing framework [15], slice life-cycle management and orchestration [16], high-level architecture of the slice management plane [17], mechanism for multiple slice attachment of UE [18], network slicing business models [19], architecture of IMT-2020 network [20] and its capabilities exposure requirements [21]. There are still on-going works on network capability exposure function [22], network slice orchestration and management [23] and network slicing with AI-assisted analysis [24].

3.5. IETF slicing

The IETF organization has published several BOF-level drafts related to network slicing (problem introduction and statement [25]-[26], architecture [27], management and orchestration [28], implementation of slicing in segment routing network [29] and autonomic slice networking [30]). More importantly, they have formed a working group named 'NetSlices', which is responsible for network slicing research. However, the group has not provided any new output yet.

3.6. 3GPP network slicing

The first 'pre-slicing' concepts developed and defined by 3GPP, known as DÉCOR [31] and eDÉCOR [32], were introduced in 3GPP Releases 13 and 14, respectively. It is assumed that an operator deploys several dedicated Evolved Packet Cores (EPC) connected to the same Radio Access Network and each EPC is tailored to support a specific type of mobile service. Networks deploying (e)DÉCOR may have a common core network (CCN), which may be assigned to UEs when a dedicated core network (DCN) is not available. Initially, the mobile terminal could be attached to the CCN and meanwhile redirected to be served by DCN, based on the specific Subscription Parameter stored in the Home Subscriber System (HSS) and in case of eDÉCOR also based on DCN

'suggestion' from the UE, which corresponds to the requested service (UE-assisted selection). So, the network shall enable the redirection of affected attaching/attached UEs to appropriate DCN. The key disadvantage of (e)DÉCOR is that UEs can be attached to one EPC at the same time, i.e. it cannot work as a 'multi-service' terminal in the sense of the ability to be attached to different service-tailored EPCs. The (e)DÉCOR approach is an enhancement of existing LTE networks and cannot be treated as a fully-fledged network slicing solution.

The fundamental document for network slicing within the 3GPP release 14 is the TR 23.799 'Study on Architecture for Next Generation System' (NextGen) [33]. The document lists the key features relevant to the design of the new architecture for next-generation networks. Support of network slicing (as a concept that allows multiple logical networks to be built on the top of common shared physical infrastructure) is treated as one of the key features and is noticed as an enabler to provide networks on as-a-service basis and meet the wide range of use cases. Besides network slicing, the list of key features contains QoS framework, mobility management, session management, session and service continuity, efficient user plane paths, network functions granularity and interaction, network capabilities exposure, policy and charging framework, security, network discovery and selection, interworking and migration and NextGen core support for IMS. The NextGen is a very visionary concept which assumes:

- network slice as a complete logical network including RAN and CN, which may be built of logical and physical resources, according to some specific network slice template;
- network slice instances may be fully or partly, logically and/or physically isolated from each other;
- new functions: Network Slice Selection Function (NSSF), Network Exposure Function (NEF);
- separation of CP and UP at the level of single network functions;
- existence of common (e.g. AMF, NSSF or UDM in the CP, RAN in both planes) and slice-specific network functions, the scheme of functions sharing between different slices may be function-specific;
- implementation of the intra-CN-CP communication according to the Service-Oriented Architecture (SOA);
- involvement of UE in the selection of network slices; the preferences to be provided to network in RRC and NAS; additionally, UE capabilities and subscription data may also be used in the process of slice selection;
- selection of slice instance is by Core Network, not by RAN.
- handovers between slices (e.g. in case of leaving the area of coverage by dedicated slice a handover to 'generic' one);
- the ability of simultaneous use of different parallel PDU sessions that belong to different network slices.

The 5G Core (5GC) Stage 2 Architecture defined in TS 23.501 [34] follows the majority of the NextGen visions, with some simplifications (UP functions chain rather slice-specific, while the majority of CN functions rather common except SMF). Following the separation of planes,

important is the transition from 'network of entities' reference model into a 'network of functions' system. Thereupon, 5GC defined by 3GPP should provide stateless, modular and interactive with each other network functions to enable flexible and efficient network slicing. Additionally, such requirements as minimization of dependencies between Access Network and Core Network or support of concurrent access to local and centralized services have been stated. The UE PDU session may belong to only one Network Slice Instance (NSI) per PLMN, and it is agreed that dedicated Session Management Function (SMF) will handle the PDU session of Network Slice Instance.

The 3GPP 5G architecture defines NSSF [34], [35] and supports the attachment of UEs to specific network slices (simultaneously up to 8 slices of the same PLMN [36]; however, the PLMN limitation can be overstepped with UE dual/multi-SIM capability). NSI is identified by Single Network Slice Selection Assistance Information (S-NSSAI), which is composed of Slice/Service Type (SST) that describes a type of service supported by the network slice and Slice Differentiator (SD) – the optional information that may be used to differentiate network slice instances of the same service type. Some of the NSIs may be marked as default and associated with default S-NSSAI. UEs may contain pre-configured S-NSSAI stored in local storage. During initial registration, UE may request access to specific Network Slice Instance by proposing 'Requested S-NSSAI'. Upon successful registration, UE may obtain from AMF a list of 'Allowed S-NSSAI' for serving PLMN.

From the management point of view, 3GPP considers network slices as a specific case of complete logical networks which may be built of network slice subnets (sub-slices, being a specific case of subnetworks). The life-cycles of each sub-slice and of the complete slice built thereof are separate [37]. The 3-layer management hierarchy is, hence (i) communication network/network slice, (ii) subnetwork/network slice subnet and (iii) network function (see Figure 8). 3GPP recognizes two fundamental types of slices: internal network operator's slice and 'Network Slice as a Service', which is offered as a communication service [38]. However, the 3GPP vision of network slices management architectural framework is very high level and distinguishes the layers of management of network functions, network slice subnets, network slices and finally – communications service [39]. The 3GPP *Network Function Management Function* and *Network Slice Subnet Management Function*} are respectively mapped to EM and OSS/BSS into ETSI NFV architectural framework, and they should use the Performance, Fault and Life-Cycle Management interfaces defined by ETSI at respective reference points. At each layer of management, the configuration (i.e. provisioning), fault and performance management services should be provided. Aside from each layer, the management functions, the *Management Data Analytics Function* and Exposure *Governance Management Function* have been defined.

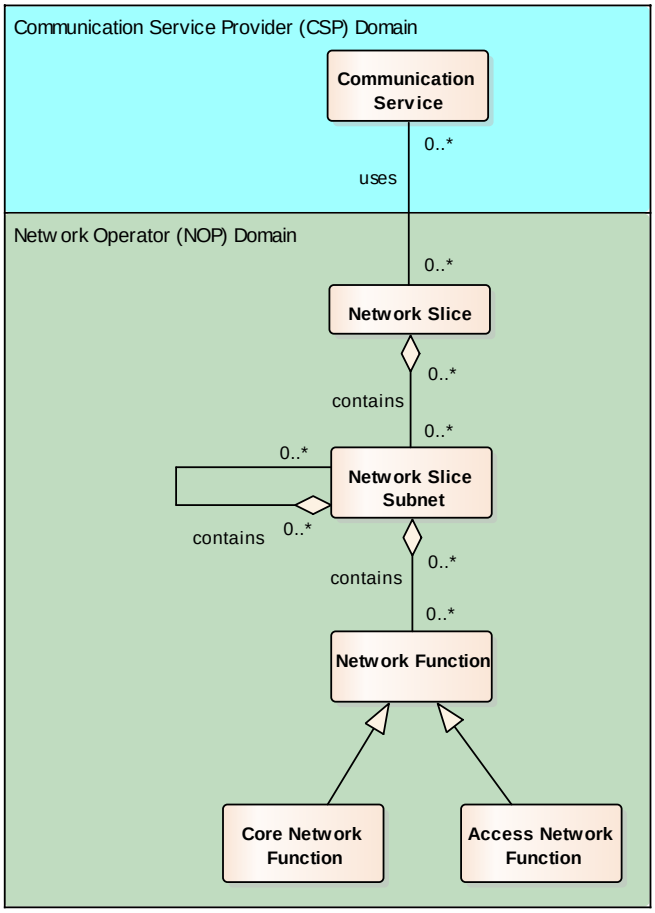

Figure 8 – Network Slice-related information model [37]

The 3GPP TS 28.500 [40] provides management concepts and requirements from the operator's point of view. These are related to mobile networks that include VNFs as a part of EPC or IMS. The management architecture for virtualized mobile networks has been proposed in the context of the ETSI MANO framework. The architecture shown in Figure 9 extends ETSI MANO to support also existing physical network functions (PNFs). System design assumes that legacy 3GPP Management System entities – NM, EM and DM – may take participation in the process of management of both Physical and Virtual Network Functions. NM acts as OSS/BSS and interacts with NFVO. It may initiate life-cycle management of ETSI-defined VNFs and Network Services. EM/DM is responsible for application-level FCAPS management of VNFs and domain- and element-level management of PNFs. EM/DM may also interact with the VNF Manager during life-cycle management of network functions.

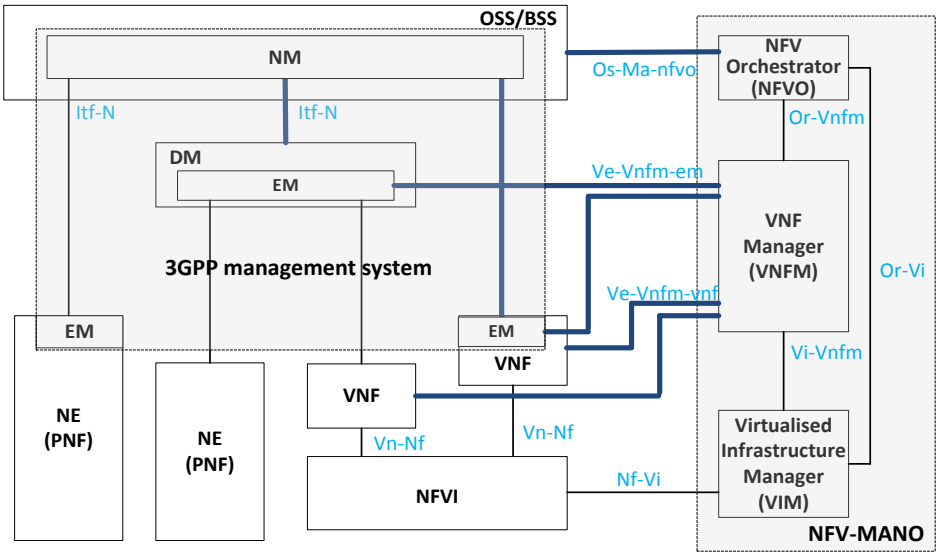

Figure 9 – The mobile network management architecture mapping relationship between 3GPP and NFV-MANO architectural framework [40]

3.7. ETSI NFV activities on slicing

In February 2017, ETSI NFV Industry Specification Group (ISG) published the white paper on 'Network Operator Perspectives on NFV for 5G' [41]. This paper treats network slicing as one of the key features of 5G networks. First of all, authors notice that standard bodies and open-source communities use the term of network slicing in contextually different ways. This conclusion has been also noticed by 5G!Pagoda consortium as one of the risks in early deployments and standardization efforts of the network slicing concept. However, ETSI NFV ISG has defined properly network operator's viewpoint on network slicing as '*service-oriented network construct providing network-on-demand to concurrent applications*'. The important aspect for network operators is support for different services in an efficient way, with required and guaranteed Quality of Service.

The ETSI NFV concept [42] enables software implementation of network functions as software-only entities, abstracted from the underlying hardware. The specific group of interconnected functions forms a *Network Service*, which is intended for implementation of a communication service and can be used for telco-oriented network slicing. Since Release 3, the ETSI NFV MANO framework has been adapted to network-slicing support [43] and can be used for the life-cycle management of network slices. The *Network Slice Management* entity is located in OSS/BSS, i.e. outside ETSI NFV MANO. However, it will be supported by MANO (e.g. delivering necessary monitoring information for possible decision to request corrective resource management actions, which will guarantee the performance described in the network slice descriptor). The framework addresses multi-tenancy [44], multi-domain issues [45], various architectural scenarios (multiple MANO domains, the existence of multiple VIMs/VNFMs within the same NFVO domain) [1]. The inter-NFVO reference point has been defined in [46].

3.8. Summary of the state of the art

The work on 5G network architecture and all of them take into account the network slicing concept as a key enabler to deploy services with different functional requirements. At the time of writing of this deliverable, there were many ongoing research activities related to network slicing, and the concept has been demonstrated so far only in small scale trials. Although the analyzed approaches have defined several interesting mechanisms of network slicing, there is currently a missing cohesion on how the different architectural developments can be integrated into a common architecture, this deterring the fast adoption and further technology development in the area. For example, the 3GPP concepts are rather evolutionary ones, on the other hand, the 5G PPP developments focus more on the 'softwarized' infrastructure, missing yet many features related to the management and the enforcement of network slices, particularly in case of multiple domains.

It seems that in the future, the approaches presented in this chapter will have to converge to a more matured, carrier-grade solution that provides reliability, scalability and especially will be able to address large scale deployment requirements.

All of the analyzed approaches have some different features but are also similar to each other, at least at the high level. The common features of the analyzed architectures are the following:

- There is an Infrastructure/Resources Plane that provides a pool of resources (computing, storage, networking, radio) which can be virtualized and part of resources can be allocated (with isolation) to network slices, this is the basis of cloudification of the network functionality.
- The Network Function Virtualization Plane provides a set of Virtual Network Functions implemented as software and running atop of virtual resources.
- The E2E Orchestration and Management Plane provides functionalities for dynamic deployment of new E2E network services on the top of virtual resources and using a predefined set of interconnected network functions (aka slice template). This plane also takes care of life-cycle orchestration of network slices.
- The Application/Business/Service Plane providing a high-level description of E2E network service as well as a business interface to provision new services.

The state of the art analysis has proved that the 3GPP has proposed the most detailed solutions and procedures of network slicing, but the proposed by 3GPP network slicing approach do not exploit all the possible network slicing features.

We have also discovered several topics or problems that are not addressed yet well or are addressed in a limited scope:

- **Decomposition of orchestrators**. In order to provide reliability, scalability and high availability (stability), the orchestrator entity should be decomposed into functional entities, for example into Multi-Domain (Service) Orchestrator, Domain (Service) Orchestrator and Resource Orchestrator. The Resource Orchestrator should be also decomposed into SDN-Orchestrator and NFV-Orchestrator (MANO) and other technology-specific orchestrators. Such approach let the implementer's use/choose any domain-specific orchestration tool they want to while maintaining a common way of multi-domain communication. The domain-specific

orchestrators should be still able to cooperate with each other using well-defined interfaces in order to provide the multi-domain E2E network slice.

- **Multi-domain slicing and orchestration issues.** The multi-domain architecture is the subject of research of 3GPP and several 5G-PPP projects, but we have found several issues, that should be investigated more deeply. The E2E slices should be created across multiple technological or administrative domains, but not all of them will be composed of domains with full functionality, because of slice-specific requirements or limited ownership capabilities. Moreover, in the context of multi-domain slicing, an E2E network slice description is still a subject of research. The E2E slice template should be aware of the capabilities of each administrative domain.
- **Slice selection and discovery**. This functionality is one of the most important components of the sliced network architecture. Generally, the slices should be advertised to UE, and the UE should be able to attach to multiple slices. Another possibility is UE agnostic slicing in which a proxy entity provides traffic redirection to dedicated slices. The 3GPP has already described slice selection and discovery mechanisms.
- **RAN slicing**. The full support of Radio Access Network slicing is difficult – the data plane or application plane of RAN slicing is relatively simple, but some control plane operations of RAN have to be common for many slices (for example the handover).
- **Data Plane Programmability**. A programmable data plane may be an important enabler for slicing. Various slices may use different data plane protocol stacks, so an effective implementation should be developed, in which different data plane protocols can be docked to the same data plane entity and could be dynamically composed to be used in parallel by one or more slices. Moreover, the NFV-based functions should guarantee high performance of forwarding plane operations by properly selecting the architecture location of the data plane functions and by providing the appropriate shared control.
- **Control Plane customization.** When deploying multiple network slices, there is a need to be able to use the same software network components, while deploying highly customized services. This flexibility should be reflected in the functional architecture of the active service components implemented as software in order to be able to use interoperable and highly configurable products.
- **Legacy systems compatibility**. For telecom operators, it will be a huge advantage if they may use existing hardware-based systems in the slicing context. This is especially true about the RAN, which typically takes 70% of the overall mobile network cost. This problem has only been addressed by some projects, for example, 5G-NORMA, but without details on how the legacy systems will be adopted and later on migrated to fully sliced, software-based networking solutions.
- **Slice management**. One of the challenges for the network slicing architecture is how to provide effective and scalable management in the multi-operator, multi-tenant and multi-domain environment. In fact, the management operations, due to scalability and effective optimization of resources, should be automated as far as possible. On the other hand, the slice owner/operator should also have a management interface in order to monitor slice

performance (SLA) and to configure it according to his needs. Such an interface should be simple and comfortable.

3GPP has introduced an orchestration and management architecture [38], consisting of: (i) a service management function that analyzes incoming slice requests, converting service requirements into networking ones and (ii) a network slice management function, which performs the mapping onto network resources and takes care of the LCM. Although the resource mapping process is carried out across different technology domains, including the RAN, transport and core, the current 3GPP efforts concentrate only on NSIs deployed and managed by a single administrative entity.

A hierarchical multi-domain orchestration architecture is introduced in [47], based on the concept of recursive abstraction and resource aggregation that 'stitches' NSI heterogeneous resources initially on per domain level and then across federated domains. Although different technology domains may belong to a distinct administration, the solution assumes a unified orchestration and management provided by a single administrative domain. Such a unified orchestration and management act as an aggregator without supporting service federation to form an end-to-end multi-domain NSI.

