5G CITY

Grant Agreement No.761508 5GCity/H2020-ICT-2016-2017/H2020-ICT-2016-2

D2.2: 5GCity Architecture & Interfaces Definition

Dissemination Level		
☒	PU:	Public
☐	PP:	Restricted to other programme participants (including the Commission Services)
☐	RE:	Restricted to a group specified by the consortium (including the Commission
☐	CO:	Confidential, only for members of the consortium (including the Commission Services)

Grant Agreement no: **761508**	Project Acronym: **5GCity**	Project title: **5GCity**

Lead Beneficiary: **ITALTEL**	Document version: **V1.1**

WP2 - 5GCity Architecture, Requirements and Use Cases

D2.2: 5GCity Architecture & Interfaces Definition

Start date of the project: 01/06/2017 (duration 30 months)	Contractual delivery date: 31.05.2018	Actual delivery date: Date of submission to Coordinator

Editor name: Viscardo Costa (ITL)

List of Contributors

Participant	Short Name	Contributor
Italtel	ITL	Antonino Albanese, Viscardo Costa
Fundacio I2CAT	I2CAT	Shuaib Siddiqui, August Betzler, Hamzeh Khalili, Apostolos Papageorgiou, Sergi Figuerola
INCITES	INC	Theodoros Rokkas, Ioannis Neokosmidis
Retevision (Cellnex Telecom)	RTV	David Pujals, Luis Moreno
ADLINK	ADLINK	Gabriele Baldoni
Nextworks s.r.l.	NXW	Nicola Ciulli, Paolo Cruschelli, Elian Kraja, Elio Francesconi
WindTre	WindTre	Maria Rita Spada
Ubiwhere	UBI	Pedro Diogo, Ricardo Preto
Virtual Open Systems	VOSYS	Michele Paolino, Daniel Raho
Accelleran	XLRN	Simon Pryor, Antonio Garcia, Trevor Moore
MOG-Technologies	MOG	Alexandre Ulisses, Pedro Santos
IMI – Barcelona City Council	IMI	Mariano Lamarca, Jordi Cirera, Gonzalo Cabezas

List of Reviewers

Participant	Short Name	Contributor
Italtel	ITL	Antonino Albanese
Bristol University	UNIBIO	Carlos Colman
WindTre	WindTre	Maria Rita Spada
NEC	NEC	Felipe Huici

Change History

Version	Date	Partners	Description/Comments
v0.0	06-04-2018	ITL	Table of Content
v0.1	19-04-2018	ADLINK	Inclusion of contributions by ADLINK
v0.2	24-04-2018	WindTre	Inclusion of contributions by WindTre
v0.3	26-04-2018	VOSYS	Inclusion of contributions by VOSYS
v0.4	27-04-2018	NXW	Inclusion of contributions by NextWorks
e00	02-05-2018	ITL	Inclusion of contributions by XLRN, ITL, RTV, inCITES, i2CAT
e11	14-05-2018	ITL	Update contributions by NXW, ITL - Inclusion of contributions by UBI
e15	18-05-2018	ITL	Clean version
e16	24-05-2018	ITL	Version ready for 1^ review

e17	28-05-2018	ITL	Review by WP2 Leader (Antonino Albanese)
v1.0	29-05-2018	NEC	Review by Technical Manager (Felipe Huici)
v1.1	22-11-2018	VOSYS	Added details on scalability and fog05 interoperability in Section 3.3 and Small Cells NFVI support in section 4.6 Clarified approach to interface design and added refences to future deliverable with specifications in section 5.

DISCLAIMER OF WARRANTIES

5GCity has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 761508. The content of this deliverable does not reflect the official opinion of the European Union. Responsibility for the information and views expressed in the deliverable lies entirely with the author(s).

Table of Contents

Figures

Tables

Executive Summary

Based on the Use Cases and requirements outlined in deliverable D2.1 [1], the purpose of this document is to provide the 5GCity overall system architecture and specifications as the main outcomes of task T2.2 and T2.3, which have the following objectives:

- to design the high level architecture, guaranteeing the different vertical Use Cases deployment in the same architectural framework, considering the 5GCity three-tier domains: Data Centers, MEC nodes and small cells/Wi-Fi APs at the extended edge
- to provide the functional description of all the subsystems of 5GCity architecture
- to define system interfaces and workflows considering virtualization and provisioning aspects.

This deliverable will also include a first analysis of the novel business models and techno-economics (neutral host) enabled between the cities and the telecom providers through the 5GCity architecture.

The overall contributions of D2.2 can be summarized as follows:

- General architecture and new business model scenarios (neutral host)
- Service layer functional description
- Orchestration & Control Layer functional description
- Infrastructure Layer functional description
- Interface and workflows specification

This deliverable provides the basis for the design and implementation work that will be performed in the work packages 3, 4 and 5. As it is expected that the architecture can change over time based on findings throughout the project, it illustrates the first version of the 5GCity architecture, as a result of the first iteration. The Gantt of the project envisages other two iteration of the architecture:

1. **Deliverable D2.3**: 5GCity Architecture and Interfaces Update, on M19;
2. **Deliverable D2.4**: 5GCity Final Architecture and Interfaces, on M25.

1. Overview

1.1. 5GCity Architectural principles: an overall look

Figure 1 depicts the 5GCity functional architecture providing a view of the different functional blocks necessary to adapt city infrastructure to the cloud and virtualized edge network. This architecture is composed by three layers:

- Service/Application Layer;
- Orchestration & Control layer;
- Infrastructure layer.

The architecture will be detailed in the following paragraphs of the deliverable.

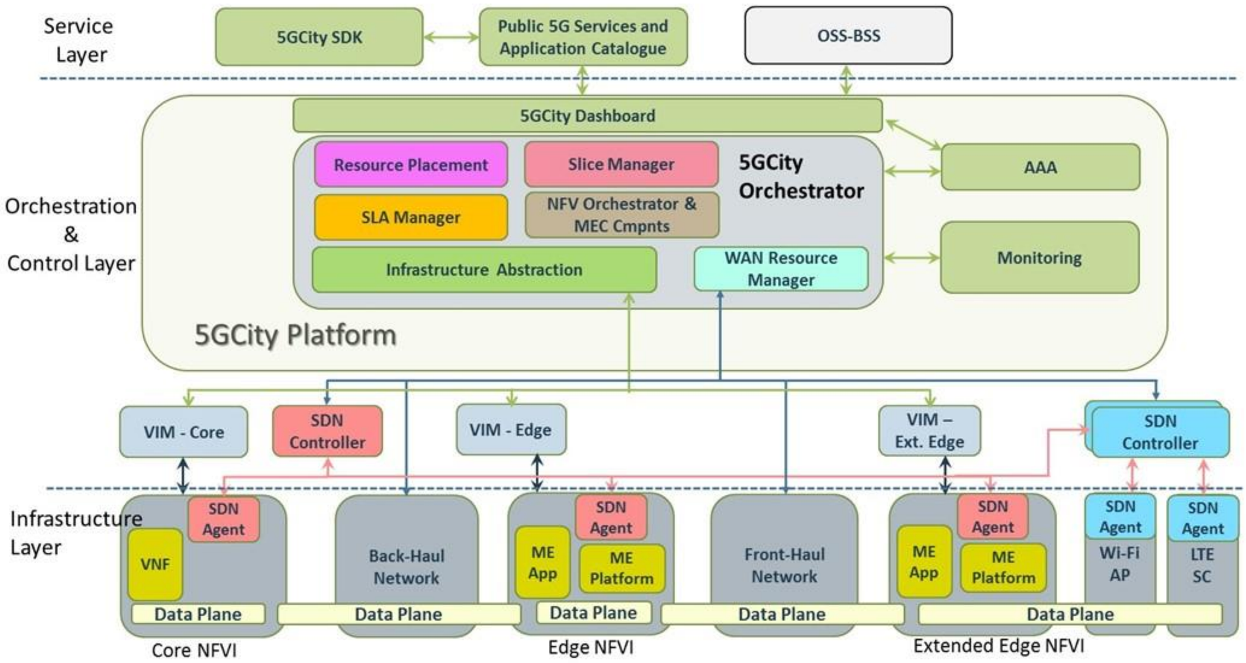

Figure 1: 5GCity architecture

1.2. Actors and Roles Description: definitions update

The 5GCity Architecture is designed taking into account also the roles that the different actors play; in particular we focused on the aspects related to the network services or application services development, the storage, and the access to the Service Catalogues using the 5GCity platform and those aspects related to the offering of the end-to-end network slice to both the Slice Requester and User.

Figure 2 shows the roles and actors envisioned in D2.1, [1].