With regard to multi-domain support, ETSI NFV to date has published two informative reports. The first report [48] deals with managing the connectivity of an NS deployed over multiple NFVI sites, referred to as NFVI-PoPs. A single MANO system then manages interconnectivity issues over WAN links linking these NFVI-PoPs. The second report [45] highlights the different architectural options and recommendations to support MANO operations in multiple administrative domains. To manage multi-site/multi-domain NSIs, a direct reference point between the NFV Orchestrator (NFVO) functional elements is recommended in each NFVI-PoP. Such a peer-to-peer approach does not only bring more complexity but is not optimal in view of the delay sensitive nature of MANO operations. In view of the challenges and gaps discussed above, we present a novel multi-domain architecture for the LCM of NSIs deployed across heterogeneous federated infrastructure domains.

Other approaches considered include standard efforts based on 3GPP, ONF-SDN, ITU-T and ETSI NFV-MANO and representative research projects focusing on 5G-EX and 5G-NORMA. To make the analysis concise, the following set of features are selected:

- Multi-domain support: Multiple administrative domains and technology types including RAN.
- Multi-domain service and resource management: Service broker, service management and federated LCM.
- Multi-domain tenant control: 3rd party NS control/orchestration, programmability and recursive virtualization.
- Multi-domain resource 'stitching': unified multi-domain connectivity and cloud mediation.

Table 3 summarizes the details of the quantitative analysis, showing the functional and operational features whereby our proposed multi-domain architecture advances state of the art.

Table 3 – Comparison of selected network slicing orchestration architectures

Orchestration Architectures	Multi Admin. Domains	Multi-Tech. Domains	RAN Orch.	Broker: AC/Neg.	Service Chain & SDN	Service Mang.	Federated LCM	3rd Party Cntl./Orch.	Program-ability	Unif. Connect. Mgmt.	Unif. Cloud Med.
3GPP 28.530 [38]	no	yes	yes	no	no	yes	yes	no	no	no	no
SDN TR-526 [49]	yes	no	no	no	yes	yes	yes	yes	yes	no	no
ITU-T Y.3011 [50]	yes	yes	no	yes	no	yes	yes	yes	yes	yes	no
5G-EX [51]	yes	yes	no	no	no	yes	yes	no	yes	no	yes
5G-NORMA [52]	no	yes	yes	yes	yes	no	no	yes	yes	yes	no
NFV-MANO MD [45]	yes	yes	no	no	no	no	yes	no	yes	no	yes

4. 5G!Pagoda architectural requirements

5G!Pagoda architecture is driven by technical and business requirements. Some of the requirements were collected via state of the art analysis, the analysis of 5G standardization efforts and some of them are the result of work described in deliverable D2.1 [53] that concerns 5G!Pagoda use cases.

4.1. Generic technical requirements

In 5G!Pagoda the following high-level requirements regarding network slicing solutions have been formulated:

- **Flexibility:** A slice is a dynamic entity, which lifecycle has to be managed, from both the perspective of the service (adapting to the momentary service needs) and from the perspective of the resources (dynamically using some of the common resources).
- **Service agnosticism:** The slice can support a single service (1:1 mapping) or multiple services (1:n mapping). In the latter case, the service lifetime management should be separated from the slice lifetime management.
- **Isolation:** A slice has to provide physical or logical isolation from other slices that include security and privacy aspects.
- **Functional structure:**
    - The internal architecture of a slice should be composed of Data Plane, Control Plane, Application Plane, Management Plane and Slice Resources, each with specific roles in the data forwarding, control of the connectivity, application level communication, management of the virtual network and the underlying resources used. The planes can also be sliced partially having a common part that is used by many slices. In specific cases like the legacy solutions and some RAN solutions, the slicing may concern only selected system planes. The planes can also be sliced partially, having a common part that is used by many slices. Such an approach is justified by technological limitations, for example, concerning slicing of the control plane in RAN, on the other hand, such an approach may lead to more efficient slicing.
    - The slicing may concern only single or some domains of the multi-domain solution (e.g. RAN domain, EPC domain, etc.).
- **Data plane performance:** The sliced Data Plane has to provide mechanisms that will guarantee the high performance of data plane operations.
- **E2E Slice:**
    - An E2E slice can be composed of slices belonging to different technological or administrative domains (multi-domain slicing), like access network slice (e.g. RAN), mobile core slice, transport slice, etc. Each type of slice has to be appropriately described for the purpose of the E2E slicing.

  - An E2E slice can be composed of many slices (called also sub-slices) by slice stitching – this approach provides compatibility with the NGMN network slicing approach.
  - A sub-slices contributing to the E2E slice may be existing ones or created on demand. The operation of composing of the E2E slices should be recursive.
  - A slice should be able to compose the E2E slices of slices or resources that belong to different domains.

- **Legacy systems compatibility:**
  - The solution should be software oriented (use NFV, SDN, cloud technologies, etc.). However, the slice may also accommodate network functions implemented in hardware (PNF according to the ESTI MANO terminology) or the existing legacy hardware subsystems. This is motivated by the huge amount of installed mobile network hardware nodes and the expectation that network softwarization will be a gradual process.
  - Slice-agnostic terminal operations should be allowed. In such a case, the UE is not aware of the existence of multiple slices, but an appropriate proxy provides the required traffic redirection (to VoIP slice, to VoD slice, etc.).
- **Slice discovery and selection:**
  - Due to the dynamic lifecycle of slices, a slice discovery and selection mechanism have to be implemented.
  - The slice selection should be based on policies.
  - The slice selection can be driven by a user equipment (statically or dynamically, based on a list of slices provided by the orchestrator) or by the slice operator (i.e. the network and not the UE, which is responsible for slice selection). A negotiation between the network and UE during slice selection process should be also possible.
- **Dynamic slicing:** The UE should be able to attach to multiple slices and request creation of a slice.
- **Policy-based slice management:** slice operations and lifecycle management should be based on policies. That is being currently the only dynamic mechanism that is able to encompass dynamically different administrative requirements.
- **Management functionality scalability:** Slices management and orchestration should be scalable, allowing dynamic and efficient allocation of resources to slices and dealing with multiple domains.
- **Compatibility:**
  - The 5G!Pagoda architecture should accommodate 3GPP approaches (mobile core slicing, slice selection mechanisms, the inclusion of not sliced RAN, full E2E slicing).
  - The 5G!Pagoda architecture should be aligned with the 5GMF architecture as far as possible.

- **Roaming:** If a slice is composed of distant RANs, a roaming mechanism should be deployed within the slice.
- **Instantiation:** The instantiation of the 5G!Pagoda architecture to a specific use case should address in the most performant manner the requirements of the specific use case.

4.2. Generic business requirements

The use cases discussed in sections 3.2 and 3.3 of D2.1 [53], clarified that 5G!Pagoda targets multiple stakeholders. The network slicing is a key enabler for a new business model that comprises of the infrastructure providers, slice providers and operators, application providers and end-users. The existence of multiple stakeholders impacts the architecture in the following way:

- There have to be defined interfaces between different owners/operators.
- There has to be a business interface that will allow the slice operator (tenant) slice management, especially the creation of the E2E slices in multiple administrative domains.
- The slice operator should have high-level management capabilities that include: PBM, configuration, security operations, accounting and performance monitoring.
- Fault and performance management should be automated and performed by the slice orchestrator.
- The orchestrator operator, slice operator or the end-user should be able to create a slice. In the latter case (slice on-demand), the slice should be advertised, and for some of them, their roll-out time can be a critical one.
- There has to be an administrative interface between the slice operator and slice provider.

4.3. 5G!Pagoda scenarios specific requirements

As discussed in the section 3.4.9 of the deliverable D2.1 [53], the 5G system is expected to provide various capability sets depending on the use case. The common capabilities have been described in the previous sections of this chapter. The additional architecture-related requirements that have been identified in D2.1 [53] include:

- The 5G systems equip computational resources at network edge entities for a connectivity path (less data traffic in the core network, lower delay, secure local communication, etc.).
- The service providers should operate on two types of strata, i.e. physical resource layer where various services (implemented by software) can run and the software layer, which controls and manage the services.
- The software layer entities have the capability to be relocated on demand over the physical resource layer and across different domains (business regions).
- The 5G system should manage the mapping between users and capabilities belonging to the users. It should track the user subscription and their capabilities dynamically.

- The hardware resources assigned to the system component implemented in software should be updated (added, removed and replaced) when the capability set gets updated (the scaling feature).
- The 5G system should have a redundancy mechanism that provides individual redundancy levels that can be enforced by service providers (as a part of policies).
- The 5G system should have a mechanism for providing the E2E QoS service warranty. Specifically, the multiple stakeholders must be able to interwork for providing E2E service guarantees.
- The 5G system resource allocation to software system components should be updated in a second or shorter.

4.4. Design features

The main goal of 5G!Pagoda was not to develop 'yet another slicing architecture' – as it has been described in Chapter 3, there are already many attempts to solve this problem. The project ambition, however, is to develop an architecture framework that provides harmonization of the existing architectures in a converged manner. On the other hand, the ambition was also to develop in more details mechanisms that are missed or not fully addressed yet by other approaches. It is worth emphasizing that the 5G!Pagoda architecture will accommodate the 3GPP efforts related to slicing. However, we will go beyond 3GPP approaches by integrating in a graceful manner other network components which are not in the direct goal of 3GPP standardization. The developed concept can be seen as a first step toward the convergent slicing that deals not only with mobile networks but also with transport ones, etc.

As already described in this deliverable, a slice offers a dedicated, networking solution that is tailored for the needs of a specific application or several applications. In general, the slice, despite being realized in software, should provide operations comparable to hardware-based networks. It has to be emphasized; however, that a network slice can offer much more than a 'pure network' – it can also include application layer functionality.

A software implementation of network slices atop of common hardware resources provides a high level of flexibility and the dynamicity of slice functions and slice oriented operations. Due to some constraints and the existence of legacy solutions, 5G!Pagoda also considered solutions based on hybrid slices, i.e. composed of both, software and hardware subsystems.

In 5G!Pagoda we have followed the classical telecommunication system architecture pattern, in which the system is composed of Data, Control, Application and Management Planes. The Data/User Plane is controlled by a clearly separated Control Plane, following the principles of carrier-grade telecom networks. On top of the Control Plane, the Application Plane is established. It consists of different application enablers (slice exposure) in order to offer the appropriate services to specific applications. The Management Plane is added to the slice-based network, enabling the appropriate operations for all the other planes and their resources.

The mechanism of independent slicing of any of the system planes (per plane slicing) can be deployed in order to simplify the design of slices. We have introduced the split between Common Slices and Dedicated Slices. The Common Slice in cooperation with a Dedicated Slice (by 'vertical concatenation') provides the complete functionality of a slice. We used the Common Slice abstraction to denote a set of functions which are commonly used by multiple Dedicated Slices. The approach provides the benefit of reusability of common functions at the same time, making the Dedicated Slices footprint smaller. Moreover, the approach enables the use of hardware subsystems that cannot be softwarized but can be programmed to allocate its resources or functions to a slice. The Common Slice can also act as a Data Plane, while the requested slice does not exist yet.

The 5G!Pagoda architecture in-line with NGMN it also enables the creation of an E2E slice as a combination of domain slices. Such domain slices can be RAN slices, EPC slices (e.g. 3GPP DÉCOR), IoT slices, transport slices, etc. In order to create the E2E slice, there is a need to describe the slices in a uniform and readable manner.

The slicing of RAN is problematic. In RAN it is possible to share the radio resources among slices (i.e. the Data Plane), but slicing of Control Plane and the Management Plane is hardly possible. Some of the Control Plane operations have to be common due to the mechanisms implemented in the physical radio layer. An exception is a Cloud-RAN concept, which fully enables RAN slicing (in case of LTE at the BBU level). In general, the RAN slicing with shared Control Plane has to be accepted. In such cases, the Control Plane can be fully or partly shared, but the Data Plane can be fully sliced. Due to the mentioned issues, the Dedicated Slices can use services of all planes that are offered by so-called Common Slice of RAN.

The mechanism of slice selection can be implemented in different ways. In some cases, slice selection can be implicit (for example, in the IoT case); in some of them, the existing legacy networks can be used. Another considered option is the use of specially designed lightweight control plane and distributed mechanisms for slice selection. In the Network-on Demand approach, slices can be created dynamically as requested by the end-users.

In 5G!Pagoda both, slice and resource level operations are automated. The slice oriented operations are logically centralized and handled by the orchestrator(s) whereas sliced network management allows the slice operator (for example a vertical) to manage his network. In the case of the network-on-demand scenario, the management of the slice created by the end-user should be as much automated as possible.

It is widely assumed that the sliced network instances will be implemented on top of the virtualized infrastructure that offers connectivity, storage and computing resources using the ETSI NFV approaches. There are, however, some reasons that make the generic network slicing and the mobile network slicing different from the classical ETSI NFV approach:

- The E2E slicing should deal with the multi-domain issue that is so far not supported by the ETSI MANO architecture.
- It is generally required that each slice should be isolated from other slices. Such isolation can be done in many different ways, and one of them is the isolation at the slice resource level. The

latter approach impacts the architecture in a way that resources allocated to a slice have (optionally) some resource isolation oriented mechanisms, like data/transmission ciphering, etc. The mentioned mechanisms are, in general, not compatible with the ETSI NFV framework.

- The installed base of mobile networks worldwide is enormous and therefore when considering slicing in future mobile networks, Telco operators desire to include the devices in their future solutions that use slices. It is not only about single devices (PNFs according to NFV terminology), but also about (for example) whole RANs.

5. 5G!Pagoda reference architecture

In this chapter, the 5G!Pagoda architecture is described. The described architecture is a so-called reference architecture, which means that it can be instantiated in different, implementation-specific ways. This approach is motivated by the many different usage scenarios of network slices. It is a top-down approach based on analysis of network slicing relating requirements and use cases. For some use cases (e.g. videoconference, streaming) relatively simple architectural approach is needed, for another one (e.g. MVNO) much more advanced mechanisms should be implemented. Therefore, the proposed approach allows for the selection of components of the architecture according to the intended use case.

The architecture description is split into two parts:

- the part related to the support of network slicing operations (slice selections, subscriptions, etc.). It also includes the description of the generic slice structure;
- related to the orchestration and management architecture that covers also multi-domain orchestration.

In this deliverable, there are no described management and orchestration procedures that were described in the deliverables of WP4. We also do not describe in this deliverable the issues related to 'sliced' solution (e.g. EPC implemented as a slice) except cases when the sliced solution has an impact on the architecture. Such issue concerns, for example, RAN.

The described reference architecture deals with all planes of sliced systems and adds functional components that are required in the multi-slicing environment. We did not provide details on the interfaces (with some exceptions), but we described the required functionalities of the interfaces. The reasons for that are the following:

- The functional components of a slice that belong to control, data or application plane already have interfaces. They are related to a specific networking solution (e.g. EPC, RAN, etc.). There is no way and no need to redefine them and make them generic. Therefore, they should be implemented as they are. Also, due to the implementation of functions in software, the message bus, as recently proposed by 3G PPP in the context of 5G architecture, could be used for the communication between them.
- The management and orchestration functional components of our architecture are software-based and 5G!Pagoda consortium has decided not to use interfaces, but the Service-Oriented Architecture (SOA) instead. It means that the functional components will use a common message bus in order to exchange messages between the functional entities. Therefore, the functional components will communicate via message broker instead of using predefined interfaces. The major difference between service-based and point-to-point based approach is that in the former model, service queries a service repository function, which implements service discovery functionality (similar to the 3GPP NRF function described in the 5G core architecture). By dint of service discovery, entity services may detect other communication peers, what makes fixed reference points avoidable. The basic architecture of mentioned approaches is composed of publishers, subscribers and message brokers (message buses). The

approach is in line with orchestration solutions such as Open Baton, or ONAP. However, the service-based architecture may be translated into traditional reference point architecture.

- As much as possible, we tried to align our approach with the ETSI MANO framework, ensuring compatibility for some of the orchestration related functional component interfaces of our approach. It is worth to say that ETSI MANO framework does not define protocols that should be used between its functional components. Moreover, they do not address multi-domain orchestration. The work MANO is in progress, and new mechanisms related to orchestration performance and security are added.

5.1. Functional architecture of the system

The 5G!Pagoda reference architecture for a single-domain slicing is presented in Figure 10. In the figure, the following basic blocks of the architecture can be found:

- Infrastructure
- Common Slice (CS)
- Dedicated Slices (DS)
- Domain-Specific Slice Orchestrator (DSSO).