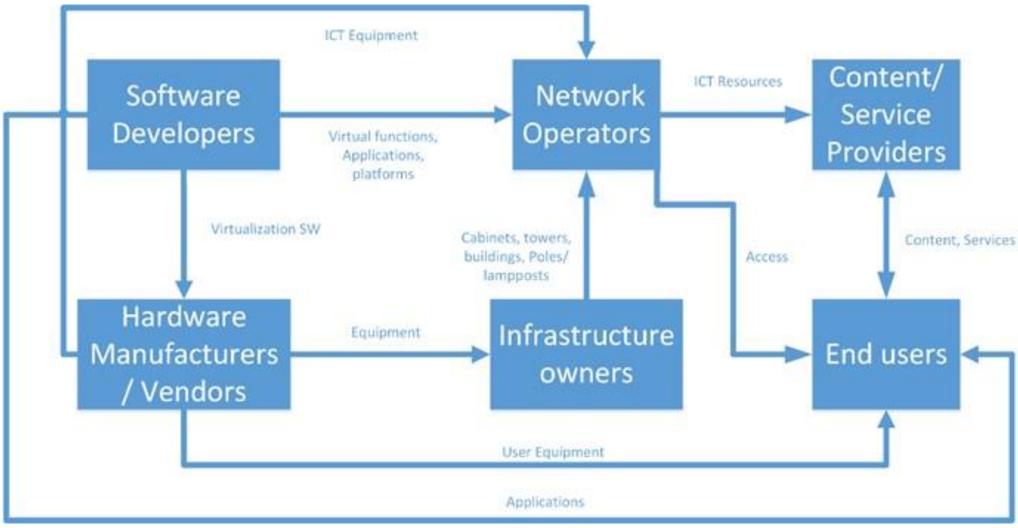

Figure 2: Relationships between Actors

Although the relationship between the Infrastructure Owner and the Network Operator can have many different forms and implications on a "business model"-level, the decisive aspect for the 5GCity system from a technical perspective is the fact that they (or one of them; or even a broker…) will be the 5GCity platform operator (being the so-called neutral host). Therefore, in the following we describe a simplified view of the main technical 5GCity "roles", as well as different scenarios/possibilities about the (combinations of) actors/stakeholders that can play those roles.

Table 1 summarizes the technical roles in the 5GCity framework and the actors that are expected to play them.

5GCity Role	Actors
Service Developer	Application providers, free-lance developers, etc.
Neutral Host	Infrastructure Owner, Network Operator, InfO+NO
Slice Requester/User	Content Provider, Service Provider, virtual Network Operator

Table 1 - Roles and Actors in the 5GCity Framework

A general scenario is described in Figure 3, in which the roles and actors can be understood as follows:

- **Service Developer** is the *role* that uses 5GCity SDK to develop network services or application services, which are stored in Service Catalogues to be accessed by 5GCity platform.
- Content/Service provider, Virtual Mobile Network Operator, and Verticals are *actors*, which take the *role* of **Slice Requester/User**.
- **Neutral Host** is the *role* that offers the end-to-end infrastructure slice. This *role* can be taken up by Infrastructure Owner – InfO (Luca use case), Network Operator – NO (Barcelona city use case), InfO+NO (Barcelona city use case), or a broker entity. In 5GCity, Neutral Host is the same entity that is also the 5GCity Platform Operator.

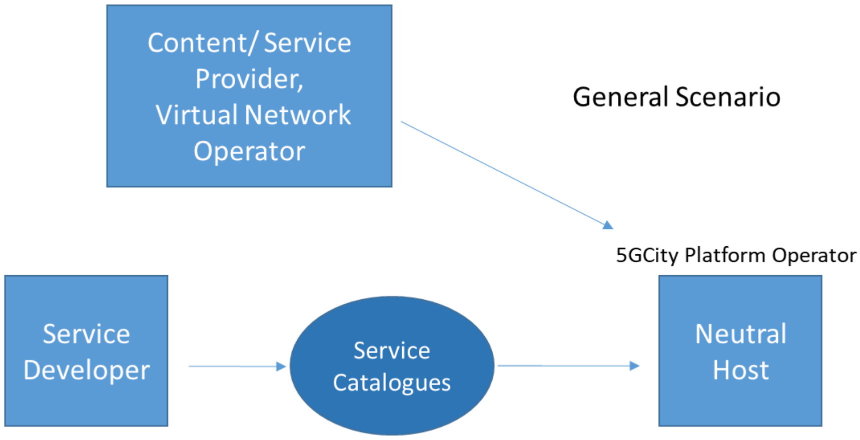

Figure 3: Relationships between Actors in a General Scenario

By looking deeper into the relationships of Network Operators with Infrastructure Owners and the different possibilities about who will operate the 5GCity platform and act as a neutral host, we come down to three concrete scenarios, which are relevant for different Use Cases of the project.

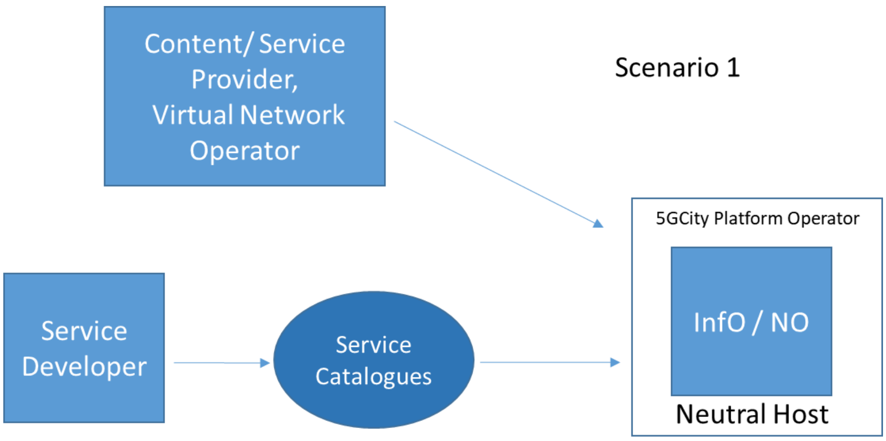

Figure 4: Scenario 1: Infrastructure Owner and Network Operator collapsed in one role

Figure 4 shows a scenario in which the different actors InfO and NO are collapsed in one role, which is also the 5GCity platform operator, and hence act as the Neutral Host. This scenario may apply to Barcelona city, as well as in the case where Cellnex owns the infrastructure sites as well.

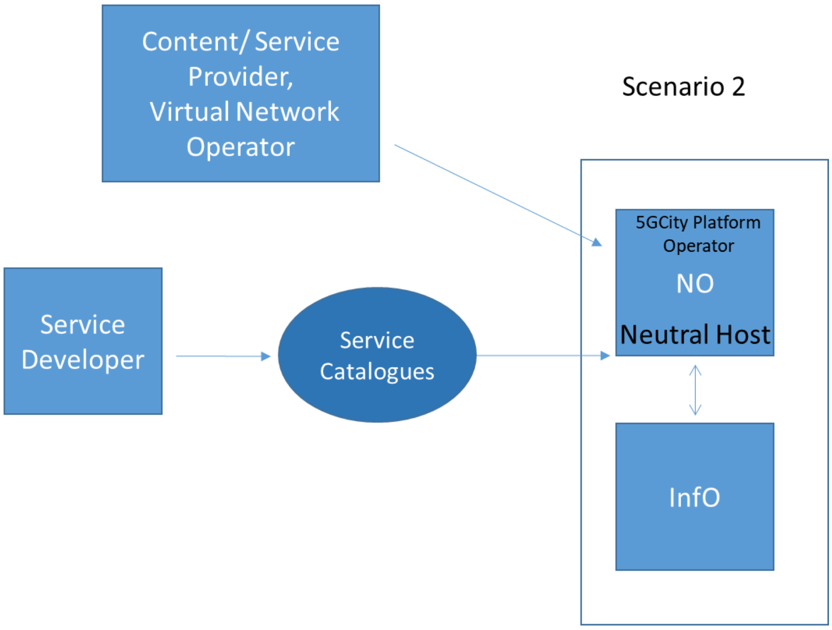

Figure 5: Barcelona City scenario

However, the scenario in Figure 5 is more probable to represent the Barcelona city situation, where a Network Operator (Cellnex) manages its infrastructure at sites owned by a municipality. Also, it buys equipment from vendors and installs them in the infrastructure sites. The relation between NO and InfO is managed between them and does not directly concern 5GCity operational workflows. For Barcelona it is important to evaluate several infrastructure configuration in order to adapt neutral operator architecture to different urban models. These items are focused in MEC adequate installation point (pole, cabinet, macroblock technical room or data center) and definition of interconnection reference point between Neutral Operator and Conventional Operators (basically macroblock level and data center level). Barcelona City council would be use a public procurement processes to select Network Operator roles. In case of 5GCity project deployment Cellnex and IMI implement this role.

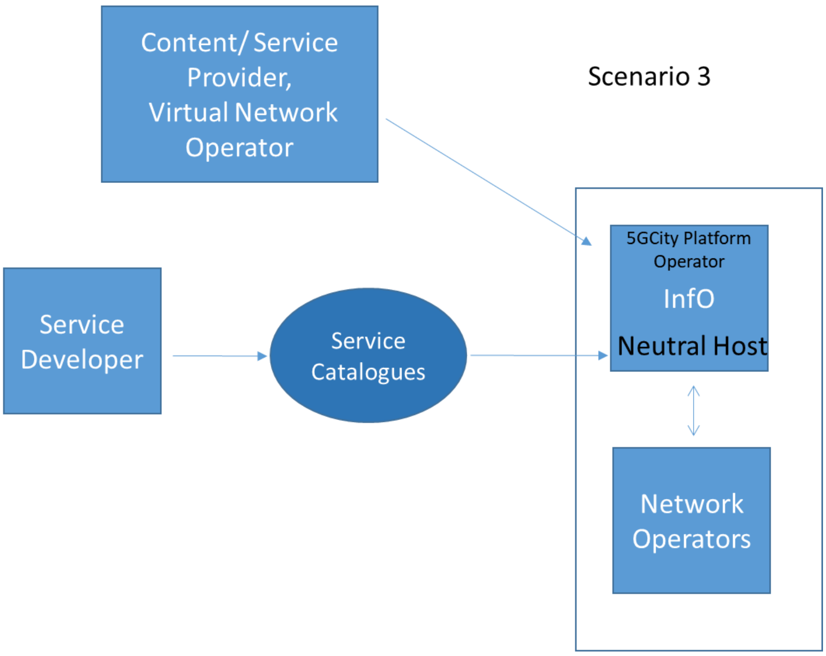

Figure 6: Lucca City scenario

The scenario shown in Figure 6 represents the Lucca city situation, where a municipality manages network connectivity for its infrastructure through a network operator itself. Similarly, it buys equipment from HW vendors and installs them in its infrastructure sites. The relation between InfO and NO is again managed between them and does not directly concern 5GCity operational workflows.

In addition to these scenarios, it is also thinkable that a broker operates the 5GCity platform and does pure neutral hosting business, hiding its relationships with the InfO and NO from the Slice Users.

1.3. 5GCity Architecture and Business model scenarios

In the last years, the traditional telecom market is constantly shrinking due to several reasons such as service price reduction, regulatory interventions and the introduction of the role of Over-The-Top players that are pushing telecom operators. The advent of new technologies has significantly changed the telecom landscape. These technologies have promised improved performance and are expected to open new business opportunities to all the involved players in order to thrive in the future demanding telecom environment. It is anticipated that new players (infrastructure providers, facility managers, network functions developers etc.) will enter the market while the role of existing ones (telecom operators, service providers) will be enhanced. In addition, services can be provided from different stages of the value chain.

Among others, 5G networking leveraging the concepts of software defined networks (SDN) and network functions virtualization (NFV) promises unparalleled speeds, low latency and enhanced security/reliability. Taking into account the improved performance, 5G is expected to support several use cases that can be included in the broad families of the following scenarios:

- eMBB (enhanced Mobile Broad Band) → Providing high throughput;
- mMTC (massive Machine Type Communications) → Providing high number of connected devices;
- URLL (Ultra Reliable Low Latency) → Providing very low latency systems (up to 1 ms).

In order to achieve these enhanced capabilities, 5G will make use of several concepts and technologies like the densification of networks through the deployment of small cells or the implementation of MECs (Multi-Access Edge Computing). However, the high number of small cells required in order to accommodate the increased demand, makes economically unsustainable the deployment of parallel access networks. This limitation along with the new network management capabilities offered by 5G could reveal a new player/business model known as Neutral Host Model that enables a single physical network to support a number of virtual networks and heterogeneous systems and functions with different performance characteristics in a variety of different locations as required by the service demand.

Towards this direction, 5GCity architecture adapts distributed cloud technologies in order to build its combined edge and network infrastructure providing a multi-tenant, cost-effective platform for deploying virtualized, heterogeneous services and supporting the defined use cases:

- Unauthorized Waste Dumping
- Neutral Host
- Video Acquisition and Production
- UHD/360º Video Distribution
- Mobile Backpack Unit for Real-time Transmission
- Cooperative, Connected and Automated Mobility (CCAM)

The 5GCity architecture allows, from the business point of view, the exploitation of the Neutral host model, taking into account the features of the different layers:

- Application Layer: offers the tools and functions available for the operators of the infrastructure, customers, subcontractors, and third parties' actors
- Orchestration and Control Layer: allows to manage a non-homogeneous set of physical resources, abstract physical resources, operate a horizontal slicing thus providing coherence cast end-to end services tailored to a multi-tenant framework
- Access Layer: the Neutral Host model requires some report and information to deal with the access layer (end-user devices - physical resources providing connectivity); in the contract is necessary to define how to deal with the complete chain from Radio elements, Fronthaul network to the end-user devices in order to check the availability of the service and the quality.

The 5GCity architecture is able to offer the instruments to:

- Provide a list with all available resources, applications and services
    - For applications: catalogue with description, requirements and the type of license for each application
    - For services: catalogue with description and requirements
    - For resources: catalogue (type, coverage, location, etc.)
- Design the solution that will include:
    - Applications
    - Services
    - Resources (GUI with geolocation)
- Define an SLA agreement with some terms regarding:
    - Details about the resources (number, type, geolocation, etc.)
    - QoS guaranteed levels
    - Charging options
- Ensure that the terms of the agreed SLA are covered:
    - Monitor KPIs
    - Report if KPIs are not met (association with charging)
- Charging mechanism based on different mechanisms (one or a combination of below):
    - Flat Rate
    - Usage (capacity, time)
    - QoS
    - Difference depending on utilization of the network (busy hours)
    - Security level provided
    - Other
- Network management:
    - What level of management is allowed (if any)
    - Monitoring of resources
    - Expand / reduce number of resources
    - Change QoS
    - Interconnect with existing network
- Support/problem reporting:
    - automatic
    - contact people
- Security:
    - Isolation
    - Different levels of security
- Maintenance

- Regulatory compliance

For Telcos all these elements are important to have the same schema to deploy and manage the services.

Several services can be provided using the 5GCity platform. These service applications will have different usage and, thus, they can be categorized depending on their objectives. For each type of service, we can implement the Table 2 in order to have the complete description and the involved resources.

Parameters	Description
Name of Service	
Type of service	eMBB, mMTC, V2X, eHealth, Tourism, Industry 4.0
Rate of service	Data rate required for this service.
Technical specs of the service	Data rate Fast Rerouting / MPLS Interfaces 1/10 Gbps
Coverage	Area Coverage
Quality of service indicators	100% Guaranteed bandwidth Latency Throughput
Resources	Freq. Bands Network (back haul, front haul), MEC, Slicing

Table 2 - Services and involved resources

In the following subsections, the Neutral Host model is presented using the powerful tool of the Business Model Canvas. It should be highlighted that this is an initial version of the Neutral Host business model that will be further elaborated in deliverable D2.5, due in M30.

We have applied and investigated the "Business Model Canvas" to the different scenarios/markets with regard to the Neutral Host model.
In the first scenario (Table 3 and Figure 7), 5GCity can be assumed as a platform through which the Neutral Host operator will develop and provide "network slices" consisting of a set of virtualized resources that share the same physical infrastructure to different tenants.

In the second scenario (Table 4 and Figure 8), 5GCity can be seen as a market place where developers will have the opportunity to upload their services. Moreover, other stakeholders like service provides will be able to combine functions or/and services from the market place in order to develop their service.

In the above cases, there are a set of important points such as resource allocation, resource management, BSS&OSS, network resources, network management, Support and Maintenance and Security and Regulatory compliance the operator will have to manage with. These can be seen as a list of requirements that should be mapped into the architecture that will be described in the following sections.

1.3.1. Business Model: Provider of network slices

<table>
<tr><td rowspan="2">Key Partners

Manufacturers
Deployers
Integrators</td><td>Key Activities

Configure services</td><td rowspan="2">Value Proposition
CPU
Router
Site
Ethernet connectivity</td><td>Customer Relationships
SLA</td><td rowspan="2">Customer Segments

Operators
Service providers
Brokers</td></tr>
<tr><td>Key Resources
Physical resources</td><td>Channels</td></tr>
<tr><td colspan="3">Cost Structure
Capex of infrastructure</td><td colspan="2">Revenue Streams
One shot and maintenance</td></tr>
</table>

Table 3 - Infrastructure/Network Provider Business model relationships

As described above, an infrastructure owner offers the physical infrastructure resources to several services providers. Software techniques as Network Slicing will allow the different Service Providers to access those resources they may need for each specific service in the same or different locations, other software techniques like NFV (Network Functions Virtualization) will allow the service operators be able to offer different services over those same resources.