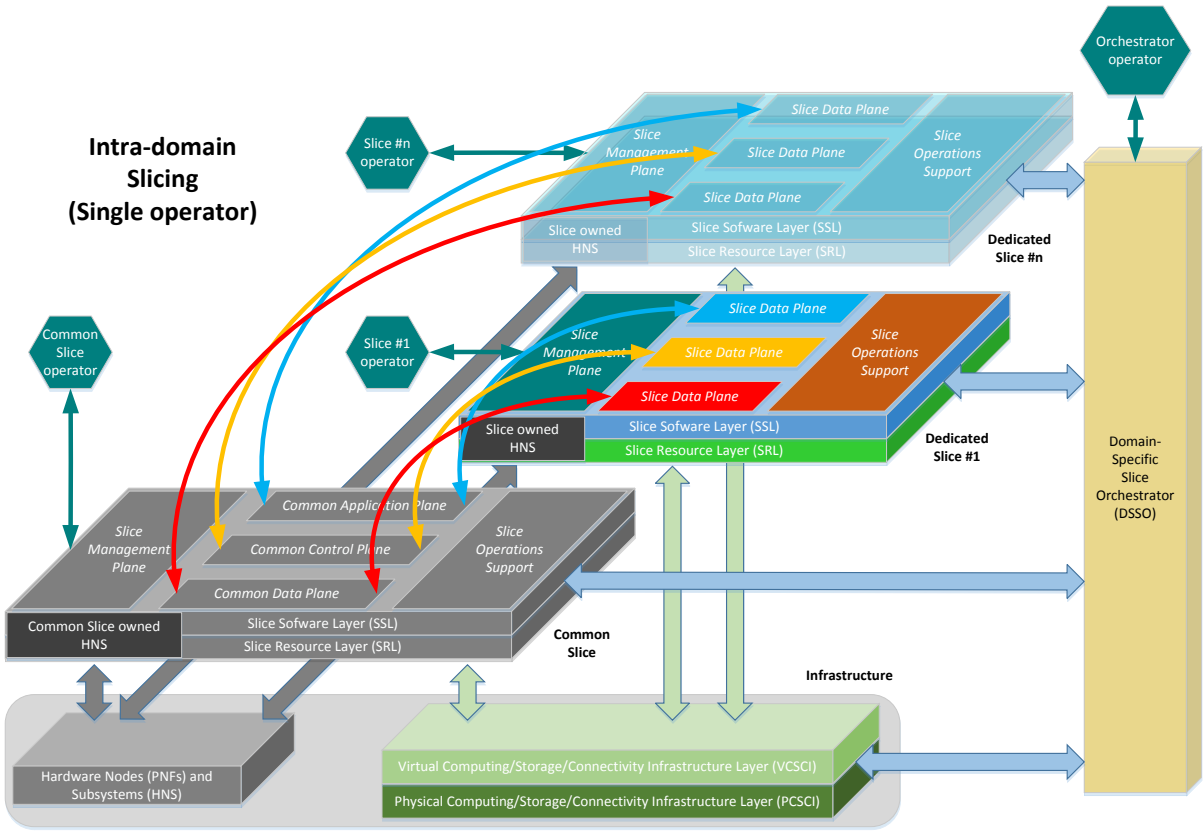

Figure 10 – Instantiated 5G!Pagoda slices on top of the same infrastructure

The **Infrastructure Layer** consists of resources that are separated into two main groups:

- Virtual Computing/Storage/Connectivity Resources (i.e. interconnected data centres) that are built atop of respective Physical Resources.
- Hardware Nodes and Subsystems (HNS) that can be used by the Common Slice and Dedicated Slices. HNS may include RAN or Radio Nodes (eNodeBs), specific transport nodes, etc. These nodes or subsystem resources can be allocated to slices, but their resources are not virtualized, i.e. cannot be expressed in the form of connectivity/storage or computing. In ETSI NFV terminology, they can be seen as PNFs. However, they can also include PNFs that form complete networking solutions or can be seen as in-software implemented functions that are accessible via API.

Both types of resources mentioned above can be dynamically allocated to slices. The allocation of infrastructure resources to the Slice Resource Layer is done by the DSSO and is optimized during the whole life-cycle of the slice. The architecture of DSSO is specified in Section 6.

The architecture enables two different generic slice types: the **Common Slice** and the **Dedicated Slice**. It has been assumed that there is only one Common Slice per technological domain, and there can be as many as needed Dedicated Slices.

In the presented concept, the **Common Slice** is an abstraction of grouped functions (VNFs, PNFs) that can be reused by Dedicated Slices created later on. Such abstraction enables simpler management of the shared functions and enables recursive operations. For example, a new Dedicated Slice can be combined with the existing Common Slice in order to implement its functionality. It can be said that the Dedicated Slice is a client of the Common Slice. By recursion, we may update the Common Slice functionality. The Common Slice may offer four types of services:

- A set of functions that provides the Dedicated Slice users with information about available slices and subscribes the users to these slices based on their privileges and operator policies, etc.
- A set of Data Plane, Control Plane and Application Plane functions that can or have to be used by respective planes of Dedicated Slices that cooperate with this slice. These functions may include, for example, handover handling, MAC scheduling procedures at the RAN level.
- Basic communication means for the end-users of the system which are not attached to any dedicated slice (in a similar way to the default bearer in LTE). This mechanism will be used, if the required dedicated slice cannot be found or before an on-demand slice is created.
- A complete 'network' – in such case, the Dedicated Slices are used for service implementation only.

The **Dedicated Slice** combined with the Common Slice provides a complete network and service implementation (in some cases the Common Slice can be non-existent) – they have different roles and in some cases are complementary ones.

The existence of both slice types is motivated by two reasons. The first reason is the inclusion in the architecture of solutions that cannot be fully sliced (for example the Control Plane of RAN, legacy solutions, networking solutions that are out of the operator control capabilities in terms of

slicing). The second lies on making the Dedicated Slices lightweight by providing the commonly used functions via the Common Slice. It is worth emphasizing that from the flexibility and isolation of slices point of view, it is preferable to minimize the Common Slice. It has to be mentioned, however, that if Control Plane is common (cf. DÉCOR), the other planes, like the Data or Application Plane (for example in the MEC case), can be mostly implemented as Dedicated Slices.

Many slices can be instantiated and run in parallel on the same infrastructure, which shows the importance of proper allocation of resources to slices. As the resources are virtualized, they can be allocated to the slices dynamically during runtime and placed in different parts of the infrastructure (if required/possible). As illustrated in Figure 10, each Dedicated or Common Slice has a Slice Resource Layer that consists of isolated and dynamically allocated generic storage/computing/connectivity resources, as well as hardware resources allocated to the slice (HNS).

The slice orchestration allocates resources to slices during their deployment, but also to adapt them to different usage conditions (how the users of the slice behave) and to the exceptional network situations related, to faults or emergency situations. The life-cycle orchestration of slices provides automated FCAPS, and its role is different from the internal slice **Management Plane** operations, which are specific and private to the slices and their services. The Management Plane of a slice is seen as generally lightweight and cooperating with the slice orchestrator to realize the management goals. In general, it has been assumed that the orchestration architecture will be hierarchical and composed of multiple, typically single domain orchestrators, DSSO. The **DSSO** is responsible for the life-cycle of slices and the optimization of their operations during slice run-time. It is also used in the multi-domain (E2E) slicing. Each of the technology domains has its own Resource Orchestrator (RO) that is a part of DSSO. The existence of per domain RO is motivated by different types of resources that require different operations. It is foreseen that, for example, for the transport network, an SDN WAN controller will act as the RO whereas, for a data centre, a typical VIM (e.g. OpenStack) from the perspective of ETSI MANO may be used. The detailed description of DSSO is provided in section 6.

5.2. Generic network slice structure

5.2.1. The internal architecture of the Dedicated Slice

The Dedicated Slice can offer a full set of services tailored for a specific set of applications (a slice can provide a single application or many applications). In some cases, these services will be offered with the cooperation with the Common Slice.

Each slice is running on top of the **Infrastructure Layer** that consists of virtual resources (computing, storage, connectivity and virtual radio) as well as hardware ones (PNFs according to NFV or whole hardware-based solutions). These resources are allocated on demand through orchestration architecture.

5G!Pagoda proposes a slice structure that is composed of three main blocks:

- **Slice Core (SC)** functions, i.e., functions that implement the requested slice functionality (for example, EPC).
- **In-Slice Management (ISM)** functions
- **Slice Operations Support (SOS)** functions.

5.2.1.1 Slice Core

The Slice Core part of a slice is decomposed **Application Plane, Control Plane and Data Plane** that form a functional network that can be managed by the slice operator (tenant) using the **Management Plane**. The slice oriented lifecycle operations, and automated FCAPS are performed by **Slice Operation Support** (SOS) functions that may have its internal Control, Data and Application Planes entities, that support respective plane operations. It is assumed that the SOS structure is common for all dedicated slices but the functionalities of internal SOS blocks may differ.

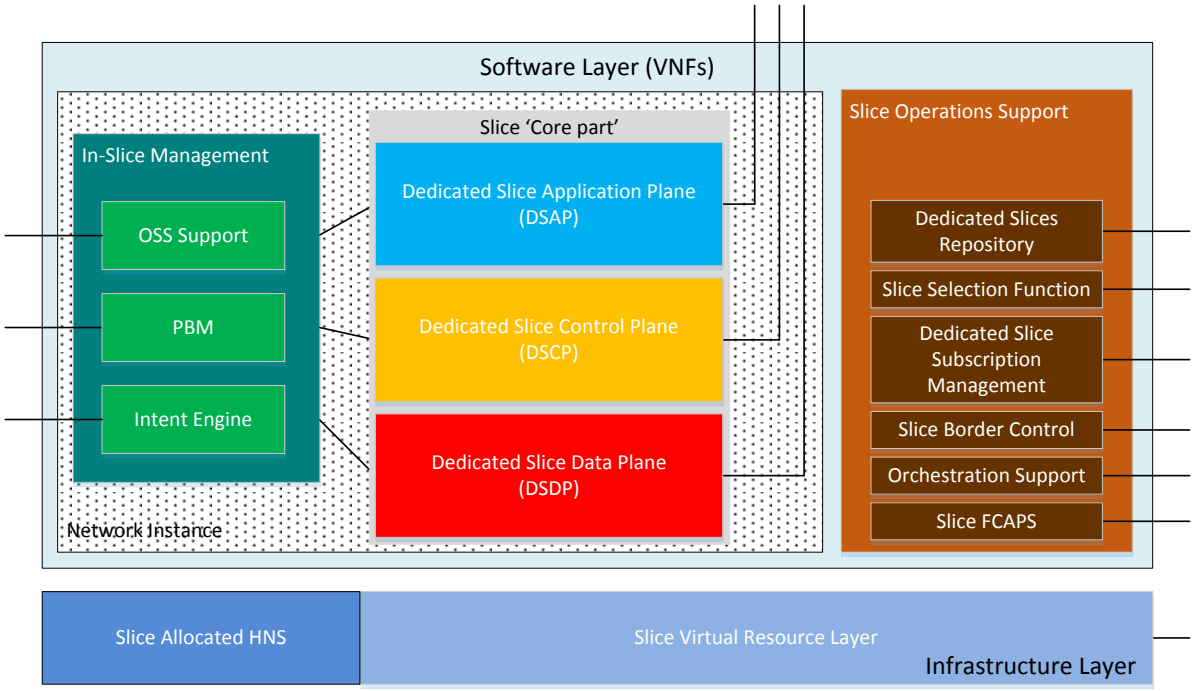

Figure 11 – Dedicated Slice internal architecture

The Dedicated Slice Data, Control and Application Planes functionalities are use case (EPC, RAN, transport) dependent and they can provide:

Data/User Plane:

- Data forwarding components, enabling the forwarding of the data packets between the different entities, pertaining to the same or to different slices, including functionality needed for load balancing and high availability;
- Data storage and processing components enabling the aggregation and dissemination of information;
- Data processing entities enabling the changing of data formats for proxy nodes and data aggregation, derivation, splitting of the data path for back-to-back;

- Data analytics enabling the generation of active and passive insight into the specific data communicated;
- Data plane security enabling the encryption of data traffic between different entities;
- Data Plane components related to the connectivity (e.g. Serving Gateway User Plane – SGW-U, Packet Data Network Gateway User Plane – PGW-U);
- Data Plane components related to the content routing and storage (ICN, CDN);
- Deep Data Plane programmable components are enabling the sharing of the available resources to implement the previously described data plane functions.

Control Plane:

- Control of the forwarding plane for functionality such as routing and forwarding control, including security, load balancing and high availability;
- Control of data storage and processing (data path) components;
- Control of authenticated connectivity over RAN and packet core system;
- Control of inter-slice communication;
- Control of the applications deployed at the data plane level.

Application Plane

- Enables the deployment of advanced services (i.e. Application Servers according to 3GPP terminology);
- It may consist of the services as well as service enabler that are used to compose a new service.

5.2.1.2 In-Slice Management

The In-Slice Management (ISM) functions of each slice are used by slice operator (for example a vertical) for managing the network slice and its services. It has been assumed that these operations should give as much as possible flexibility of configuration of the slice and providing to slice operator the information about achieved slice KPIs. However, all the operations related to troubleshooting of slice operations should be automated (for example fault management) and/or handled by SOS. In case of the slice is triggered by the end-users (the network on-demand case) the functionality provided to user should be limited to basic operations only or offer a predefined set of management functionalities with a basic interface to the user.

The **Dedicated Slice ISM** is used by slice operator in order to monitor slice behaviour, change its configuration and provide billing support. It is assumed that this type of management is lightweight and comfortable and most of the management tasks, including most of FCAPS functions, are handled by the orchestrator. The Dedicated Slice ISM consists of the following functional entities:

- **OSS Support Functional Component.** For the operations regarding installation, the slice O&M should be able to request on demand the addition of a new network function to a running slice or the reconfigure the existing one. This part provides also information about

allocated resources to slice, their consumption, SLA fulfilment and alerts. Part of this component is a portal that allows slice management and interactions with slice orchestrator.

- **Policy-Based Management** is a tool for high-level management of sliced network functions by slice operator without the need of the detailed specification of the low-level operations, necessary to achieve the management goals.
- **Intent Engine** translates the high-level description of slice operator management goals into detailed low-level operations. In order to achieve this goal, it cooperates with PBM and the components of other planes of the dedicated slice as well as with the orchestrator. It has been assumed that the OSS Support functional entity is initiated for each slice and supports slice specific operations interacting with other Dedicated Slice components, whereas the slice orchestrator provides slice functions agnostic orchestration.

Using the Dedicated Slice ISM, the slice administrator is able to:

- Request a services catalogue from the Business Service Slice Orchestrator (BSSO).
- Select and configure a slice based on the services provided in the catalogue.
- Trigger the deployment of the slice according to the configured services.
- Administrating PBM in both the orchestration and within the slice as much as allowed and possible through policies within the policy engine. Most probably this will be done through pre-defined templates.
- Administrating the services within the slice through policies within the slice specific OSS as well as through user profiles.

The slice management scalability may have multiple dimensions:

- It has to deal with the number of the 'slice events' that need to be handled at the same time and related to the management of slices and their resources. If the number of events increases, unified management will be too difficult to ensure.
- Frequency of management objects states changing. The objects which have only static or semi-static states do not influence the management performance; on the contrary, the objects, which state changes frequently, greatly influence the management performance.

In the 5G!Pagoda architecture, it is assumed that the slice management is split between ISM and orchestrator(s). The slice specific management is done by ISM, whereas the orchestrator is responsible for the management and orchestration of virtualized resources.

Basically, the management of resource scaling is made by the orchestrator.

5.2.1.3 Slice Operation Support

The **Slice Operations Support (SOS)** functions are focused on slice level operations, which include: operations related to subscriptions to a slice, slice combination (stitching), slice lifecycle support and automated slice management based on policies. The SOS functions may deal with all planes of a slice in order to achieve its goal. Each slice SOS offers the same set of services, some of which in specific implementation cases can be null. These internal functions of SOS are the following:

- **Slice Repository** that contains a repository of active slices to which the end-users that are allowed can subscribe. This is a local copy of Slice Repository of the Common Slice (if this slice is in use) created for the sake of scalability and reliability. It may intentionally consist of a more limited number of slices to which the slice user can roam (for example only these ones which are owned by the operator of this Dedicated Slice).
- **Subscriber Configuration Management.** One, a specific type of configuration are related to the subscription profiles. Although it is not foreseen that the subscription profiles will be frequently modified during the runtime of the slice, two major operations have to be considered:
    - Completing the database information for authentication, authorization and access control rules (i.e. the subscription profile) for all the users at the deployment of the slice; this highly depends on the number of users foreseen to connect as well as on a possible previous completed database with such subscription profiles.
    - Adding new subscription profiles during runtime on-demand.
- **Slice Border Control (SBC).** The SBC functions can be included as both the Control and the Data Plane functions and play a key role in multi-domain operations, slice traffic redirection, inter-slice mediation, etc.
    - The **SBC-Control Functional Entity** ensures the interconnection at the protocol level between the different components within the slices. It may include for Diameter peering a Diameter Router Agent (DRA) and for IMS communication a Session Border Controller, both with the role of peering with the foreign domain, appropriately routing the requests to the other domain, as well as the anonymization of the private slice information and the encryption of the communication. It may be also used for information about domain-specific properties, expose to other chained slices abstracted network topology, etc.
    - The **SBC-User Functional Entity** ensures the proper interworking between the data path components in case the communication requires other protocols than IP only. The functionality may include GTP peering (as in the case of packet core roaming), SFC peering, multimedia transcoding and content compression as well as possibly the encryption of the data path.
- **Slice PBM** addresses the slice-related issues, including inter-slice connectivity management policies that can be adapted depending on the momentary network function placement. For example, if the functions of two components of the different slices are co-located, it could be better to establish between them a connection compared to components which are located in different data centres.
- **Slice FCAP** functions used to implement slice level FCAPS functionality towards complex events processing and NFV environment-specific adaptations (e.g. actions for re-creating the network on components' failure, configurations depending on the dynamic network as established by the NFVO during the runtime, differentiated accounting rate depending on services, time of day, etc., enhanced performance and security optimizations through the adaptation to the environment such as deployment of more appropriate VNFs to the momentary situations, reconfiguration of the components depending on the momentary topology of the system for

increasing the resilience and the availability, ensuring of the service KPIs across deployments on top of heterogeneous infrastructures).

- **Slice Selection Function (SSF)** is used for the allocation of UEs to slices. The Slice Selection Function in a Dedicated Slice complements or replicates the Slice Selection Function in the Common Slice. The process can be based on UE preferences, operator decisions or a result of negotiation between the UE and network. A slice discovery is part of this function. It is also used by the UE for the creation of a slice on-demand if such operation is allowed. Note that in the case of slice stitching, only the 'edge slices' will have this component. The Slice Selection Function should be backward compliant with 5GC mechanisms [35] (e.g. S-NSSAI), but it has to be extended. The mechanisms should allow for attachment of slice unaware UEs, UEs with a predefined slice (for example in SIM), UE able to attach to multiple slices, slice creation on UE demand if a required slice cannot be matched.

5.2.2. The internal architecture of the Common Slice

The Common Slice is a slice that can play multiple roles in the architecture:

- For a specific technological domain, it can be used as a 'default' network slice which offers services to the users, which are not yet attached to the Dedicated Slices.
- It can be also as temporary slice used before the 'network on-demand' slice will be created (during the Dedicated Slice bootstrapping).
- In a reduced form as 'Control Plane only slice,' it can be used as a signalling path for slice redirection.
- It may offer some services that will be used by dedicated slices. Such an approach makes the Dedicated Slices simpler and shortens their booting time. This approach is especially important in the case of Network-On-Demand slices.
- It is mandatory in solutions in which there exist legacy solutions which planes are non-splittable into dedicated slices (at least at the plane level), or depend on a hardware node. Such a case can be 4G RAN based on eNodeBs, etc.
- As a permanent database of information about all slices, subscriptions also used for legal intercept. This information can be copied to Dedicated Slices (it is about Slice Repository, subscriber management, Slice Selection functions rules, etc.).
- As a basic mean of communication during emergency services in order to provide most users connectivity.

The Common Slice is seen as a permanent entity. According to research directions, the functionality of this slice should be limited. The distribution of Common Slice functionalities among Dedicated Slices improves system flexibility, scalability, reliability and isolation between slices.

The internal architecture of the Common Slice is similar to the Dedicated Slice. The main difference lies in its permanency and implementation of some functions in specialized hardware.

It can be assumed that the Common Slice in a minimal version will have neither Application nor Data planes (in such case it will be similar to 3GPP DÉCOR). The SSF is necessary in the Common Slice in order to initially connect users to the Dedicated Slices. The Common Slice ISM is separated from the management of Dedicated Slices

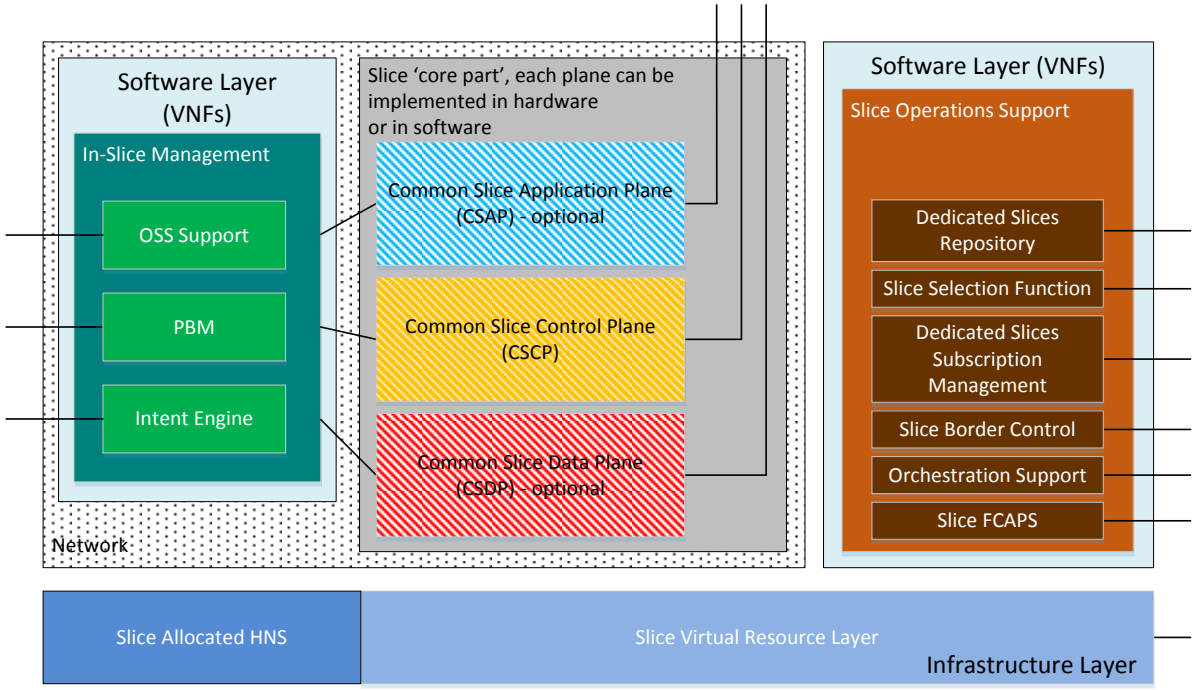

Figure 12 – Common Slice internal architecture

5.2.3. Slice description and capabilities exposure

5.2.3.1 Slice descriptors

There are at least two reasons that require a description of slice functions. The first case is related to slice selection by the UE, whereas the second one is related to slice stitching that can be used in a multi-domain environment. In the latter case, a direct implementation of a multi-domain E2E slice, based on a single slice template is possible, but it can be much more complicated and time-consuming than the mentioned stitching of independently created domain slices. The creation of E2E slice can be recursive, and for example, several transport slices can be chained together as a single transport slice that is a part of an E2E slice that also has RAN and EPC domain slices. Generally, each E2E network slice should be composed of Radio Access Network, Mobile Core Network and Transport Network.

In order to deploy an E2E network slice, the orchestration system needs to use a common and unified model of slice description. Moreover, each network slice shall be uniquely identified in order to allow end devices to select appropriate network slice that will provide specific network services. In fact, in order to provide global interoperability, there is a need for standardization of slice description and identification, and it has already been addressed by 3GPP. In 5G!Pagoda, we made a step forward in order to solve the problem, at least for the purpose of our implementation. In general, it is expected that this problem will have to be solved by standardization organizations in order to be accepted worldwide. {3GPP two sentences to be added here}

In the heterogeneous networking environment, there exist various network slice types. In order to provide easy operations at a slice level, a technology-specific categorization of slices is necessary. This categorization enables the creation of a combination of slices as sub-slices and inter-slice operations. The slice types include:

- Access Slices (AN) and their subset Radio Access Network (RAN) Slice,
- Mobile Core Slices,
- Transport Slices,
- Composite E2E slices, being any combination of Access, Core and Transport Slices.

This technical categorization of slices impact construction of slice template that has to be domain technology specific.

From a business point of view, there is another way of categorization of slices: service-specific categorization. Slice types of this category may include:

- Generic Service slices that describe slices tailored to generic services such as IoT or CDN;
- Service Specific slices (YouTube, Netflix, Skype, etc.), which needs specific parameters of QoS, security, placement, etc. These may be owned and operated by 3rd parties.

These categorization impacts slice identification scheme. We assume that end- device should be able to request access to network slice tailored to specific service. It is worth to note that a particular network slice instance can be created to handle a specific service or it can be used for handling of multiple services.

5.2.3.2 Slice capabilities exposure

In many network slicing concepts, there is a functionality called slice capabilities exposure that in general is used to access all the information concerning a slice through an API. This access should be provided to all 3rd party software; then one of them could request via the API for modification(s) of the network slice to realize a specific use case or task. In general, it can be seen as a sliced network API to services (as it is implemented in classical networking solutions). As stated earlier, in it assumed in 5G!Pagoda (according to NGMN slicing vision) that there can be a single service or multiple services per slice and these services may have lifecycle management separate from the slice lifecycle management. The mentioned slice capabilities exposure function can be seen as an API between the (sliced) network and their services. The specific solution of slice capabilities exposure functionality is Network Exposure Function defined by 3GPP within the 5GC. There is no doubt that some non-standardized solution should provide their API, but this API is in general, not dependent on sliced or not-sliced implementation.

The management operations in the 5G!Pagoda is allowed via slice Dedicated Slice ISM and Slice Operation Support functions. It has therefore been decided to use the OSS Support component of Dedicated Slice ISM to provide slice capabilities exposure functionality to slice tenants.

6. The architecture of slices orchestration

The orchestration architecture of 5G!Pagoda corresponds to the single-and multi-domain orchestration of slicing. The main functionality of the orchestration is related to the life-cycle management of slices and less to the slicing functionality itself, thus being the same, no matter which slice type is deployed and regardless of the domain. The 5G!Pagoda orchestration architecture is based ETSI NFV MANO framework that is extended to support multi-domain operations, slice specific functions and PBM of the orchestration process. The orchestration functions defined by ETSI MANO are described in this section together with the other new components introduced into the system, making references towards existing specifications when needed. The details of the orchestration have been developed within the work package WP4 of this project. The architecture of generic orchestration is presented in Fig. 13.

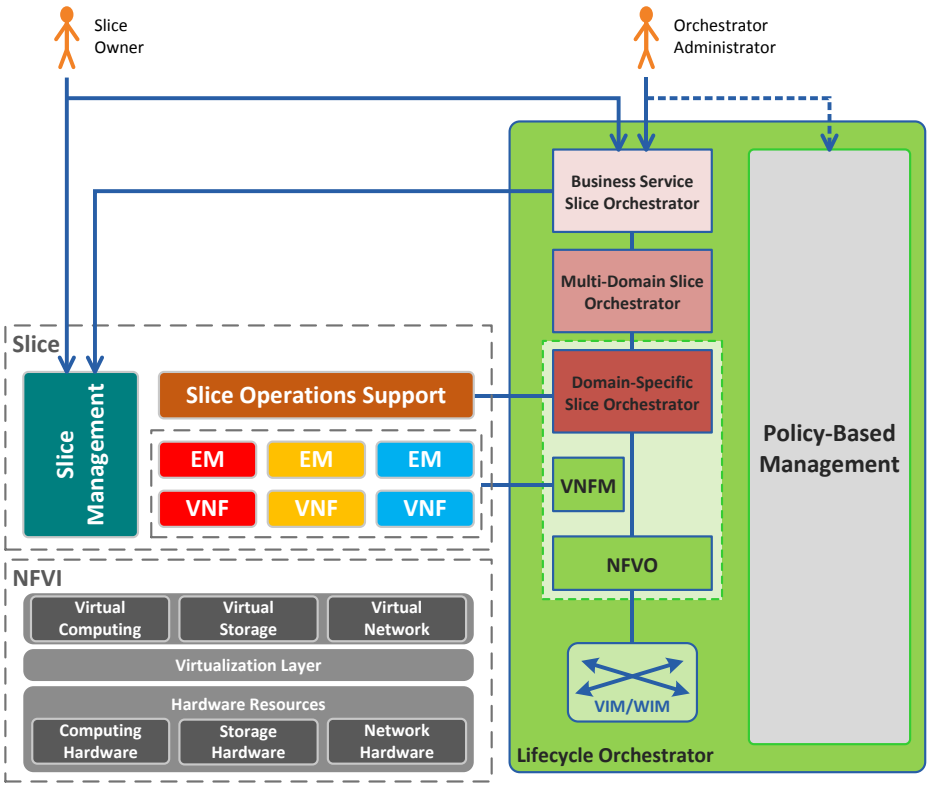

Figure 13 – Orchestration architecture

The functionality of the orchestration blocks is described below:

- **NFVI** –there are no modifications to the NFVI as defined in the high-level ETSI NFV architecture. However, a specific implementation of the virtual network is considered, covering deep data plane programmability and inter-data centre WANs.
- **VIM/WIM** – the Virtual Infrastructure Manager (VIM) is defined in the ETSI NFV architecture. Additional functionality of the VIM includes the capability to control the data plane functionality such as in the form of an SDN controller or an ICN or CDN information and content control in order to be able to provide a separation of the data plane when the data traffic is directly routed through the network (i.e. deep data plane programmability). Inclusion

of the SDN Controller is common in many 5G PPP projects. The Wide Area Network Infrastructure Manager (WIM) has the role in defining the virtual networks between different parts of a slice on top of common transport networks (i.e. the inter-data centre environment sharing rules).

- **NFVO** – the Network Function Virtualization Orchestrator has the functionality defined by ETSI NFV with its two roles of:
    - Resource Orchestrator (RO) enabling the brokering of the NFVI resources between the multiple parallel slices. The NFVO represents an aggregation point for the administrative domain for resources management. The NFVO communicates with multiple VIMs and WIMs and is able to allocate the resources appropriately across them.
    - Network Service Orchestrator (NSO) providing indications on how the system should scale and where the network functions should be placed following the Network Service Descriptor (NSD) information.

  Additionally, the NFVO is extended to support commands which result in the dynamic changing of the Network Slice Blueprint information. By this, the active service can be dynamically modified during runtime with additional actions compared to the static Network Slice Blueprint based decisions. For example, with this new functionality, new VNFs can be added during runtime to a running system (e.g. a more performant firewall in case of a network attack).
- **VNFM** – its role is defined by ETSI MANO specification. It should:
    - allocate appropriate resources to the VNFs or to delegate this operation to the NFVO
    - receive events on the completion of the specific operations and information on the dynamic configurations
    - dynamically configure through the Element Management (EM) the VNFs and control the execution of life cycle events
- **DSSO** – it is able to communicate with multiple NFVOs located into the same domain information on the specific slice split between the different NFVOs. The DSSO is similar to NFVO as defined by ETSI MANO, but with additional functions. Indeed, the latter is used to interact with the multi-domain slice orchestrator, for example, north-bound APIs for the multi-domain slice orchestrator to consume. The entity for handling the E2E orchestration has to be able to collect and to transmit the contact points, which enable the interconnection with other administrative domains including: IP addresses within the technology domains to connect different technologies from different administrator domains, IP addresses where virtual networks from one domain are bound to the other, technologies of the virtual networks binding the two domains and VNF level IP addresses in case the sliced network is explicitly settled across the domains. Note that IP addresses are allocated in the slice based on a common addressing schema, which is done through the orchestration, as in the case of any PaaS or NSaaS, but not for IaaS where only resources are allocated, and the tenant has to create the network through its own administrative means. The DSSO receives from the multi-domain slice orchestrator (MDSO) the information on the life-cycle management of the part of the slice, which is allocated to run on the specific domain. Note that this functionality is already offered

by the northbound API of the NFVO in the form of the processing of NSDs. It shall be noted that in the envisioned architecture (Figure 13), a DSSO and NFVO may be the same entity.

- **MDSO (Multi-Domain Slice Orchestrator)** – its main role is to provide a slice on top of multiple administrative domains if the multi-domain orchestration is applicable. The BSSO interact with the administrator of the slice in the management of the life-cycle operations and to offer the administrative entry points to the software elements from which the slice is composed of. It contains the following functionalities:
    - Receives from BSSO requirements concerning for a specific slice. The requirements may be received in a static description form such as TOSCA or an NSD file.
    - Establishes secure connections to the multiple DSSOs.
    - Acquires, if permissible, knowledge on the available resources in the specific administrative domains in terms of available infrastructure and available services (e.g. stored virtual machine images).
    - Negotiates with the DSSOs the resources and their locations to be allocated for a slice customer.
    - Makes decisions based on the requirements received on the split of the slice functionality across the multiple administrative domains.
    - Triggers the installation of the slice over the multiple administrative domains.
    - When the installation is successful, exchanges connectivity parameters between the different DSSOs to be able to stitch together the parts of the slice.
    - Announces the tenant through the business slice orchestrator on the successful installation of the slice as well as on the connectivity and management points.
    - Informs the tenant, through the business slice orchestrator and/or the slice-specific OSS, of any SLA breaches or any other types of major failures of the deployed slice.
    - Disposes a slice from the multiple domains.

  The MDSO implements Slice Placement Function that is responsible for allocating and interconnecting slice-specific virtual network functions (VNFs) according to resource constraints and service requirements, e.g. related to latency. The task of the multi-domain Slice Placement Function is the selection of domains covered by the slice. In particular, the set of domains must be connected so that there is a path from each domain to each of the other domains. For this purpose, it may be necessary to select from a set of alternative domains the particular domains to be on the path between two end domains, e.g. between the two domains of a UEs participating in a call. For the above purpose, the multi-domain placement function is required to have information on available resources and the cost of intermediate domains. Based on this information, the multi-domain slice selector can consult a domain selection policy function to, according to given policies, determine the domain to include into the slice. Domains can also be proactively covered by a slice, even if there are no UEs or transport needs connected to that domain. In particular, for the placement of services and VNFs, domains such as generic virtualization infrastructure providers, can be included. This may be, e.g. based on the price of hosting infrastructure, a central location or the availability of resources (e.g. bandwidth). The decision is made by the domain selection policy function. VNF Placement function may use different placement algorithms.

- **BSSO (Business Service Slice Orchestrator)** – it has the role of a portal to advertise the possible services, to trigger their deployment and in case of success, to transmit to the slice administrator the specific entry points to the new slice management. It is also used by slice tenant to reconfigure their slices after slice deployment. The BSSO provides the interface toward the slice operator (a tenant or vertical), including the API to query for availability of resources and blueprints, pricing information and status of deployed resources as well as the API for uploading new blueprints, deploying new slices and destroying slices. The BSSO is connected to one or several multi-domain slice orchestrators. The BSSO also interfaces the OSS/BSS of the domains in which it offers services. BSSO is responsible for slice admission control and negotiation considering service aspects. Typically, it collects abstracted service capability information regarding different administrative domains, creating a global service support repository. It also interacts with OSS/BSS in order to collect business, policy and administrative information when handling slice requests.

5G!Pagoda orchestrator(s) interacts with in-slice placed SOS components as well as Slice Management functions (that provide slice specific operations) in order to provide proper performance of slice operations (that includes dynamic allocation of resources) and handling alerts in an automated way.

5G!Pagoda-compliant orchestrators utilize TOSCA for specifying a slice. In particular, TOSCA is used in the interfaces between domains as the standardized slice descriptor. New node types, relationship and properties may be required to be defined in order to describe all necessary detail of a slice. The OASIS TOSCA standard language allows describing the topology, relationships and the management life-cycle of nodes in a cloud-based service. TOSCA is an extendable format, allowing the definition of new node types and linking to externally defined node types. Each node can be assigned properties – for example; a network node can have QoS properties. TOSCA is utilized to configure NFV services. It can also be used to define the relationships between the services.