There will be a catalogue of provided resources that the infrastructure provider will be able to make use to best suit their business cases (Figure 7):

- SDN
- MEC computing
- Data Storage
- 10/100Gigabyte
- Macro RAN
- Nano Cell RAN
- Cabinet
- Metro DC
- Frequency coverage

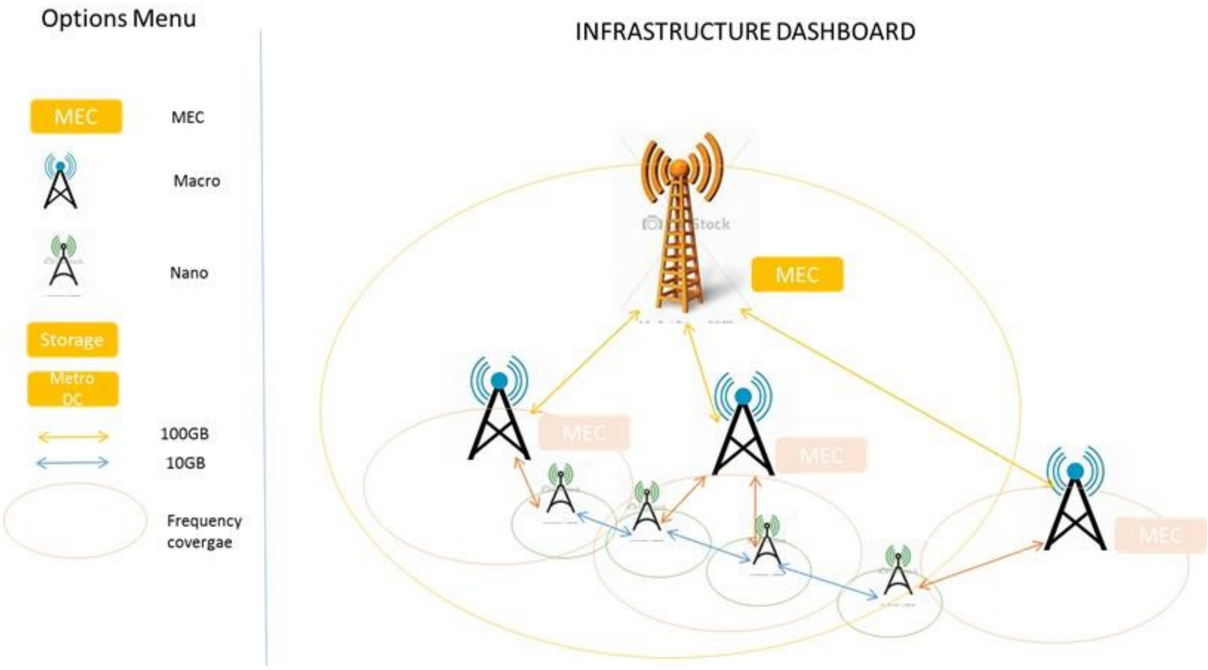

Figure 7: Infrastructure Dashboard

1.3.2. Business Model: Network functions and applications market place

In this case, 5GCity platform can be considered as a market place where 5GCity SDK can be used by developers in order develop network services or application services, which are stored in Service Catalogues. Moreover, Service providers can also access the platform in order to use and combine the developed services/applications.

Key Partners	Key Activities	Value Proposition	Customer Relationships	Customer Segments
Infrastructure Network Manufacturers	Configure services Neutral Host	Slicing SLA Infrastructure agnostic eMMBS RLL mMTC		MNO Any enterprise
	Key Resources		**Channels**	
	Network Slicing Virtualized Functions		5GCity Providers Distributors	
Cost Structure		**Revenue Streams**		
Platform Development Service Platform Management Management of Operational Chains		SLA agreement Maintenance		

Table 4 - Service Provider Business model relationships

There will be a catalogue of provided functions in the platform that the Service Providers will be able to make use to best suit their business cases (Figure 8):

- Network Slicing
- Virtualized Functions
- Bandwidth
- Latency
- Massive IoT

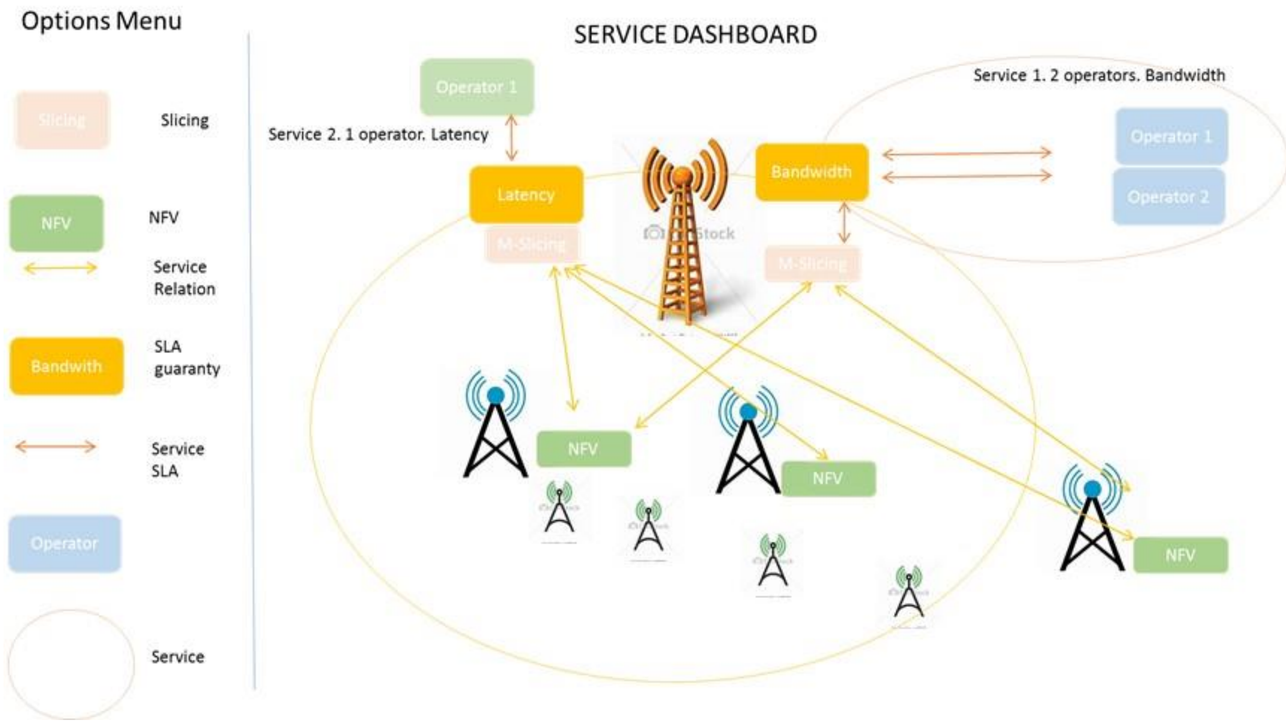

Figure 8: Service Dashboard

There will be a catalogue of provided services that service providers will be able to make use to best suit their business cases:

- eMBB (Multimedia delivery, Home Entertainment, Virtual Reality, Augmented Reality, Gaming, etc.)
    - Requirements:
        - High data rate front-haul and back-haul
        - Cell Density
        - Network and service management
    - Resources:
        - High data rate Network
        - Network Slicing
        - MEC (Mobile Edge Computing)
        - Virtualization of functions
        - RAN resources (Small Cells)
- Automotive (Autonomous vehicle, Infotainment, Platooning, Remote monitoring, etc.)
    - Requirements:
        - Very Low Latency Network

        - Wide access area
    - Resources:
        - MEC
        - Small Cells deployment

- Industry, Energy and Smart Cities
    - Requirements:
        - Massive device communications
        - Low latency
    - Resources:
        - RAN deployment
        - Virtualization of functions

- Health Care
    - Requirements:
        - Very Low Latency
        - Narrow access area
    - Resources:
        - High data rate Network
        - Network Slicing

2. Service Layer

The Service (& application layer) consists of a specific set of functions/tools of the proposed 5GCity architecture available for the operators of the infrastructure, their customers, subcontractors and any third party actor.

Its main blocks are described in the following paragraphs.

2.1. SDK

The SDK toolkit is a self-contained, stand-alone software platform that provides a set of services to support the design of network service templates, ready for the deployment upon a generic NFV MANO infrastructure.

The SDK toolkit is able to selectively offer different functionalities depending on the user/role who is exploiting its functionalities. This design ensures a high level of configurability of that SDK is able to be deployed in wide range of scenarios, by wide range of users, and agnostic with respect to the underlying infrastructure.

The description of the design for the 5GCity SDK toolkit is provided in Figure 9.