6.1. Multi-domain orchestration options

To create an E2E slice, in many cases, it will be necessary to go across multiple technological or administrative domains. Creation of slices in such cases can be done in two ways:

- The first way lies on a direct creation of the E2E on the slice. In such a case, the first step consists of the integration of all domain resources via multi-domain resource orchestration first. Later on, the orchestrator that initiated the process deploys the network like in a single domain case. In this case, a single slice template is used. The approach can be applied to different administrative domains, but it has probably limited usage in domains with different technologies, as in which it would be difficult to exchange information among the orchestrators about each domain specificities (especially about the hardware nodes or whole subsystems).
- The second approach lies in the creation of per domain sub-slices and stitching them together. This case is hierarchical one, but per domain, orchestrators are loosely coupled with the BSSO. In this case, each domain orchestrator can use its own slice templates/Blueprints, and the BSSO can ask the domain orchestrator to create a slice that fulfils specific requirements (for example,

QoS). In response, the local orchestrator responds with a list of slices that can be deployed. The mechanism of local slice (sub-slice) creation is similar to the creation of slice on-demand by the end-user; it has to be noted, however, that in order to provide proper inter-domain operations, appropriate blocks of SOS, especially SBC have to be properly programmed.

It is worth noting that the creation of an E2E slice as a combination of sub-slices by the use of a parallel, per domain orchestration, accelerates the E2E slice creation. This solution is also relevant if the intermediate domain operator(s) does not allow, in its domain, creation and operation of a slice by the BSSO. Further, this model is also applicable in case when the existing slice is enlarged. Connecting (chaining) slices may impact the QoS of the E2E slice, and the process of orchestration of slices should take it into account. From the operation point of view, the following operations have to be implemented:

- Placement of a gateway between the neighbour slices. Any two interconnected slices may use different communication protocols. Protocol conversion gateway is necessary in that case, and the orchestrator should place an appropriate gateway server function in a suitable place. The conversion may deal with the data as well as control plane operations.
- Notification of the connection to a new slice. It is necessary to notify that a new sub-slice is interconnected in order to optimize its operations (for example another area is covered by the service). In some cases, a slice can handle such change and adapt to it, but in some other cases, the orchestrator should be involved in this process.

The slice exchange abstracts the interactions to domain-specific orchestration systems (DSSO in this case), implemented for the individual domains. Thereby, it enables each domain to have a single set of transactions to control and manage the network slice built over multiple domains. This can bring the individual domains a significant benefit when there are three or more domains interworking with the individual virtualization platforms and operational processes.

The mentioned functionalities are part of the Slice Border Control (SBC) of Dedicated or Common Slices. The SBC functionalities are typically split between the control plane (SBC-C) and the data plane (SBC-U). Details are implementation specific.

6.1.1. Resource aggregation in multi-domain orchestration

The Infrastructure illustrated in Figure 10 may be operated and managed by a single telecom operator, or it can be composed of multiple sub-domains operated by multiple operators and providers, e.g. telecom operators, MVNOs, cloud providers. As various types of slices are to be deployed, we note that such slices are deployed on top of various configurations of the infrastructure, where configurations are possible mainly because the resources are virtualized (i.e. dedicated and isolated) for a specific slice.

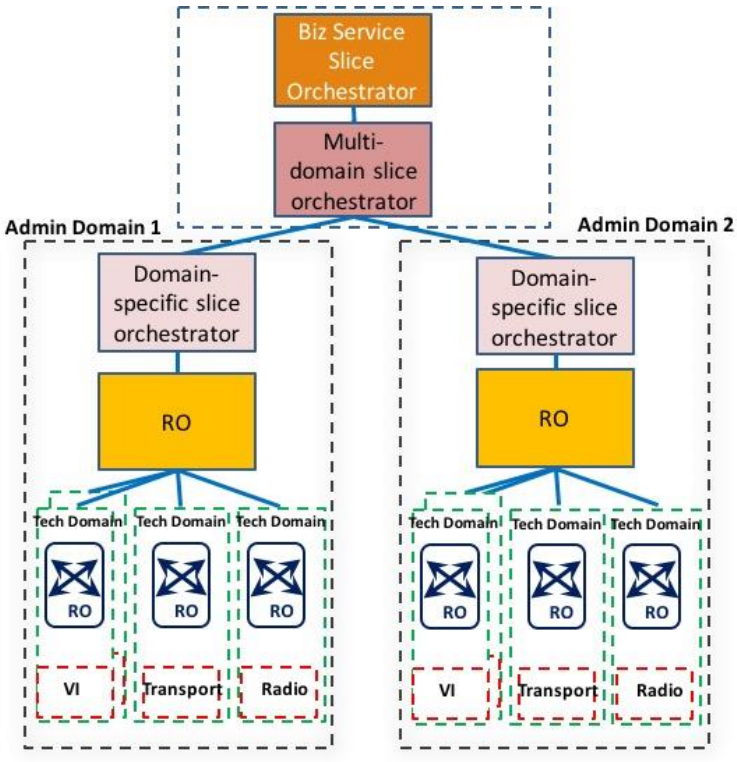

Figure 14 – Life-cycle orchestration for multi-domain architecture

To be able to use the resources in an appropriate manner across the administrative domain, an administrative domain resource orchestrator, named here **Aggregation RO**, is considered on top of the technology-specific ROs. Aggregation RO aggregates all the resources into the same RO, through this operation, making the resources transparent to the DSSO. Aggregator RO performs a hierarchical type of resource aggregation. For example, in Figure 15, an Aggregation RO can be considered for the radio technology domain across the different technologies and spectrum used. The Aggregation RO has an overview of all the resources inside an administration domain in order to place VNFs and create related NFFG (Network Function Forwarding Graph) across different resources, including multiple data centres, transport and different wireless accesses. The underlying reasons for this architecture are manifold. Primarily, with the global view of all the resources inside an administrative domain, the global RO can optimally place VNFs on the underlying resources, according to VNF's requirement, such as affiliation and special hardware requirements. Besides, inside an administrative domain, the environment could be multi-technology as well as multi-vendor. Such recursive orchestration enables clear separation of each domain's responsibilities and facilitates reliability and scalability as well as enables the enforcement of different policies in each domain.

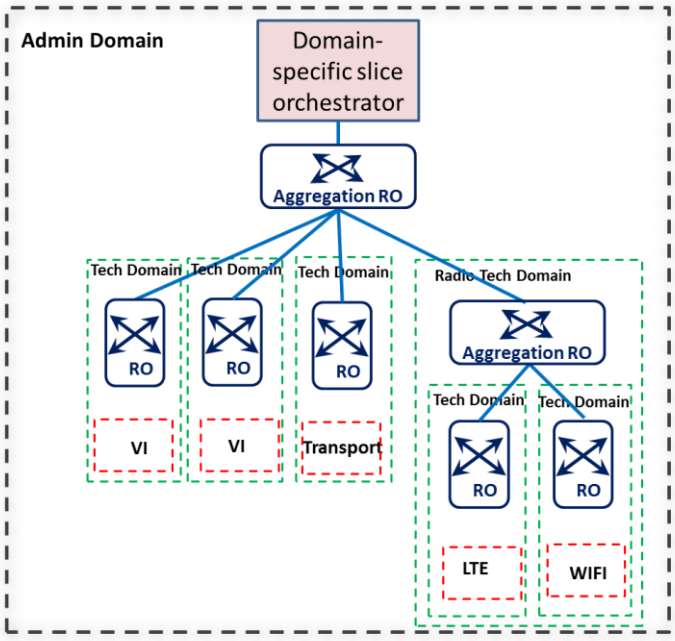

Figure 15 – Recursive resource aggregation and orchestration

The DSSO is in charge of E2E orchestration with the interaction of the global RO. To broker and to bind resources within multiple administrative domains, MDSO is used. The multi-domain slice orchestrator communicates with the orchestrators in the administrative domains to be able to stitch a slice across the multiple administrative domains, by using resources allocated in each of them.

6.2. Policy-Based Management of orchestration

The 5G!Pagoda management framework for network slicing is based on **Policy-Based Management (PBM)** approach. This model allows the system administrator to define a set of policies that govern the behaviour of a distributed system or network in a flexible and simplified manner. The key benefit of PBM approach is that network services, and functions can be managed to behave in the same way, even in the case of heterogeneous systems. Management entities will benefit from using policy rules that enable scalable and consistent programmatic control over the configuration and monitoring of network elements and services [55].

The adaptation of the life-cycle management (i.e. the life-cycle management can adapt the resources allocated to the specific slices depending on their momentary needs as well as through brokering the available resources). In NFV, due to the flexible virtual infrastructure used, which can scale on demand and due to the decoupling from the physical infrastructure, new events may be generated, sometimes highly complex combining information from different metrics of different components. Also, the software system has more possibilities into adaptation, including scaling opportunities, network function placement and reconfigurations during runtime for the new network conditions. For this reason, the basic event and logging system which accompanies at this moment the network management stack is not sufficient into a software network environment.

The domain information (e.g. performance indicators, KPIs) is exposed to the slice management, which can use the information to, e.g. replicate critical function to multiple independent domains.

Additionally, information on domain preferences, in particular regarding VNFs, can be made available to the slice management. Moreover, FCAPS operations are supported by PBM functionality of orchestration system.

The life-cycle management plane has multiple points in which and through specific policies, the functionality of the system may be adapted. The PBM can be implemented as Event-Condition-Action (ECA) or as Intent. In the Intent approach, a high-level description of goals is translated by the Intent Engine into low-level operations using an embedded intelligence of that engine. Both cases have similar inputs and outputs, but they differ in the internal structure. Additionally, to this system, an event broker may have to be added to the interconnection between the various analytics engines, the monitoring server and the policy engine. The event broker has the role to properly route the events between the different components.

Based on the monitored information from the slice, the NFVI and the life-cycle management components using the PBM can provide the following adaptations:

- At VIM level: migration of virtual machines, fault management and mitigation at the VM levels, the configuration of the infrastructure, infrastructure security protection, authentication and authorization, resources scheduling for performance and resilience;
- At WIM level: the establishment of new data paths, traffic steering between multiple data paths, QoS classification and differentiation, application differentiation through deep data plane programmability;
- At NFVO level: network functions placement in the domain, scaling policies, automatic fault management, resilience and security through application independent mechanisms, modifying the policies in selecting domain-specific ROs;
- At DSSO level: modifying of policies of NFVOs selecting;
- Multi-domain slice orchestrator – modifying the policies in selecting administrative domains, SLA breaching reports, re-selecting intermediary domains;
- Business Service Slice Orchestrator – transmitting to the tenant events in regard to the system on top of which the slice is deployed (i.e. normal behaviour and exceptional events which may influence the good functioning of the slice).

6.3. Interfaces and reference points of the orchestration architecture

As presented in Figure 13, we have defined multiple interfaces for orchestration system entities, and some of them are compliant with ETSI MANO framework. The ETSI MANO framework does not define protocols that should be used between functional blocks within the orchestration system. Protocols for orchestration system are still an open issue. As 5G!Pagoda, we suggest using a service-based communication architecture. The service-based approach is in line with the design of industrial orchestration solutions such as Open Baton or ONAP.

The service-based architecture uses a common message bus in order to exchange messages between functional entities. Instead of predefined interfaces between elements, a services model is used in which components communicates with each other via a message broker. The most known realizations of this paradigm are Advanced Message Queuing Protocol (AMQP), Message Queue Telemetry Transport (MQTT) or Apache Kafka. The basic architecture of mentioned approaches is composed of a set of publishers, set of subscribers and message brokers (message buses) that organizes messages into queues, called 'topics'. Every published message from any publisher is eligible for delivery into any client session, where a subscription to a specific 'topic' exists. The service-based approach allows distributing functional blocks and providing scalable, reliable and lightweight communication standard between them.

The major difference between service-based and point-to-point approach is that in the former model services queries a service repository function, which implements service discovery functionality. By dint of service discovery, entity services may detect other communication peers, what makes fixed reference points avoidable. This approach is closer to cloud-native networking concept, in which libraries of functions may be requested from a network micro-service catalogue and instantiated on demand.

However, a service-based architecture may be translated into traditional reference point architecture. Thus, we describe below the functional requirements for each of the interfaces that exist within the orchestration system. These functionalities should also be applied within service-based orchestration architecture.

6.3.1.1 Interface BSSO–MDSO

The interface between BSSO and MDSO is significant for 3rd parties and/or verticals. It acts as an API that allows slice tenants to provision and manages network slice instances. The key functionality of the BSSO–MDSO interface is the translation of business requirements and SLAs into technical requirements, such as network latency or resource availability. MDSO should construct TOSCA slice blueprint based on tenant's order.

6.3.1.2 Interface MDSO–DSSO

The interface between the MDSO and the DSSO has a critical role in creating and managing slices across multiple administrative domains. The interface is based on the business relationship and/or the roaming agreements between the domains of the two layers. The DSSO exposes an API that is utilized by the MDSO This API covers functions for:

- Managing blueprints, including querying, uploading and deleting blueprints.
- Creating or upgrading a slice based on an existing or included blueprint.
- Defining policies for a slice. These policies determine the rules for resource allocation, scaling and life-cycle operations on a slice.
- Overriding policies in terms of resource allocation, scaling and life-cycle operations.
- Removing an existing slice, either immediately or when the last UE has left the slice.
- Defining rules for UE assignments to a slice.

- Collecting statistical and usage information from a slice.
- Receiving notifications on actions triggered by policies as well as error and warning messages.

An alternative approach is reactive. In this approach, the MDSO may offer the available blueprints and slice descriptors to the DSSO. These slice descriptors are advertised to clients. When a slice is instantiated, e.g. one or more clients connect to it, the DSSO indicates to the appropriate MDSO, which creates the slice or complements the slice pre-created by the DSSO.

The major format utilized in the API calls is TOSCA for blueprints. TOSCA allows specifying the service as a recursive construct of nodes and relationship between nodes, where each node can recursively consist of a set of nodes and their relationships. Nodes are abstract entities which may denote applications, VNFs, VMs and interfaces. TOSCA can be extended with new node types. Each node and relationship can have life-cycle operations defined, i.e. the actions to be performed on instantiating, starting and stopping nodes. At the highest layer, the slice is defined.

6.3.1.3 Interface SOS–DSSO

In order to perform slice-related operations such as FCAPS management or slice stitching, DSSO has an interface to Slice Operations Support subsystem. This interface is used to provision a sub-slice (Sub-network Slice Instance according to NGMN terminology) and to configure slice stitching/exchange points in the data and control plane of network slice. The DSSO should manage border gateways of sub-slice to configure (if needed) translation between slice domains (e.g. to perform IP-ICN translation). Moreover, DSSO should be able to enforce FCAPS operations of network slice functions.

6.3.1.4 ETSI-compliant interfaces

The 5G!Pagoda high-level orchestration architecture is compliant with ETSI MANO framework. We have adopted the following interfaces from ETSI MANO specification:

- DSSO–NFVO/SDN-O – reference point equivalent to interface between OSS/BSS and NFVO in ETSI MANO framework. Our architecture assumes split into NFV-O and SDN-O, which orchestrates WAN interconnections between NFVI-PoPs. DSSO-NFVO interface is responsible for Network Slice Instance and VNF life-cycle management including instantiation, modification, information query, scaling and termination. Moreover, it performs a VNF package/Network Slice Blueprints management and policy management and enforcement.
- VNFM–EM – interface responsible for FCAPS operations during life-cycle of VNFs.
- VNFM–VNF – interface responsible for life-cycle management and heart-beating of particular VNFs.
- VIM/WIM–NFVI – interface responsible for resource orchestration at the virtual infrastructure level. It allocates virtual resources with an indication of computing/storage, performs life-cycle management of virtual resource runtime environment (e.g. Virtual Machine) and provides network interconnection between them.
- NFVO–VIM/WIM – interface responsible for NFVI resource reservation, release, update as well as VNF software image addition, deletion and modification.

- NFVO–VNFM – this interface is used to coordinate operations that are performed by NFVO and VIM. Before VNF is instantiated this interface is used to authorize, validate and allocate NFVI resources for VNFs, then NFVO performs life-cycle operations via NFVO-VNFM.
- VNFM–VIM – interface responsible for querying NFVI resource information and allocating/releasing NFVI resources by VNFM.

6.3.1.5 Interface to Slice Management

The network slice tenants (3rd parties or verticals) should be able to manage a network slice instance that they own. To make it possible, our orchestration architecture exposes an interface from In-Slice Management (ISM) to slice tenant This interface allows slice tenants to perform FCAPS operations according to their own policies by interacting with EM entities that are associated with VNFs. However, Slice Management may be realized as a special type of VNF Manager that is managed by slice tenant. In this approach, generic VNF Manager is responsible for life-cycle management of VNFs, but VNF Manager owned by tenant performs FCAPS operations. This model assumes that interface to Slice Management is realized as an interface to VNF Manager.

6.4. Slice descriptors used in multi-domain slicing

As it has been already discussed, there is not yet provided standardized slice identification. In the 5G!Pagoda architecture we proposed such identification looking into multi-domain operations and slice selection issues. In the approach proposed by 5G!Pagoda, each network slice is identified by **Network Slice Identifier (NSID)** and by **Network Slice Identifier for Orchestration** (**NSID-O)**, which allows identifying a particular slice instance within an orchestration system.

The Network Slice Identifier (NSID) is comprised of:

- Operator Identifier (mandatory) – required field that specifies, which operator should handle an end device. In 3GPP terminology, Operator Identifier may be a pair of Mobile Country Code (MCC) and Mobile Network Code (MNC) allocated for serving PLMN as defined by 3GPP.
- Slice Class Type (mandatory) – required field that specifies the usage class of end-user application. Network Slice Instance should be selected by mobile core based on Slice Class Type. If there are multiple Network Slice Instances of the same Slice Class Type, selection policy is up to a network operator.
- Slice Tenant Name (optional) – an optional field that specifies tenant of requested slice instance. This feature is required in a multi-tenant mobile network. If Slice Tenant Name (and Slice Tenant ID) is part of NSID, end-users attach requests should be passed directly to a network functions managed by the tenant.
- Slice Tenant ID (optional) – an optional field that is strictly associated with Slice Tenant Name. Slice Tenant Name and Slice Tenant ID should be provided together to use network slice instance managed by the tenant.
- Slice Service Linkage (optional) – an optional field that may be used as assistance information to Slice Class Type. This parameter points out specific service linkage, e.g. YouTube, NetFlix, Facebook, Twitter, etc.