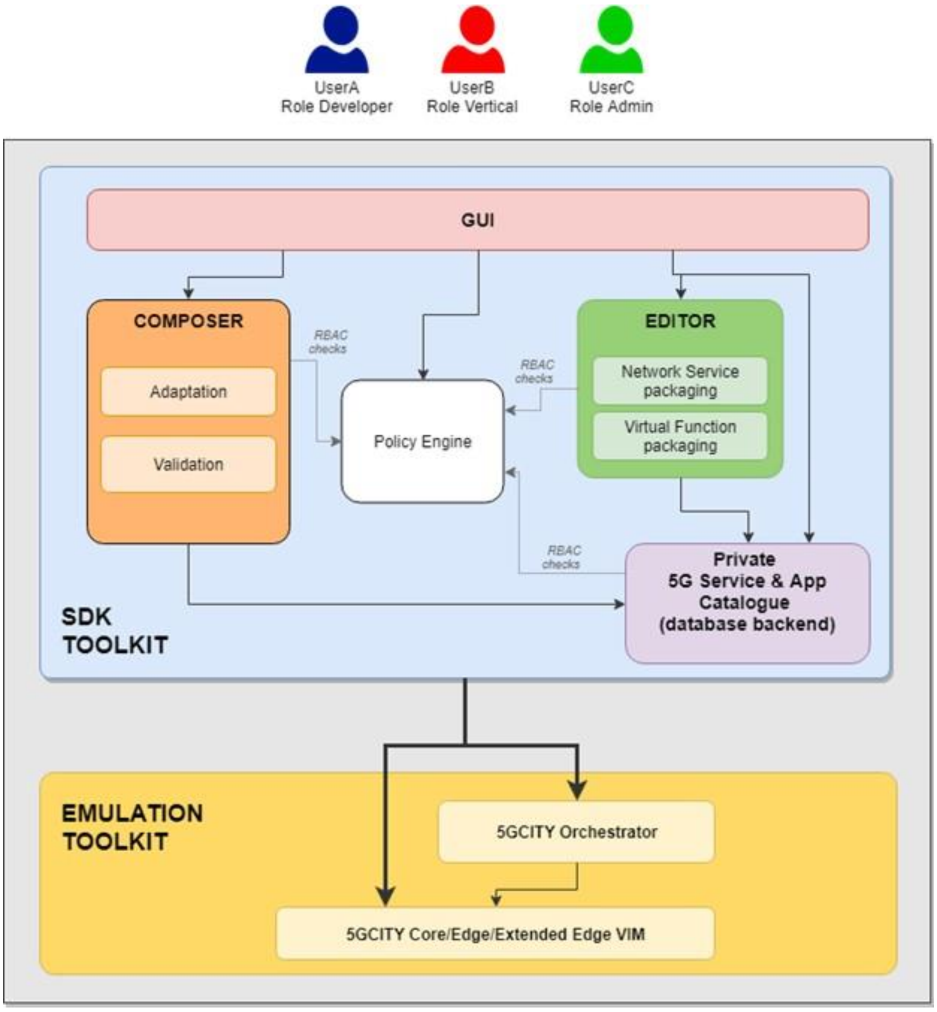

Figure 9: SDK Toolkit high-level architecture.

5GCity SDK toolkit has been designed having in mind that the same platform has to be used by two different user roles: the first role (role=developer) consists in a user profile which is fully aware of the low level NFV MANO details, is able to design a Network Service as a composition of all low level entities described in ETSI GS NFV IFA 014, [7], while the second role (role=vertical) utilizes the SDK in the fashion of a Wizard tool, i.e. he is offered a simplified view of the Network Service Element (which is embodied by simplified NFV information model) and the SDK toolkit itself is in charge of translating the simplified view of the vertical user into a fully compliant ETSI NFV. 5GCity toolkit contains two separate modules, namely Composer and Editor which implement the different functionalities required by the envisioned roles.

The full 5GCity SDK toolkit architecture is composed by the following main functionalities:

- **Graphical User Interface** represents the entry point for the platform and allows user to interact with services exposed by SDK toolkit. Operation performed by the user at GUI level are translated in API calls towards Composer and Editor blocks
- **Composer** is a set of tools which allows the user (Role=Vertical) to manage Network Service items by providing the following capabilities:
    - Intercept commands (Create modify, delete operations on Network Service descriptors) issued by user via GUI
    - Interact with local database with the main purpose of storing/retrieving NFV items
    - Adapt the 5GCity SDK toolkit information model to fully ETSI NFV information model
    - Validate the network service items created by user
- **Editor** (OPTIONAL, described here only for reference) is a set of tools which allows the user (Role=dev-ops) to manage Network Service items by providing the following capabilities:
    - Intercept commands (Create modify, delete operations on Network Service descriptors and VNF packages) issued by user via GUI
    - Interact with local database with the main purpose of storing/retrieving NFV items
    - Validate the network service items created by user
- **Policy Engine** is the function block which implements RBAC access to resources, by checking the matching between user/role which issue the request and the requested resource
- **Private App service and Application Catalogue** is a collection of services which provides the following capabilities:
    - A dispatcher engine which takes care of adapting the API calls sent-received on North Bound interfaces in order to be served by local database and MANO plugin
    - A local database where the NFV items packages (NS, VNF) are store according a full ETSI information model
    - A MANO plugin which is able to translate a generic NS descriptor to a NS descriptor which is related to a specific MANO stack
- **Emulation Toolkit** (OPTIONAL, described here only for reference) provides an emulated full stack MANO framework where the user is allowed to deploy the Network service designed by means of the 5GCity SDK Toolkit.

The 5GCity SDK toolkit has been designed in order to implement an RBAC policy (Role Base Access Control) to grant selected access to the resources. A clear definition of user and roles is provided in the following table, together with a matrix that states correlated the roles with the operations they are allowed to perform on the resources.

Role Type	Role Description	Resources accessed and allowed operations
Role=Admin	is in charge of the configuration and maintenance of the SDK framework with full access (super-	• Life Cycle management of the platform.

	user) to each of the components composing the SDK	
Role=Vertical	has access to a basic set of SDK functionalities, mainly consisting in the editing of a service item as an ordered set of given (atomic) functions. The Vertical role is exposed a simplified-abstracted view of 5GCity resources.	• Full R/W access to NSd • ReadOnly access to VNF packages
Role=Developer	which has access to the full capabilities of 5GCity-SDK, being able to create/modify/delete NSd, VNF packages.	• Full R/W access to NSd • Fll R/W access to VNF packages

Table 5 - 5GCity SDK roles definition and access to catalogue resources

The SDK Toolkit has been designed to be agnostic w.r.t to the NFV MANO infrastructure where the networks service will be deployed. This agnostic behaviour is implemented by taking into account that:

- The SDK Toolkit is able to crate network Service deployment templates according an ETSI NFV generic information model, as specified in (ETSI GS NFV-IFA 011 V2.4.1 (2018-02))
- The network Service deployment template produced by SDK Toolkit is then translated to a Network Service tailored to an information model specific to the MANO framework where the Network service will be deployed

This translation between information models is deployed as a specific service in the Public 5G Service & Application Catalogue.

2.2. **Application Catalogue**

The Public 5G Service and Application Catalogue is a stand-alone platform, that acts as an exchange point between NFV SDK platforms and NFV MANO enabled infrastructure. The platform can be as a market place where:

- Generic NFV SDK user can upload Network Service Deployment templates;
- Generic NFV MANO user can download and on-board network services upon their NFV MANO infrastructure.

The Application catalogue has been designed to offer a great degree of interoperability with several NFV SDK toolkit and several NFV MANO framework, as depicted in Figure 10:

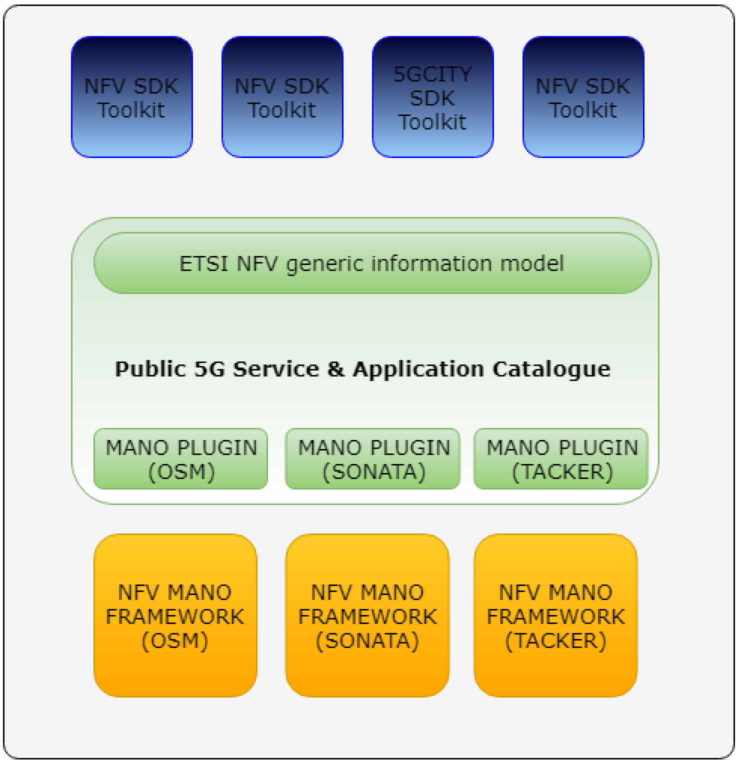

Figure 10: Interoperability among NFV SDK, NFV MANO framework and Public 5G Service & Application Catalogue

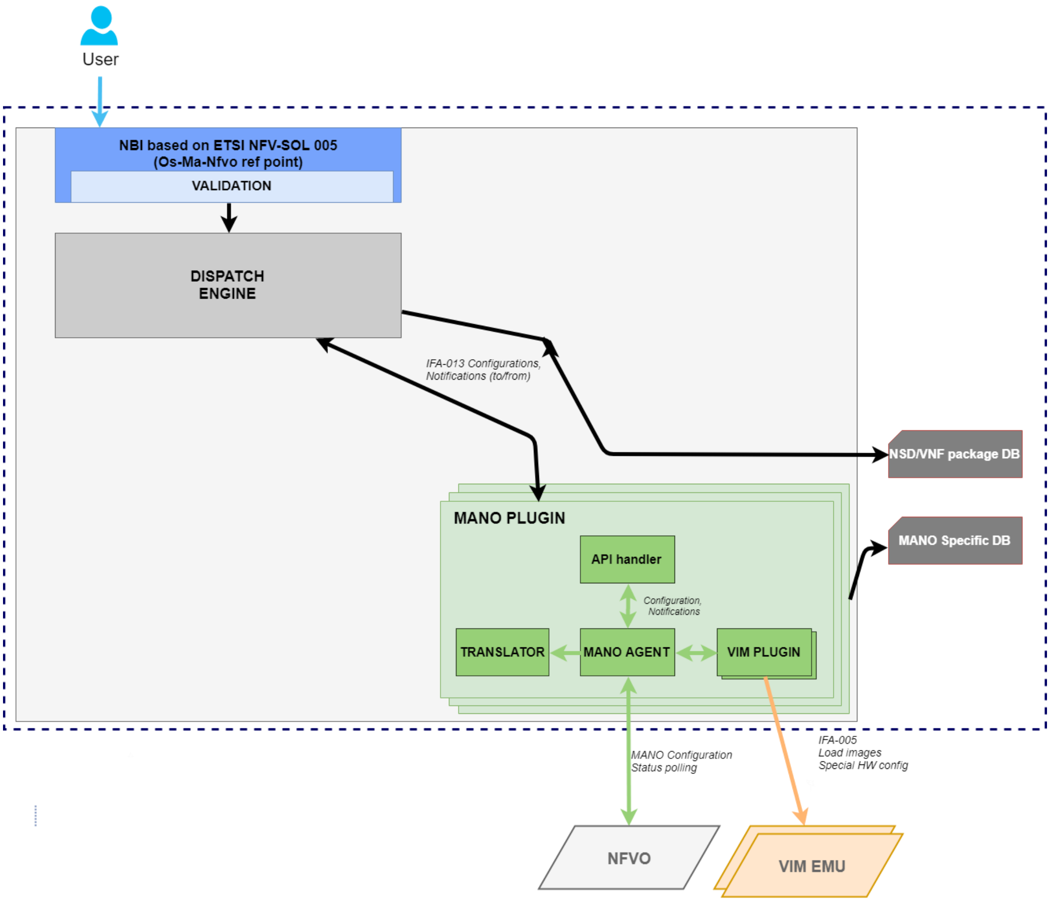

Figure 11: Public 5G Service & Application Catalogue high level design

The Private Catalogue design and high-level architecture is described in Figure 11 and it is composed of the following main elements:

- Dispatch Engine which is in charge of delivering messages among catalogue main components;
- Database back-end where network Service templates and VNF packages are stored;
- MANO plugin which is in charge to translate a deployment template according the generic ETSI NVF information model, to an information model which is specific to the MANO platform deployed.

Several MANO plugins, specialized for each specific MANO platform, can coexist, allowing the 5G Service & Application Catalogue to interoperate with a wide range of different MANO platforms.

2.3. **OSS/BSS**

The OSS/BSS service provides Operating and Business supports functions in order to manage 5G network infrastructure (OSS) as well products, customers and customer life cycle (Orders, Upgrades, Complains, Usage & Invoicing, Leave) of the services provided by 5GCity framework (BSS), see Figure 12.

The need for highly automated and standardized processes is particularly stressed in 5G environments due to frequency/network/IT resource sharing in Product/services deployment with the target of making customer experience, services and network resources part of one orchestration.

Therefore, the integration should be based on “Open API” provided by the 5GCity framework (via service orchestration, usage if needed and management alerts).

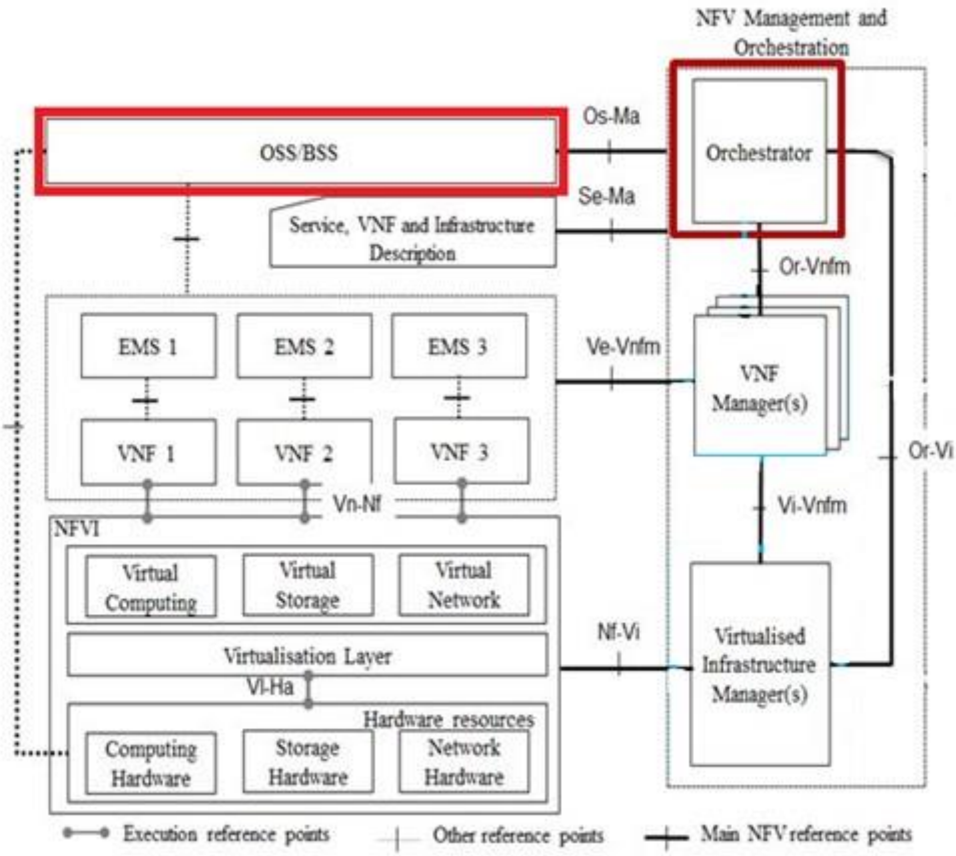

Figure 12: General schema 5GPPP

Additionally, as the Network Infrastructure moves towards virtualized infrastructures on COTS hardware, OSS and BSS systems evolve the a virtualization path based on IaaS, PaaS, or SaaS, by using a common commercial hardware and storage, and by providing separate virtualized infrastructure, virtualized platform, and virtualized software to a set of tenants using the shared physical infrastructure. Specific application environments can be than made available for vertical implementations, including Artificial Intelligence and Machine Learning functionalities in order to provide enhanced analytics needed for a quick service time to market.

In the following, the principal features expected from these modules are introduced; the general 5GCity architecture has to take into account what is available to use according to the Business model adopted.

2.3.1. Product/Service Catalogue Management

The definition of a product is an item that satisfies a market's want or need. Product/Service Catalog Management is the ability to create and maintain products that can be sold to customers in the target market. More specifically, it is the ability to explicitly model the structure of a product, then create and centrally manage the instances (or "catalog") of products based upon that structure. Products are not always discreet, single items. A product can be a number of components associated together and sold as a single purchasable entity. Therefore the product may be comprised of multiple components, tangible or intangible, such as services, features, devices, etc., that are "assembled" together to form a single sellable entity. Some of the components within a product will be enabled by shared/common/reusable services (e.g., location finder). Some of the components within a product will be enabled by shared/common/reusable resources (e.g., network exchange). These underlying services and resources may be managed by different parts of the organization.