- Quality Constraints (optional) – set of optional parameters that specify quality-related properties of network slice instance. It is usable in case of NSaaS approach. 5G!Pagoda architecture allows users to requests Network Slice Instances on demand. Then, quality constraints should be passed. Otherwise, the best effort network slice will be allocated for a particular application (which is also the default case when a specified by UE slice cannot be found).
- Additional Parameters (optional) – set of parameters allows 3rd parties, tenants and operators to customize network slice identification scheme, what impacts slice selection procedure and policy.

The Network Slice Identifier for Orchestration (NSID-O) identification format is dependent on implementation, and some hash function may be used to generate unique NSID-O. However, for management and orchestration purposes, some additional information should be associated with NSID-O. The NSID-O set of parameters should include:

- Unique Slice Identifier (e.g. hash code);
- Slice Number;
- Slice Type – technology dependent (4G RAN, EPC, etc.). A special case of Slice_Type is the Composite Slice, i.e. the slice that is composed of more than one sub-slice. In the case of 3GPP it can be described as eMBB, uRLLC or mMTC;
- Slice Tenant;
- Slice Quality Parameters;
- Identifier of BSSO that manages a particular slice instance.

The set of parameters for NSID-O is only a suggestion, and it may be treated as an implementation hint. The key agreement is that network slice instance shall be uniquely identified within the orchestration and management systems.

The proposed schemes are in line with the 3GPP approach.

7. Key implementation issues

In this chapter, a series of key implementation issues are considered and further detailed as part of the high-level architecture. Indeed, the conceptual description of the key implementation elements represent the main approaches taken by the 5G!Pagoda project in order to implement the missing key technologies as well as to enable the proper integration within a comprehensive system of the existing 3rd party developments.

7.1. Slice selection, matching and advertisement

In this section, an assessment of the possible mechanisms for slice selection functionality is presented. Slice selection represents the functionality of the system, which enables the end devices to connect to the appropriate slice while accounting that slices are dynamically deployed and removed from the system.

The available mechanisms for slice selection can be classified as:

- Network-based mechanisms in which the network is in charge of selecting the appropriate slice for the UEs while the devices are unaware which would be the selected slice;
- UE controlled slice selection mechanisms in which the UE makes the decision to which slice to connect;
- UE assisted slice selection mechanisms in which the UE makes an indication on which slice should be selected, and the network will select based on the indication the appropriate slice.

For network-based mechanisms, there are two main high-level functions: proxy and redirect. Proxy slice selection functionality presumes that there is a Slice Selection Function (SSF) located in the network which will select the appropriate slice for processing the service of the subscriber. In this case, part of the functionality should be located in the common slice functionality. For example for the 5G system, as illustrated in Figure 16, if the Access and Mobility Function (AMF) is part of the common slice, the SSF will be a functionality added to the AMF to select the proper Session Management Function (SMF) which in its turn will select the proper data path components. In case the AMF is not part of the common slice, then the SSF will be selecting the proper AMF to which the devices to communicate to which will result into a redirect to the proper AMF through the Network Slice Selection Function (NSSF) of the (R)AN.

A network-based slice selection mechanism must be implemented in order to be able to handle properly legacy devices which are not aware of the multi-slice system and are not able to implement UE controlled or UE assisted mechanisms.

As the current standardization within 3GPP is still in a very early stage, there was no clear decision made if the AMF will be part of the common slice (as having the role to manage the access and the mobility of the devices to the common access network) or if the AMF will be also part of dedicated slices (as having the role to maintain customized access and mobility through the same or through customized (R)AN components). Based on this decision, one or the other of the mechanisms will be selected.

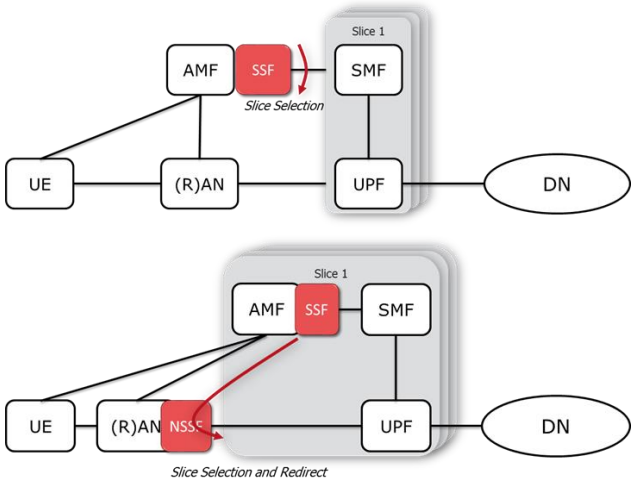

Figure 16 – Network-based slice selection mechanisms

In the UE controlled slice selection model, the UE makes the decisions to which slice to connect. For this, it includes a Slice Discovery Function (SDF), which should receive dynamically the information on the slices available and to which the device is allowed to connect to. The information on the available slices is received dynamically from the Slice Discovery Advertisement Function (SDAF) which is located in the network and has information on the deployment and the termination of the different slices as well as on their specific identifiers and other information which can be transmitted to the UEs in order to make an informed selection decision.

The Slice Discovery Function (SDF) is the end-device function that is responsible for acquiring slice attachment information, especially NSID, which is used to request access to a network slice. The 3GPP in TS 23.501 uses a term of S-NSSAI, which is composed of a list of 'Service/Slice Type – Slice Differentiator' pairs. As 5G!Pagoda we propose to use slice identification format described in section 6.4, because it allows passing additional parameters of the slice like QoS, security level or specific application requirements, e.g. YouTube. Moreover, in 3GPP terminology, Slice Differentiator is loosely defined. Thus we propose more detailed parameters of NSID. However, the Additional Parameters (AP) field allows adding some customizations.

The communication between the SDF and the SDAF can be executed using two different mechanisms:

- Slice advertisement – the SDAF is broadcasting information towards the UEs whenever there is a change in the network slices available in order for the UEs to change their slice decision. This type of operation requires a large number of messages to be transmitted to the UEs and it is necessary only in case the UEs may change their slice while attached to the network (i.e. if a new slice appears in the network, the UEs will hand-over to the slice for better service). Alternatively, the slice advertisement can be used to push updated slice discovery information to the UE to be used for a subsequent attachment to the network in order to be able to speed up the process.
- Slice discovery – the SDF is querying the SDAF on which slice to select at specific moments in time. This can be done synchronously to the attachment procedure, in which case the UE should

connect through a default (or previously selected slice) or can be done at regular time intervals, through which the UE queries during its attachment on which next slice to select.

The device should know all slice instances that are available in its geographic area. In the case of Attach-based acquirement of NSID, SDF function is responsible for obtaining from the network a list of NSAIDs. The list should contain only network slice instances that end device is allowed to use. In this phase, some procedure of Access Control should be performed using UE subscription information. The end device may request access to a specific network slice instance identified by one of the NSAIDs in the list.

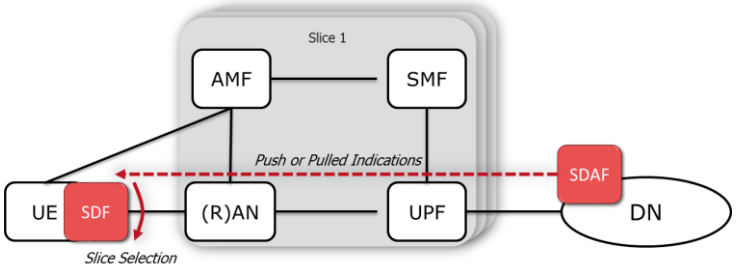

Figure 17 – UE controlled Slice Selection Functionality

From the perspective of the functionality placement, SDAF may be part of the attachment process, or it may be only a recommendation function which transmits information over the established communication paths.

In the first case, the SDAF entity should be the first point of contact for every UE/device that wants to connect to a mobile network. The UE attach requests should be load-balanced and routed to SDAF based on RAN policy. The SDAF and UE should perform basic authentication and authorization as well as redirect the end device to the appropriate network slice. The address of network slice gateway, e.g. IP address, should be delivered. Then the device is served by network slice mechanisms. For instance, session management is performed within a network slice connection. As the authentication and the authorization are performed in the AMF for the attachment to the RAN, this type of functionality is considered redundant mainly because there is no need for double authentication to the RAN.

In the second case, the SDAF is similar to the ANDSF (Access Network Discovery and Selection Function of the 3GPP 4G system) in which after the device attaches to the network to either a default slice or to a previously known slice; new slice information is transmitted to the devices. In this case, the communication can be established over the data path and the information can be transmitted either in the push (SDAF broadcasting) or pull (SDF queries) mode. The pull mode is preferred as long as the devices do not have to take an immediate decision to disconnect and to connect to a new slice in case a new slice appears. Especially, in this case, a grouping with the ANSDF would be beneficial in case the (R)AN also has to be selected as there may be a slice separation starting from the physical environment.

The same process of slice selection may be used also for non-3GPP devices such as IoT LoRa devices. However, it is likely that IoT nodes will have a predefined slice type, which will not change during the lifetime of the device. For the sake of energy saving an IoT, the node does not exchange signalling information with a network and may have the NSID pre-configured in local memory. In

this case, the IoT node sends a request for a particular network slice instance directly, so the static way of slice selection is used.

As the UE selection may be misused to gain access and to execute different types of Denial-of-Service attacks, the UE selection should be complimented in the network by a proper access control in which only the selected devices will be allowed to connect to the specific slices and by the network slice selection means in order to properly select in the network the slice for the legacy devices and for devices for which the discovery information is not anymore actual. These mechanisms are generically named UE assisted slice selection. The UE assisted slice selection is practically a combination of the UE selected mechanisms with a network backup in case the UE slice selection is not properly made.

From the perspective of this specification, the UE assisted mechanisms in presenting the highest probability to get standardized by 3GPP as a similar decision was taken in a similar case of access network discovery and selection. With this consideration, depending on the immediate next steps in the 3GPP standardization, an implementation decision will be taken by 5G!Pagoda to select and to foster the development and the acceptance of the proper slice selection mechanisms for the 5G system.

7.2. Inclusion of hardware nodes and subsystems

The mobile network operators currently operate hardware-based and non-NFV enabled network equipment. The slicing concept is mostly based on software components. There is, however, an important need to include the legacy solutions (especially 4G and data transport networks) into the network slicing ecosystem and provide a smooth migration of NFs from hardware-based solutions to software-based ones. There is no doubt that during the transition phase, both solutions will have to coexist. Therefore, the slicing environment has to be aware of HW-based entourage, and its MANO layer should be able to interact with hardware solutions. The MANO is already incorporating hardware nodes as PNF (Physical Node Functions), but in the context of slicing, we are going beyond a single PNF and include in the overall picture also Hardware Networking Solutions (HNS) as whole subnetworks, i.e. complete solutions based on PNFs only.

The HNS systems or PNF nodes are part of the 5G!Pagoda reference architecture. They are typically used as a part of the Common Slice. However, some of them may already exhibit some features that will make them 'sliceable', for example, they can be seen as virtual routers or devices for VPN creation that in turn that can be used by different slices. It is also worth to mention that some new networking concepts like SDN use software-based components in the control plane. However, the data plane is still hardware-based that is not programmable per se (OpenFlow switches are typically made of hardware).

The generic concept of integration of VNFs and HNSes/PNFs under the same MANO is proposed in 3GPP TS 28.500 [40] and shown in Figure 9. The interaction of MANO with HNSes/PNFs is performed via EM or NM/OSS functions (Ve-Vnfm-em and Os-Ma-convo reference point, respectively). In the case of HNSes/PNFs without dedicated EMs or NMs/OSSes, they can be controlled via embedded management agents. As the MANO-NM/OSS and MANO-EM interfaces contain FCAPS functions from the functional point of view (see: [56], [57]), the westbound interfaces

of EMs and OSSes/NMs (or management agents inside NEs) should provide mediation of EM/NM/OSS native interface to MANO-compliant interface, especially:

- Translation of EM/NM/OSS internal data model of HNS/PNF exposed services to the required level of abstraction of the MANO data model.
- Supporting of NF lifecycle operations (except those exclusively specific for VNFs, as PNF cannot be totally decommissioned, otherwise it will be lost for management), especially full needed PNF (re)parametrization according to service (re)design by MANO.
- Supporting of MANO Configuration/Security Management needs – the orchestrator's repositories have to have the accurate vision of services installed in PNFs and PNFs capabilities/utilization; this requirement of the online awareness of HNSes/PNFs utilization status becomes acute if multiple channels of HNSes/PNFs management (terminated at MANO and non-MANO domains) with overlapping areas of control are allowed.
- Supporting of Fault/Performance/Accounting Management operations according to the HNSes/PNFs specificity and their services model, as it is recognized by MANO (F/P/A-related data should follow the hierarchical model of managed objects).

From the perspective of the 5G!Pagoda, the physical nodes are seen as virtual nodes with very limited capabilities, i.e. they can maintain only a single static functionality of a single type which has to be shared between the multiple slices. With this, the physical boxes may be remotely managed and integrated into the different slices representing either a common part or can be stitched as a common slice within the E2E system.
Apart from legacy systems compatibility, network slicing platform can make use of other hardware-based networking solutions in order to increase its performance. The ETSI NFV has specified 'NFV Hardware Acceleration Technologies' concept [58]. This approach assumes that VNF is not fully-virtualized, but some of the VNF's functionalities are realized by hardware. The acceleration model enables VNFs to leverage acceleration services from the underlying infrastructure, regardless of their implementation. Hardware-based accelerators may increase performance by performing such operations as network stack acceleration, network payload acceleration, cryptography, transcoding, storage, digital signal processing (DSP), or algorithmic acceleration. There are various technologies supporting NFV Acceleration. For instance, DPDK or OpenDataPlane (ODP) may be used to provide a set of libraries and drivers for faster packet processing on the x86 architecture. The NFV Hardware Acceleration technologies will be evaluated during the forthcoming 5G!Pagoda project phase, as it is related to the data plane programmability concept described in section 7.4.

7.3. RAN slicing

One of the key technologies into the development of an E2E customized system is the RAN slicing, where the same RAN is offering customized functionality to different groups of subscribers, especially related to the access and the communication over the shared wireless environment. This functionality enables the shaping of the wireless resources according to the service needs as represented into the specific slices.

7.3.1. RAN slicing options

Focusing on RAN slicing, there are different envisioned models to implement it. Depending on the level of resource isolation we aim to achieve, these models are: dedicated resources and shared resources models.

In the dedicated resource model (i.e. most of RAN functions implemented as Dedicated Slice), the RAN slice is built by separating and isolating slices in terms of control and user plane traffic, MAC scheduler and physical resources. Each slice has access to its own communication protocols (such as RRC, RLC, PDCP and MAC in case of LTE) and the physical resources are strictly dedicated to a specific slice, e.g. a percentage of Physical Resource Blocks (PRB)s is dedicated to each slice, or a subset of the channel is dedicated to each slice. Although dedicated resource model ensures committing elementary resources to the slice, it reduces the slice elasticity as well as scalability and limits the multiplexing gain. Indeed, using the dedicated resource model does not allow a slice owner to easily modify the amount of resource (i.e. PRB) committed to a slice during its life-cycle. Furthermore, the dedicated resources model may lead to a waste of resources, as the PRBs are strictly dedicated to a slice, even if they are not used.

The second approach, i.e. shared resources model allows the slice to share the same: control plane, MAC scheduler and physical resources – it means case the Common Slice functionality is rich. However, we still add the dedicated slice for some advanced Control Plane or Data Plane operations. In this solution, the PRBs are managed by a common scheduler that distributes the PRB to Slices' users according to different criteria, like Service Level Agreement (SLA), priority, etc. Whilst this solution exploits statistical scheduling of physical resources, which ensures more scalability and elasticity by the report to the dedicated resources model, it may lack the support of strict QoS guarantee for Slices and traffic isolation.

Regarding the literature, many works have addressed the challenges of RAN sharing from two main perspectives: (i) resource sharing among Mobile Virtual Network Operators (MVNO), by modifying the MAC scheduler; and (ii) radio resources isolation. For resources sharing, authors in [59] introduce Network Virtualization Substrate (NVS), which operates on top of the MAC scheduler. Its objective is to flexibly allocate shared resources modifying the MAC scheduler to reflect MVNO's traffic need and SLA. NVS was adapted to the case of RAN sharing in LTE [60], with the aim to virtualize the RAN resources. Arguing that most of the MAC schedulers for RAN sharing are less flexible and consider only SLA-based resource sharing, the authors propose AppRAN; an application-oriented RAN sharing solution. The aim is to adapt the RAN sharing mechanism to the applications' need in term of QoS. Looking to radio resource isolation, RadioVisor [62] represents one of the major works that addressed this issue. RadioVisor aims to share RAN resources, which are represented in a three-dimensional grid (radio element index, time slots and frequency slots). The radio resources (in the grid) are sliced by RadioVisor to enable resources sharing for different controllers, which provide wireless access to applications. Each controller is allowed to independently use the allocated radio resources without interfering with the other controllers. It is worth noting that these works mainly focused on RAN sharing issues without considering flexibility and dynamicity as required to enable Network Slicing.

7.4. Data plane issues

7.4.1. Data plane deep programmability

In order for slices to deal with various types of QoS requirements (i.e. xMBB, uRLLC, mMTC), demanding network softwarization should be organized through clear Control and Data Plane separation, flexible programmability in all infrastructure functionalities (e.g. application, control plane, data plane, management and orchestration). Among conventional R&D efforts, OpenFlow and SDN mainly cover the Control Plane programmability through defining common API to control network nodes, NFV and MANO mainly cover the application, management and orchestration programmability introducing open source management and control software which supports standardized API defined by ETSI, for example. Only a few efforts on enhancement of data plane programmability. Although OpenDataPlane and OpenFlow HAL (Hardware Abstraction Layers) have tried to extend the data plane programmability, these limit the flexibility only defining extended API for hardware-based functional blocks. The P4 language is gaining popularity as a tool that can be used to program the behaviour of SDN switches.

The FLARE [63] platform is another solution that allows data plane programmability. It provides reasonable and predictable performance (QoS, i.e. delay, throughput, jitter, reliability) and isolation among multiple data plane slices. In order to provide both high performance and programmable capability, FLARE utilizes general purpose processors (i.e. Intel x86 multi-core processor) and many-core network processors (e.g. TILERA). In such environment, C-plane software modules are used to be executed on Intel CPUs, similarly to current data center technologies and D-plane Software modules are used to be executed on network processors with various virtualization techniques (e.g. Linux hypervisor, Docker container, processor core assignment) similar to the current added value functionality development mechanisms for physical and software switches. Thanks to the combination of various virtualization techniques, each slice is completely isolated from other slices and enables to configure and execute various types of virtual network functions as specified.

The basic concept of FLARE Data Plane architecture is illustrated in Figure 18. As described in [65], FLARE is a programmable node architecture that can concurrently run multiple isolated slices on a physical network node. In Figure 18, three slices are illustrated. Each of the slices is configured and executed in a physical network node, and each slice consists of a control plane (C-plane) modules, data plane (D-plane) modules and their virtual (logical) interfaces. A Node Manager is responsible for the management of slice resource partitioning and utilization and for the dynamic operations for all software modules in slices. The Slicer Controller and Slicer are responsible for the classification of all incoming packets and distribute them into an appropriate slice. The classification parameters are defined and indicated by FLARE central, outside integrated controller, through the Slicer Controller.

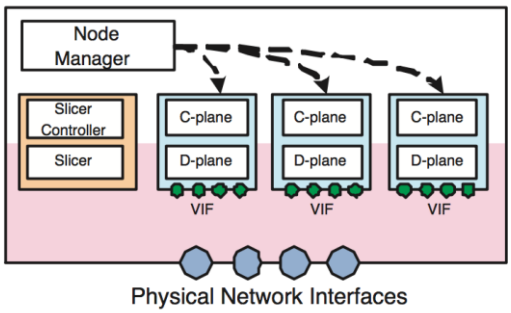

Figure 18 – Data Plane architecture of FLARE [65]

Concerning the data plane performance, network processors contribute important roles for achieving both higher programmability and higher performance. In the current implementation for examples, power efficient and low frequency 36 core processors are available and are very useful for massively parallel processing for a large number of flows with advanced/extended capability (e.g. security, reliability, QoS, etc.) in an isolated manner.

As illustrated in Figure 19, the data plane programmability presumes to bring functionality, which is specific to the different slices in the data path to the common forwarding entities. Thus, performance optimization is obvious as the data packets do not have to switch context between the common forwarding entity and the slice data plane components, being replaced with direct processing in the common forwarding entities, with the specific functionality of the different slices. In this case, the performance optimization highly depends on the virtualization mechanisms as well as on the capacity of the programmable switches to efficiently add new data plane functionality.

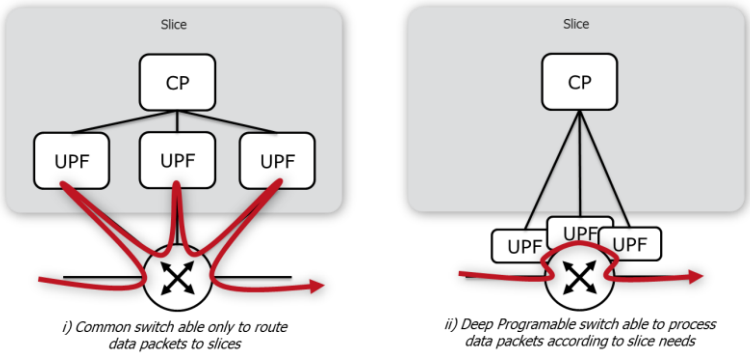

Figure 19 – Data Plane Programmability i) without and ii) with programmable switch

In FLARE, a toy-block networking programming model has been introduced. This allows FLARE users to develop their application and service logic in drag and drop programming manner. The developed software modules are assigned appropriate container (e.g. Docker) and processor cores, then executed in an isolated manner directly on the programmable switch, enabling the implementation of dedicated, customized User Plane Functions (UPFs), while being controlled by external CP functions.

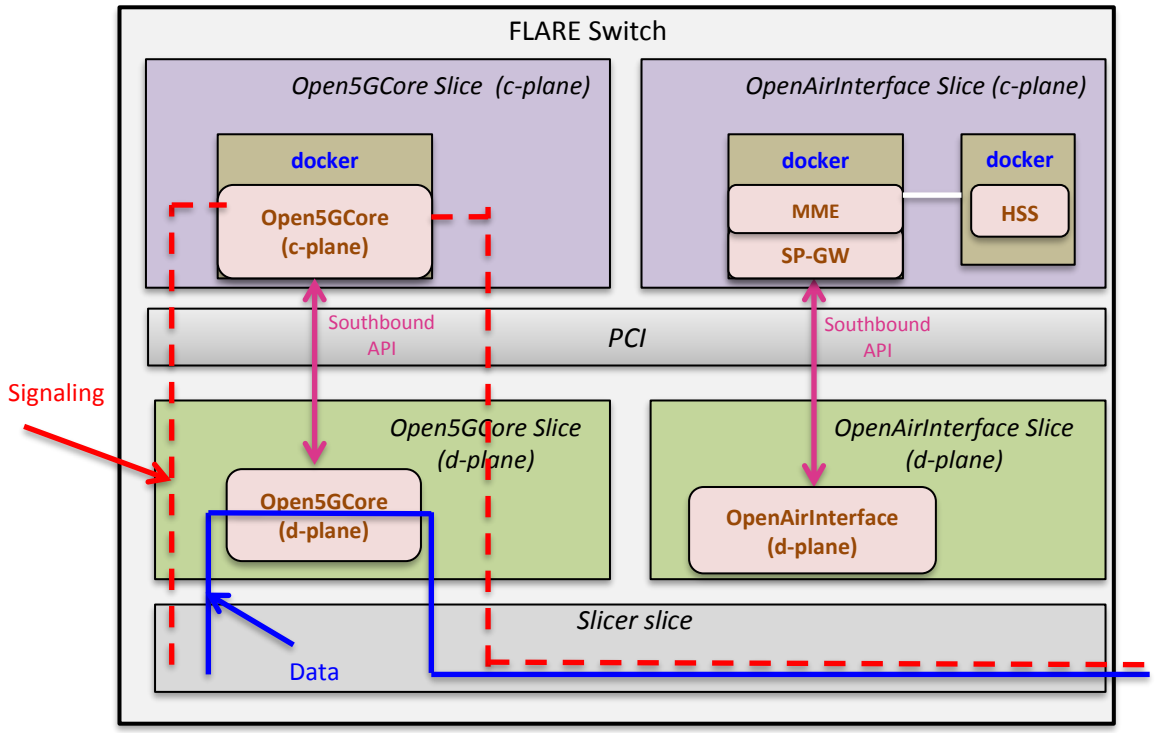

Figure 20 – Data Plane Programmability i) without and ii) with programmable switch

An example of usage of the FLARE concept in 5G!Pagoda testbed is shown in Fig. 20.

Possible and effective use-cases of data plane programmability are application specific traffic controls, M2M Smart gateways, customized OpenFlow Switch Actions and ICN functions, for instances described in [64]. Furthermore, we note that FLARE enables to improve security and reliability of all networking capabilities through integrating with in-network security treatment and in-network big data analysis.

7.4.2. Segment routing

One of the troublesome points of 4G and earlier 3GPP mobile network concepts is the user plane transport that is based on GTP. The GTP tunnel for the specific user's traffic is stretched between two termination points located, e.g. for LTE case, at the eNB and PDN-GW. That way, the user's mobility in the data plane is handled. However, the disadvantage of this technique is the fact that the transmission of a specific user is encapsulated into this UDP pipe and cannot leave it anywhere but the SGi interface, even if it is the traffic flow between two UEs located in the same room. Typically an LTE network contains several SP-GWs per country, locating a nearest user's IP traffic loopback at the SGi interface implies unreasonable delay related to the physical location of UEs, wasteful transmission in terms of traffic path and hereby its aggregation, which causes an increase of demand for transmission hardware and media. Moreover, this makes problematic the usage of the MEC concept.

The 3GPP Release 15 definition maintains the use of GTP in 5G user plane transport layer. Hence, such concepts as MEC or advanced and selective processing of user plane traffic become harder for implementation. Different workarounds are possible (multiple packet data bearers for different service applications, multiple GTP segments within the UP for adding traffic forking points, etc.),

however they have some limitations (e.g. maximum allowable/reasonable number of bearers per UE), imply excessive use of resources (e.g. effort on management of multiplied bearers quantity) or introduce additional UP delay (e.g. segmentation of GTP tunnels means additional termination points and protocol stack processing).

One of the proposed ways of removal of GTP from the user plane is IPv6 Segment Routing (SRv6) [66]. SRv6 is a source routing technique developed by the IETF, in which the packets carry also the information about their forwarding path inside a network using the IPv6 Extension Header mechanism. That way the network operator can actively control the traffic path followed by packets on per-source (per-UE) basis for assurance of some assumed QoS, e.g. end-to-end latency. Moreover, SRv6 is also being considered as a technique for transport network slicing, as it allows for the selection of a particular link and queue for each slice. In case of application of SRv6 within the UP transport layer, the transport control plane and orchestration/management plane have to be extended to support coordination of SR decisions with slice allocations in the mobile network over the underlying transport layer.

The 5G!Pagoda architectural network slicing framework has not analysed the SRv6 usage for network slicing, 3GPP works are at the preliminary stage at the moment of writing of this deliverable and concern only SRV6 based generic transport without using this technology for network slicing. However, 5G!Pagoda framework is transparent to transport network technologies, and the abovementioned extension can be considered as a functional upgrade of 5G!Pagoda framework management layer with the ability to interact with decision engines of user plane forwarding functions.

7.4.3. Flex-E

The Flex-Ethernet (Flex-E) technology has been introduced in order to cope with high bitrate datalink and for the need of QoS provisioning. The Flex-E lies on the decoupling of the PHY layer data rate from the client-MAC data rate. In Flex-E the time-division multiplexing is used in order to aggregate the MAC flows to the single PHY data stream. The time slot mechanism is driven by the Calendar, that has coarse granularity (20 slots per 100G PHY allowing for 5 Gb/s granularity. The Flex-E operations require a control plane that so far has been not yet defined. Among control plane candidates there is the GMPLS approach (RSVP-TE based) and SDN. The control plane can be enhanced by mechanisms that support the network slicing. Such conceptual approach has been presented in [67]. There is, however, no doubt that in order to support network slicing with relatively low granularity, the Flex-E concept has to be modified yet.

8. Network slicing performance metrics

One of the most important issues of network slicing is to provide proper performance of operations. Unfortunately for network slicing the high-level performance metrics (commonly called Key Performance Indicators (KPIs) have not been defined yet by any standardization body. The KPIs should be:

- limited – too many KPIs causes infobesity and moves the effort from dealing with real problems to management of management-related data;
- abstracted – to provide easy and clear comparability of specific solutions of e.g. different vendors at some architectural level of view;
- meaningful – to provide synthetic view of the system's health/behavior and clearly indicate problems which need a reaction or provide a comparative benchmark.

Telco operators define and use KPIs for:

- requirements for system definition and implementation;
- verification of functioning of installed networks, sub-networks, systems or sub-systems;
- definition of customer's requirements and provider's obligations as a part of the Service-Level Agreement (SLA) contract;
- comparison of performance provided by specific vendors or technologies.

For a system composed of multiple subsystems, keeping conformance of the subsystems' to their KPIs can provide the fulfilment of the overall system's KPIs and assuring the service quality.

From the technology and management point of view, a network slicing platform forms another layer, which should be taken under measurements focused on its performance and behaviour. Following the abovementioned assurance and delivery processes aspects, the network slicing KPIs should be divided into:

- **real-time KPIs** related to efficiency of dynamic autonomous processes of resources scaling – as network softwarization solutions use virtual computation (vCPUs), virtual memory (RAM, swap and storage) and virtual connectivity (virtual links between network functions) resources, monitoring of use of allocated resources (looking for their possible saturation or significant underspending, especially persistent) provides indication of overloading of orchestrator part responsible for resources allocation (VNFM, VIM) or non-optimal dynamic resource allocation algorithm; in both cases the higher layer of management will react by necessary slice reconfiguration request;
- **life-cycle KPIs** related to agility of platform's orchestrator response to requests of delivery/reconfiguration/termination of a network slice – this agility will be rather specific to slice template (blueprint), its geographical span, complexity etc.; however, having a continuously evaluated awareness of slicing system's responsiveness to delivery/reconfiguration/termination requests coming from the higher layers of

management to the network slicing platform API, the operator will be able to plan, synchronize and optimize the delivery processes.

In case of real-time KPIs, the threshold-based approach to monitoring should be applied, i.e. only threshold-crossing notifications of excessive/insufficient utilization of resources should be collected and processed to avoid wasteful transmission of the huge amount of management data. In the case of life-cycle KPIs, as the request/response mechanism at the network slicing platform's orchestrator interface will be used, the higher-layer management system will be able to measure and evaluate these KPIs directly.

The differences between network slice templates in terms of complexity, number of virtual components, geographic span, as well as a huge difference in the infrastructure that used for template deployment (single or semi-centralized cluster of data centres versus highly distributed datacenters) makes hard if not impossible. There is no doubt, however, that the 5G!Pagoda mechanisms contribute to scalability related to run-time as well as life-cycle management.

9. Implementations of the 5G!Pagoda architecture

9.1. IoT testbed

The testbed for the IoT use case follows the key concepts of the 5G!Pagoda architecture. It uses virtual infrastructure. As it does not have a dependency on any legacy components, only implements VNFs and does not contain PNFs. The testbed is based on a Common Slice as well as Dedicated Slices for each of the slices.

The common slice primarily covers the Control Plane and Management Plane. It includes the Slice Selector (in the form of an SDN controller), the Slice Selection Function (SFF), the overlay network management functions, and the access network switches. It contains databases of users, slices and access control rights that are shared between slices. The network of the common slice covers the basic networking needs allowing the setup of the Dedicated Slices.

The Dedicated Slices of the two video slices covers the Application Plane, Control Plane and Data Plane. The Data Plane in these slices includes slice specific virtual switches and Network Address Translators (NATs). The Control Plane of the video slices covers the functions for bandwidth allocation and address assignment (Dynamic Host Configuration Protocol servers). The application plane includes the Video Servers for processing video streams. In-slice management covers the functionality required for registering the slice resources, connecting to overlay networks and communicating with the orchestrator, e.g. for retrieving configuration data.

The orchestration of the IoT testbed covers the orchestration blocks of the 5G!Pagoda architecture in the following way. The NFVI, VIM/WIM, NFVO, and VNFM are implemented as specified in the architecture. The DSSO is in this implementation, only a thin layer included as part of the NFVO software (OpenBaton). The Multi-Domain Slice Orchestrator (MDSO) has been designed specifically for the testbed. Implemented functionalities include the interfaces toward multiple DSSOs, triggering of slice installation and removal, management of slice state, and management of slice configuration. Key features of the Business Service Slice Orchestrator (BSSO) have been implemented as part of the MDSO. These include the frontend toward the slice owner, authentication and admission control, as well as the APIs for blueprint management and slice deployment/removal. The orchestration platform is able to deploy slices end-to-end across multiple technology domains (access network and cloud infrastructure). It is also able to deploy slices across multiple administrative domains by interfacing multiple DSSOs and connecting the Control and Data Plane networks using overlays.

9.2. ICN/CDN testbed

ICN/CDN combined content delivery service utilizes the 5G!Pagoda architectural components as listed below.

ICN slice is a dedicated slice which uses the emerging network architecture ICN, especially CCN/NDN, in the data plane. ICN specific components are prepared in the form to be operational on the FLARE node, which is the special node hardware providing with the deep data plane programmability, and forms the 5G!Pagoda platform at Japan side. The management plane is connected to the Japan domestic DSSO (Hitachi orchestrator) and the UT's FLARE manager. As a slice operation support, ICN coordinator is prepared to manage the FIB (Forward Information Base) of ICN node. The data plane communication and management plane communication are isolated to each other.

CDN slice is also a Dedicated Slice. The multi-domain slice creation feature is utilized. As for the data plane, the Internet is used. As for the CDN slice specific components, such as content cache, media converter and streamer are developed as application plane components. CDN coordinator component is also developed which can be classified as management plane component, to manage the in-slice components, accepting the user content request, and maybe handle the business issue such as charging.

As to realize the end to end service provisioning, ICN slice and CDN slice must be stitched (inter-slice stitching). For this purpose, multi orchestrator communication is implemented and also the ICN gateway component is developed.

9.3. ISM testbed

This In-Slice Management (ISM) concept has been implemented in a satellite testbed (not integrated with the main testbeds) in Warsaw. It lies on the implementation of certain management functions as a part of the slice template. The concept provides multiple benefits over the OSS/BSS only based management; it improves network slicing management scalability, provides natively a management interface to slice tenant and separates management spaces. The present implementation uses virtualized OAI platform in which multiple slices share common RAN (using FlexRAN approach) and multiple EPCs. The key component of ISM is Slice Manager (SM) that performs a master role in interaction with all EMs. The SM provides also interface to slice tenant. Via SM, it is possible to monitor certain parameters of a slice and change its configuration. In order to trigger the creation of a new slice, the tenant interacts with the Business Orchestrator (BO) via BO web page. After successful slice creation, the slice tenant obtains (via a web page) access to Slice Manager. Using this interface, it can monitor the behaviour of slice functions (implemented as VNFs) or configure them.