The Product Management organization is typically responsible for managing the Product/Service Catalogue through the assembly and update of products utilizing available components of the Public 5G services and Application Catalogue.

2.3.2. Customer Contact Management, Retention & Loyalty

Customer contact management, retention and loyalty applications are a varied group of functions that are generally sold as part of a Customer Relationship Management (CRM) suite of applications. In general, these applications allow an operator to create, update and view the customer's information (names, addresses, phone numbers, organizational hierarchy), record and view all customer interactions across different communication channels and department. This way whoever is speaking to a customer can see the history of issues that have concerned that customer, be they order issues, billing enquiries or service problems. More sophisticated systems allow capabilities to highlight customers as risk of switching to an alternative carrier (churn indicator) and provide comparisons with other operator's service packages to allow customer care agents to try to persuade a customer that their current operator can provide the best value for money. These indicators can be provided via integration to business intelligence platforms.

2.3.3. Corporate Sales Management

2.3.3.1. Overview

The Corporate Sales Management application provides the necessary functionality to manage the sale to a medium to large business customer. This is fundamental in the case of Neutral Host model. Besides, given convergence occurring in the industry in the area of products and customer classification, along with the need to manage relationships across corporate and mass market customer segments, it is expected that at some point significant aspects of the Corporate and Mass Market Sales Management applications will come together into a single set of applications.

2.3.3.2. Functionality

Starting from the Telecom experience, Corporate Sales Management includes the following:

- Management of sales accounts, including the handling of leads and funnels.
- The design, price, propose lifecycle.
- This could include responding to RFIs, RFBs, etc., as well as a number of iterations on the Design/Price/Propose lifecycle (aspects fundamental in the Neutral Host model).
- Negotiation and closure on a formal contract with the customer.

2.3.4. Contract Management

2.3.4.1. Overview

Contract Management applications provide necessary functionality to facilitate a contract pertaining to a given solution.

2.3.4.2. Functionality

Contract Management handles the creation of the customer's contract and any associated service level agreements, including approval of custom language, customer contract sign-off, appropriate counter signature and contract expiration. Elements of the contract will flow through to ordering, assurance, and billing processes.

2.3.5. Customer Order Management

Customer Order Management applications manage the end-to-end lifecycle of a customer request for products. This includes order establishment (step guiding, data collection and validation), order publication as well as order orchestration and overall lifecycle management. A customer request may also pertain to already purchased product(s). Thus, the Customer Order Management application handles order requests to suspend, resume, change ownership, amend, add, change and discontinue existing ordered products. Customer Order Management application should support repackaging of the purchased offers into alternate product offering (may require sales/contract negotiation). Customer Order Management applications typically serve all the customer touch points/channels, including call center, retail, self-service, dealers, affiliates, etc. The order may be initiated by any channel and visible to the other channels if needed. The general schema has to be tailored according to the Neutral Host model.

2.3.6. Service Order Management

Service Order Management applications manage the end-to-end lifecycle of a service request. This includes validating service availability as well as the service order request. Other functionality includes service order issuance, service and or product order decomposition, and service order tracking along with orchestrating the activation and the test and turn up processes. Notifications will be issued to the Customer Order Management during the service order orchestration process (especially upon completion). Such notification can trigger other steps in the Customer Order Management (e.g. service order completion concludes these steps with Customer Order Management).

3. Orchestration & Control Layer

The orchestration and control layer of 5GCity includes the entry point of network services (Dashboard), the core orchestration components (5GCity orchestrator), as well as the controllers that reside between the central orchestration platform and the infrastructure, namely WAN managers, VIMs, and SDN controllers. The scope, high-level functionalities, and relationships of these elements are described in this section.

3.1. Dashboard

5GCity's dashboard is aimed to be the main interface to the majority of 5GCity's stakeholders. As depicted in Figure 13, the Dashboard is envisioned to be 5GCity's "top" component functioning as the single-entry point for 5GCity's platform end users as well as third party applications aiming to interact with the system.

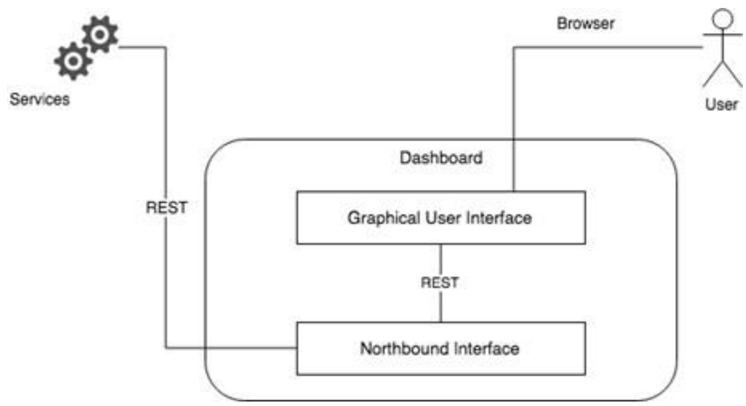

Figure 13: Dashboard High Level Architecture

This allows to check that the Dashboard is aimed to be composed by two main sub-components:

- **Graphical User Interface (GUI)**
  The GUI will be the interface provided to 5GCity's platform end users exposing its features through an appealing and intuitive interface. The graphical user interface will provide different graphical environments based on the user's permission role. The graphical interface will be able to adapt its contents (visualised information) as well as enabled features targeting the action scope of each 5GCity role. The different roles envisioned to be supported by the GUI as well as its actions are presented in a section below.

- **Northbound Interface (NBI)**
  This subcomponent is envisioned to provide a REST interface to 5GCity's platform third parties (including the GUI) enabling the access to 5GCity's information and features to authorised parties. This subcomponent will provide a complete overview of 5GCIty's features and information. To achieve so, it will interact with different 5GCity's subcomponents in order to retrieve the requested information correlating multiple sources of data, if needed.

5GCity's dashboard interface aims to be appealing and intuitive allowing end-users to perform their envisioned tasks as quickly as possible. Visualisation of information is also a point to be taken into consideration providing an interface that does not overwhelm users with information allowing quick, informed and efficiency action from users. To achieve so, the dashboard will always contextualise and correlate the displayed information diminishing as much as possible irrelevant information and starring relevant information or actions that need the user's attention. Furthermore, and aligned with this principle, based on the different role's scope associated with dashboard's end users, the interface will adapt itself to the actions to be performed by each role optimising the user's interaction with the dashboard.

3.2. 5GCity Orchestrator

The 5GCity project defines, designs, and implements the 5GCity orchestration platform, which introduces the capability to offload capacity to the Neutral Host (by allowing it to manage network slices that are provisioned to network operators and service providers) and to support a large number of devices in the network. The 5GCity orchestrator is a core management component of the functional 5GCity architecture, which is responsible for lifecycle management and orchestration of all 5G-based edge services and for the control and management of the 5G and edge infrastructure available in the city.

The design of 5GCity orchestrator follows the ETSI Open Source MANO platform with the additional functionalities needed in order to support the Neutral Host concept and the combined management of edge nodes and 5G access sharing, as envisioned by the 5GCity project. These extra functionalities are classified into four main components, namely Resource Placement, SLA Manager, Slice Manager, and Infrastructure Abstraction. Figure 14 depicts the high-level architecture of the 5GCity orchestrator.