10.Concluding remarks

This deliverable describes the final 5G!Pagoda architecture, which key feature is the scalable support for network slicing. The architecture is a top-down approach based on the intensive state of the art analysis of network slicing concepts that include 5G PPP projects, 3GPP 5G slicing works, ITU-T IMT-2020 Focus Group concepts, IETF activities related to slicing and NGMN vision of slicing. As has been shown, there are already many approaches to network slicing. Some of them address specific types of slices (3GPP, 5GEx) or specific aspects of slicing only. The analysis led to the conclusion that the research community is still looking for a universal approach to network slicing that is able to fulfil described in this deliverable requirements. The 5G!Pagoda consortium has decided not to design 'yet another network slicing architecture', but rather to make the first step towards the convergent approach that will incorporate the existing approaches, including the 3GPP ones. It led to the definition of the 5G!Pagoda reference architecture that can be instantiated in many different ways depending on specific needs. The main features of the proposed approach are the following:

- It allows also for integration of the legacy mobile systems, an issue that was so far neglected in many slicing concepts. Such property enables the coexistence of hardware and software-based solutions and later on smooth migration towards fully software based ones. It is worth mentioning that 3GPP is working on 5GS interoperability with LTE and such interoperability will be defined in Release 17.
- The slicing can be independently done on each of the network planes. Some of them can be sliced only partially – if there are specific constraints (for example in the case of RAN).
- A slice can be created as a composition of the Common Slice and a Dedicated Slice (a tandem). The approach enables compatibility with some slicing approaches as well as with legacy solutions. The Common Slice can be used when no appropriate Dedicated Slice can be found or until such slice is not created yet (the Slice On-Demand case). It helps in the reduction of the Dedicated Slice footprint and slice deployment time.
- The slice structure includes functional components that support lightweight slice management by slice tenant (ISM), components specific to slice operations (SOS). The distribution of the management contributes to the overall scalability of the concept and resiliency. The scalability of management is even more improved by the proposed automated in-slice management (autonomic ISM).
- In the context of multi-domain slicing, we address multi-slicing operations, management as well as multi-domain-orchestration. To that end, we have also defined the orchestration architecture. The proposed multi-domain orchestration allows for resource aggregation based orchestration or hierarchical orchestration architecture in which domain-level orchestrators are used at the lowest level of orchestration hierarchy. The management and orchestration details are provided in WP4 deliverables.

Finally, we have decided not to define interfaces between the functional components of our architecture. The reason was twofold. First, some of the components are dependent on a specific type of slice (RAN, EPC, transport), and they will be defined during the architecture instantiation. The second reason is related to the use of software components as blocks of our architecture. In the software world, much more typical is the definition of APIs and, recently, the use of message busses, like RabbitMQ or Kafka with the publish-subscribe paradigm for information exchange. The latter case is already in use in such solutions like OpenBaton or ONAP and provides high flexibility in terms of dependencies (flexible hierarchy) of its components.

During the definition of the 5G!Pagoda architecture, we have identified several topics that require more research effort, and we have decided to work on some of them. The selection was based on the background and competences of the 5G!Pagoda consortium members. The list of topics includes programmable data plane operations in order to provide a highly efficient data plane that supports transport slicing; another topic was RAN slicing. The proposed by us approaches (FLARE, RAN slicing) have been successfully implemented in our testbeds (Berlin, Tokyo). We have also implemented the In-Slice Management concept in a small OAI-based, testbed in Warsaw.

The initial version of the proposed architecture has been described in deliverable D2.3 that has been written before the 3GPP has published Release 15 recommendations that describe the first version of the 5G system. The careful reader may find marginal differences between the initial architecture and the final one. We have noticed (on the basis of implementation) that there is no need for the modification of the initial architecture. Despite significant work progress, the main foundations of the proposed approach are still valid. (there are, however, some changes related to the terminology and the clarity of the description). We have noticed that some of the 5G!Pagoda requirements have been already addressed by 3GPP Release 15 (for example priority of slices, multi-slice attachment, providing the tenants with some slice management capabilities). However, some of the proposed by 3GPP implementations differ from the 5G! Pagoda implementation, despite their functionality, is similar. For example, 3GPP has proposed to create the slice tenant management interface using the publish-subscribe mechanisms and the centralized OSS/BSS, whereas 5G!Pagoda has proposed separate management spaces for each slice tenant. We deeply believe that our approach is much more scalable and provide better isolation of slice management operations performed by tenants.

It seems that the mobile network and generic network slicing still require substantial work until it will become a commercial reality, and we believe that 5G!Pagoda approach can be a good foundation for future work.

Appendix A. References

[1] ETSI NFV ISG, 'Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options', ETSI GS NFV-IFA 009 V1.1.1 2016-07

[2] 5G!Pagoda, 'D2.3: Initial report on the overall system architecture definition – ver. 1.0'; source: https://5g-pagoda.aalto.fi/assets/demo/attachement/delivrables/5G!Pagoda%20-%20D2.3%20-%20architecture.pdf

[3] NGMN, 'Description of Network Slicing Concept', 2016-01; source: https://www.ngmn.org/uploads/media/160113_Network_Slicing_v1_0.pdf

[4] 5G PPP Architecture Working Group – View on 5G architecture, version 1.0, 2016-07; source: https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-July-2016.pdf

[5] A. de la Oliva, X. Costa-Pérez, A. Azcorra, A. di Giglio, F. Cavaliere, D. Tiegelbekkers, J. Lessmann, T. Haustein, A. Mourad, P. Iovanna, 'Xhaul: toward an integrated fronthaul/backhaul architecture in 5G networks', in *IEEE Wireless Communications*, vol. 22, no. 5, pp. 32-40, 2015-10

[6] S. González, A. de la Oliva, X. Costa-Pérez, A. Di Giglio, F. Cavaliere, T. Deiss, X. Li, A. Mourad, '5G-Crosshaul: An SDN/NFV Control and Data Plane Architecture for the 5G Integrated Fronthaul/Backhaul', in *Transactions on Emerging Telecommunications Technologies* , vol. 27, no. 9, pp. 1196-1205, 2016-09

[7] X. Costa-Pérez, A. Garcia-Saavedra, X. Li, T. Deiss, A. de la Oliva, A. di Giglio, P. Iovanna, A. Mourad, '5G-Crosshaul: An SDN/NFV Integrated Fronthaul/Backhaul Transport Network Architecture', in *IEEE Wireless Communications*, vol. 24, no. 1, pp. 38-45, 2017-02

[8] 5GEx, 'Initial System Requirements and Architecture'; source: https://h2020-5gex.herokuapp.com/pdf/5GEx_D2.1v1.1_public_version.pdf

[9] 5G NORMA, 'Functional network architecture and security requirements'; source: https://5gnorma.5g-ppp.eu/wp-content/uploads/2016/11/5g_norma_d3-1.pdf

[10] 5G NORMA, 'Network architecture – intermediate report'; source: https://5gnorma.5g-ppp.eu/wp-content/uploads/2017/03/5g_norma_d3-2.pdf

[11] SliceNet, 'D2.2 Overall Architecture and Interfaces Definition'; source: https://bscw.5g-ppp.eu/pub/bscw.cgi/d275903/D2.2-Overall%20Architecture%20and%20Interfaces%20Definition.pdf

[12] 5GTANGO, 'D2.2 Architecture Design'; source: https://www.5gtango.eu/documents/D22_v1.pdf

[13] MATILDA, 'D1.1 - MATILDA Framework and Reference Architecture'; source: https://private.matilda-5g.eu/documents/PublicDownload/119

[14] ITU-T, 'Requirements of soft network architecture for mobile', Recommendation ITU-T Y.3323, 2016-09

[15] ITU-T, 'High-level technical characteristics of network softwarization for IMT-2020', Recommendation ITU-T Y.3150, 2017-09

[16] ITU-T, 'IMT-2020 network management and orchestration requirements', Recommendation ITU-T Y.3110, 2017-09.

[17] ITU-T, 'IMT-2020 network management and orchestration framework', Recommendation ITU-T Y.3111, 2017-09

[18] ITU-T, 'Framework for the support of multiple network slicing', Recommendation ITU-T Y.3112, 2018-05

[19] ITU-T, 'Business Role-based Models in IMT-2020', Recommendation ITU-T Y.3103, 2018-09

[20] ITU-T, 'Architecture of IMT-2020 network", Draft Recommendation ITU-T Y.3104, 2018-12

[21] ITU-T, 'Requirements of network capability exposure in the IMT-2020 networks', Recommendation ITU-T Y.3105, 2018-12

[22] ITU-T, 'Network capability exposure function in IMT-2020 networks', Draft recommendation ITU-T Y.IMT2020-CEF; source: https://www.itu.int/ITU-T/workprog/wp_item.aspx?isn=14065 (last access: 07-06-2019)

[23] ITU-T, 'Network slicing orchestration and management', Draft recommendation ITU-T Y.NSOM; source: https://www.itu.int/itu-t/workprog/wp_item.aspx?isn=13639 (last access: 07-06-2019)

[24] ITU-T, 'Requirements for network slicing with AI-assisted analysis in IMT-2020 networks', Draft recommendation ITU-T Y.IMT2020-NSAA-reqts; source: https://www.itu.int/ITU-T/workprog/wp_item.aspx?isn=15061 (last access: 07-06-2019)

[25] IETF, 'Network Slicing – Introductory Document and Revised Problem Statement' (draft-gdmb-netslices-intro-and-ps-02), 2017-02; source: https://tools.ietf.org/html/draft-gdmb-netslices-intro-and-ps-02

[26] IETF, 'Network Slicing - Revised Problem Statement" (draft-galis-netslices-revised-problem-statement-01), Internet-Draft, informational, 2017-07; source: https://tools.ietf.org/html/draft-galis-netslices-revised-problem-statement-01

[27] IETF, 'Network Slicing Architecture' (draft-geng-netslices-architecture-02), Internet-Draft, informational, 2017-07; source: https://tools.ietf.org/html/draft-geng-netslices-architecture-02

[28] IETF, 'Network Slicing Management and Orchestration' (draft-flinck-slicing-management-00), Internet-Draft, informational, 2017-07; source: https://tools.ietf.org/html/draft-flinck-slicing-management-00

[29] IETF, 'Building blocks for Slicing in Segment Routing Network' (draft-ali-spring-network-slicing-building-blocks-00.txt), Internet-Draft, informational, 2018-07; source: https://tools.ietf.org/html/draft-ali-spring-network-slicing-building-blocks-00

[30] IETF, 'Autonomic Slice Networking' (draft-galis-anima-autonomic-slice-networking-05), Internet-Draft, informational, 2018-09; source: https://tools.ietf.org/html/draft-galis-anima-autonomic-slice-networking-05

[31] 3GPP Technical Report TR 23.707, 'Architecture enhancements for dedicated core networks', ver. 13.0.0, 2014-12

[32] 3GPP Technical Report TR 23.711, 'Enhancements of dedicated core networks selection mechanism', ver. 14.0.0, 2016-09

[33] 3GPP Technical Report TR 23.799, 'Study on Architecture for Next Generation System', v14.0.0, 2016-12

[34] 3GPP Technical Specification TS 23.501, 'System Architecture for the 5G System', ver. 16.0.2, 2019-04

[35] 3GPP Technical Specification TS 29.531, '5G System; Network Slice Selection Services; Stage 3', ver. 15.3.0, 2019-03

[36] 3GPP Technical Specification TS 38.300, 'NR; NR and NG-RAN Overall Description; Stage 2', ver. 15.4.0, 2019-01

[37] 3GPP Technical Report TR 28.801, 'Study on management and orchestration of network slicing for next-generation network', v15.1.0, 2018-01

[38] 3GPP Technical Specification TS 28.530, 'Management of network slicing in mobile networks; Concepts, use cases and requirements', ver. 15.1.0, 2018-12

[39] 3GPP Technical Specification TS 28.533, 'Management and orchestration; Architecture framework', ver. 15.1.0, 2018-12

[40] 3GPP Technical Specification TS 28.500, 'Telecommunication management; Management concept, architecture and requirements for mobile networks that include virtualized network functions', ver. 15.0.0, 2018-06

[41] ETSI NFV ISG, 'Network Operator Perspectives on NFV priorities for 5G', whitepaper; 2017-02; source: https://portal.etsi.org/NFV/NFV_White_Paper_5G.pdf

[42] ETSI NFV ISG, 'Network Functions Virtualisation (NFV); Architectural Framework', ETSI GS NFV 002 V1.2.1 2014-12

[43] ETSI NFV ISG, 'Network Functions Virtualization (NFV) Release 3; Evolution and Ecosystem; Report on Network Slicing Support with ETSI NFV Architecture Framework', ETSI GR NFV-EVE 012 V3.1.1 2017-12

[44] ETSI NFV ISG, 'Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Functional requirements specification', ETSI GS NFV-IFA 010 V3.2.1 2019-04

[45] ETSI NFV ISG, 'Network Function Virtualisation (NFV) Release 3; Management and Orchestration;Report on architecture options to support multiple administrative domains', ETSI GR NFV-IFA 028 V3.1.1, 2018-01

[46] ETSI NFV ISG, 'Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Multiple Administrative Domain Aspect Interfaces Specification; ETSI GS NFV-IFA 030 V3.2.1 2019-04

[47] I. Afolabi. A. Ksentini, M. Bagaa, T. Taleb, M. Corici, A. Nakao, 'Towards 5G Network Slicing over Multiple-Domains', in *IEICE Transactions on Communications*, vol. 100B, no. 11, pp. 1992-2006, 2017-11

[48] ETSI NFV ISG, 'Network Function Virtualisation (NFV) Release 3; Management and Orchestration; Report on Management and Connectivity for Multi-Site Services', ETSI GR NFV-IFA 022 V3.1.1, 2018-04

[49] ONF TR-526, Applying SDN Architecture to 5G Slicing, Issue 1, 2016-04

[50] ITU-T Y.3011, Framework of Network Virtualization for Future Networks, 2012-01

[51] R. Guerzoni, I. Vaishnavi, D. Perez-Caparros, A. Galis, F. Tusa, P. Monti, A. Sganbelluri, G. Biczók, B. Sonkoly, L. Toka, A. Ramos, J. Melián, O. Dugeon, F. Cugini, B. Martini, P. Iovanna, G. Giuliani, R. Figueiredo, L. Miguel Contreras-Murillo, R. Szabo, 'Analysis of end-to-end Multi-domain Management and Orchestration Frameworks for Software Defined Infrastructures: An Architectural Survey', in *Transactions on Emerging Telecommunications Technologies*, vol. 28, no. 4, 2017-04

[52] 5G NORMA, 'D3.3: 5G NORMA network architecture - Final report', September 2017; source: https://5gnorma.5g-ppp.eu/dissemination/public-deliverables

[53] 5G!Pagoda, 'D2.1 – Use Case Scenarios and Technical System Requirements Definition – ver. 1.1'; source: https://5g-pagoda.aalto.fi/assets/demo/attachement/delivrables/D2.1_Use_case_scenarios_and_technical_system_equirements_definition_1.1.pdf

[54] 3GPP Technical Specification TS 29.522, '5G System; Network Exposure Function Northbound APIs; Stage 3', ver. 15.3.0, 2019-03

[55] IETF, 'SUPA Policy-based Management Framework', 2017-03; source: https://tools.ietf.org/html/draft-ietf-supa-policy-based-management-framework-01

[56] ETSI NFV ISG, 'Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Os-Ma-Nfvo reference point – Interface and Information Model Specification', ETSI GS NFV-IFA 013 V3.2.1, 2019-04

[57] ETSI NFV ISG, 'Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Ve-Vnfm reference point - Interface and Information Model Specification', ETSI GS NFV-IFA 008 V3.2.1, 2019-04

[58] ETSI NFV ISG, 'Network Functions Virtualisation (NFV) Release 2; Acceleration Technologies; VNF Interfaces Specification', ETSI GS NFV-IFA 002 V2.4.1, 2018-02

[59] T. Guo, R. Arnott, 'Active LTE RAN Sharing with Partial Resource Reservation', in Proc. of the 2013 IEEE 78th Vehicular Technology Conference (VTC Fall), Las Vegas, NV, pp. 1-5, 2013-09

[60] X. Costa-Pérez, J. Swetina, T. Guo, R. Mahindra, S. Rangarajan, "Radio access network virtualization for future mobile carrier networks," in *IEEE Communications Magazine*, vol. 51, no. 7, pp. 27-35, 2013-07

[61] J. He, W. Song, 'AppRAN: Application-oriented radio access network sharing in mobile networks', in Proc. of the 2015 IEEE International Conference on Communications (ICC), London, UK, pp. 3788-3794, 2015-06

[62] S. Katti. L. Li, 'RadioVisor: A slicing plane for radio access networks', in Proc. of ACM of the third workshop on Hot topics in software-defined networking (HotSDN), Chicago, IL, USA, pp. 237-238, 2014-08

[63] A. Nakao, 'Software-Defined Data Plane Enhancing SDN and NFV', in *IEICE Transactions on Communications*, vol. E98-B, no. 1, pp. 12-19, 2015-01

[64] A. Nakao, P. Du, T. Iwai, 'Application Specific Slicing for MVNO through Software-Defined Data Plane Enhancing SDN', in *IEICE Transactions on Communications*, vol. E98-B, no. 11, pp. 2111-2120, 2015-10

[65] A. Nakao, P. Du, Y. Kiriha, F. Granelli, A. A. Gebremariam, T. Taleb, M. Bagaa, 'End-to-End Network Slicing for 5G Mobile Networks', in *Journal of Information Processing*, no. 25, no. 1, pp. 153-163, 2017-01

[66] 3GPP Technical Report TR 29.892, 'Study on User Plane Protocol in 5GC', ver. 1.1.0, 2019-04

[67] K. Katsalis, L. Gatzikis, K Samdanis. Towards Slicing for Transport Networks: The Case of Flex-Ethernet in 5G, IEEE Conference on Standards for Communications and Networking, Paris, France 29-31 October 2018.