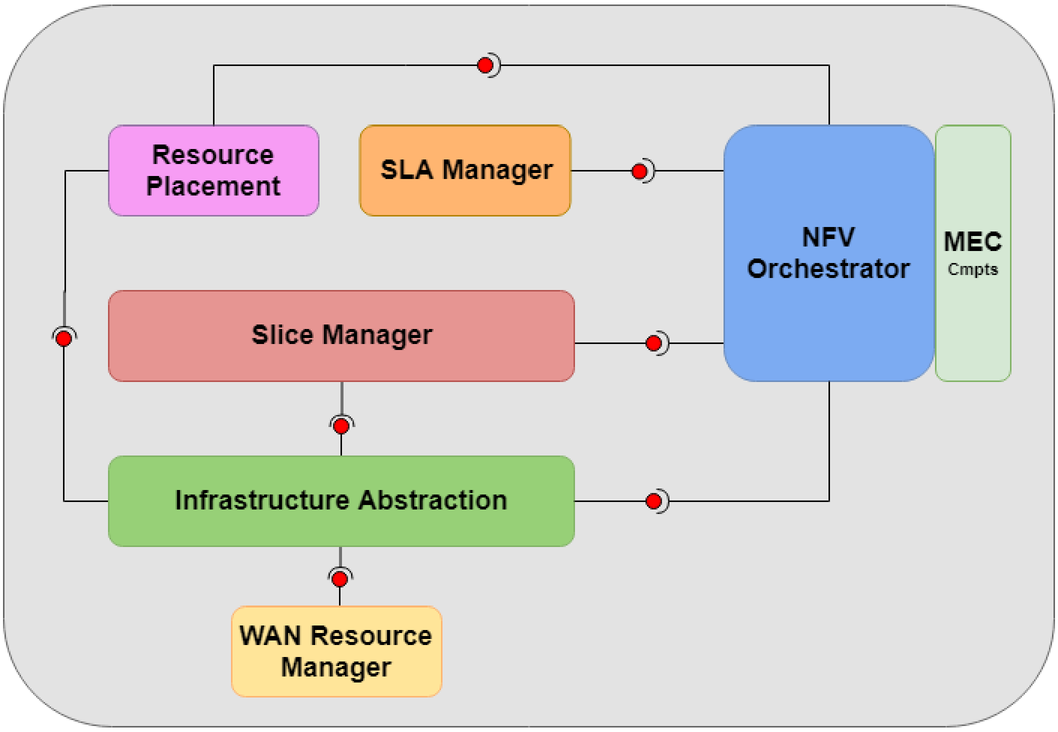

Figure 14: 5GCity orchestrator high-level architecture

A summary of the main components illustrated in Figure 14 (NFVO/MEC, Resource Placement, SLA Manager, Slice Manager, Infrastructure Abstraction, and WAN Resource Manager) is provided in the following subsections, while more details about their exact functionality and their contributions to the state of the art can be found in [4].

3.2.1. NFVO and MEC components

The NFVO shall be the core of the 5GCity Orchestrator with all functionalities of the ETSI Open Source MANO (OSM[1]) platform, which is responsible to talk to other components of the 5GCity orchestrator for managing and orchestrating the network elements. The NFVO covers the on-boarding of Network Services for which it receives requests from the Dashboard, the orchestration and lifecycle management of the physical and virtual resources, and the lifecycle management of the Virtual Network Functions (VNFs).

The NFVO shall have the functionalities described in the respective ETSI NFV specification, while including additions that are required to support (and/or incorporate) all the other components of Figure 14, as well as additions that are required for the support of Multi-access Edge Computing (MEC). The current plan is to follow the design of Open Source Mano (OSM). OSM has been chosen over existing orchestration platform in order to guarantee the performance, high service scalability, and adoption in the 5G research line. Further, the OSM is modular enough for extending its capabilities (internal and external functions/plugins). Additional reasons for selecting OSM as the core of our orchestration platform are highlighted in [4].

3.2.2. Resource Placement

Resource Placement is responsible for optimizing the allocation of Virtual Network Functions (VNFs) to physical and virtual resources of the infrastructure layer. It calculates optimal placement options by analyzing the resource usage along with further information from the Virtual Machines (VM).

3.2.3. SLA Manager

The 5GCity platform enables the Infrastructure Owner (probably identified as the City Municipality) to slice the shared physical and virtualized resources across RAN and core networks so that they can support different industry verticals for an associated SLA. Network slicing will allow 5GCity infrastructure owners to offer differentiated and guaranteed services with varying traffic characteristics on the same infrastructure. In line with the 3GPP-defined 5G slice types, we expect the typical use cases for slicing to be related with one of the following:

- Enhanced mobile broadband (eMBB) that delivers Gigabytes of bandwidth
- Massive machine-type communication (mMTC) that connects billions of sensors and machines (1 million per square kilometre)
- Ultra-reliable, low latency communication (uRLLC) that allows immediate feedback with high reliability and enables real-time remote control.

The 5G typical slicing Use Cases provide clear evidence that the design and the deployment of a network slice is strictly coupled with a set of QoS parameters, which are translated in a certain SLA.

A specific service and the associated QoS characteristics are assigned to a network slice according to a given set of parameters that ensure the above configurations. It is clear that the same 5G infrastructure which is in charge for the deployment of the networks slice, must also take care to deploy a monitoring framework capable of continuously monitor the infrastructure and check in real time if the QoS/SLA parameters coupled with network slices are respected.

The main functionalities of the SLA Manager are summarized in the following:

- It controls the lifecycle of network slices in real time and proactively manage dynamic demands and failure conditions
- It stores the SLAs related to Network Services in an internal SLA Database

[1] Please see: https://osm.etsi.org/

- It interacts with the Monitoring System based on an internal “Monitoring Driver” in order to trigger the activation on the underlying MANO infrastructure of the performance jobs, which collects monitoring data for a specific Network Service.

3.2.4. Slice Manager

Network slicing enables the virtualized and non-virtualized network elements and functions to be easily configured according to network operation requirements and isolated from the other slices in order to meet various network demands. These slices allow the virtualization of the physical network and the assignment of virtual resources (slices) to the different Mobile Virtual Network Operators (MVNOs).

5GCity’s network slicing allows network operators to seamlessly control and orchestrate services for different verticals. Each slice has its own specific requirements related to the Quality of Service (QoS) policy, cloud network resource management capability, security functions, and routing functions to name a few. The 5GCity network slicing can perform the following main functions:

- SLA-based negotiation of customizable and user-specific slices for different tenants, each of them having different requirements, which can be satisfied with flexible and dynamically deployable resources.
- Dynamic provisioning and instantiation of end-to-end network slices that include both computing and networking resources.
- Interaction with different edge VIM technologies for better support of multi-tenancy and multi-tier infrastructures.
- Seamless and dynamic service provisioning at the network level through automated processes.

The listed main functions can be enabled by combining:

- A network slice lifecycle management method that adds dynamicity with regard to the ease and frequency of slice adjustment.
- A policy management method that can enforce dynamic runtime policies over the deployed slices, such as instantiation and expiration dates.
- A slice information repository that is compatible with the –yet to be defined- information models for 5G end-to-end slices.

The above shall be reflected in the internal architecture and the non-standard features of the slice manager, which are described in detail in [4].

3.2.5. Infrastructure Abstraction

This Infrastructure Abstraction is a component of the 5GCity orchestrator that enables communication between the orchestrator and the multiple underlying Virtual Infrastructure Manager (VIM) with the support of the WAN Infrastructure Manager. For this, the Infrastructure Abstraction will require a VIM plugin and a WAN plugin in order to interact in a technology-independent manner with the Virtual infrastructure and WAN infrastructure:

- The VIM plugin shall cover the Core VIM, the Edge VIM and the Extended Edge VIM interaction requirements. It should allow the 5GCity orchestrator to interact with VIM technologies and implementations such as OpenStack, Edge-VIM (which will be an edge-oriented OpenStack extension), and Fog05[2].

[2] The http://www.fog05.io site is not reachable at the moment as it is moving the domain to the Eclipse Foundation

- The WAN plugin will expose an interface to the WAN Resource Manager element for dealing with the underlying network elements such as SDN controller, fronthaul network, and backhaul network. This keeps the infrastructure abstraction layer relatively simple and poses no restrictions on how the WAN Resource Manager controls the WAN network resources.

3.2.6. WAN Resource Manager

In 5GCity, two SDN controllers are responsible for managing and controlling network slices requested on top of the 5G neutral host infrastructure. The first SDN controller is responsible for hooking up any VNFs with the backhaul infrastructure, assuring connectivity between elements in the Metro DC and the Edge. The second SDN controller handles network slices in the RAN. Since both SDN controllers act in two different domains, yet their coordination is required to set up network slices spanning both domains, an additional architectural component is introduced: the WAN Resource Manager.

The WAN Resource Manager is responsible for translating any network slice configuration and instantiation request coming from the NFV Orchestrator for the two SDN controllers present in the architecture. Any VNF or backhaul resource requested needs to be handed over to the SDN controller that is coupled to the VIM, whereas any RAN resource requested needs to be handed over to the RAN SDN controller. Further, the WAN Resource Manager is responsible for stitching up the resources handled by the two SDN controllers so they form a single network slice.

3.3. VIM (Virtualized Infrastructure Manager)

VIM (Virtualized Infrastructure Manager) implements the functionalities that are used to control and manage the interaction of a VNF with computing, storage and network resources under its authority, as well as their virtualisation [5], [9], and [9].

Three different types of VIMs will be used in the 5GCity Architecture: VIM-Core, VIM-Edge and VIM-Extended Edge. Two of them are based on OpenStack (Core and Edge VIMs), while the VIM-Extended Edge is based on the lightweight decentralized open source technology Eclipse fog05, designed to support IoT and Fog computing environments. The interoperability among these three types of VIM will be either native (i.e. at OpenStack level between VIM-Core and VIM-Edge) or implemented through the 5GCity Infrastructure Abstraction layer provided by the 5GCity multi-layer orchestration component.

This partitioned VIM architecture and its differentiation strengthens the 5GCity capability to scale. In fact, with multiple VIMs we can more easily address the OpenStack scalability issues related to the number of compute nodes to be managed by a single VIM. Moreover, the differentiation of VIMs assures that 5GCity can benefit from both OpenStack and fog05 key features applied to the different contexts of core, edge and fog computing.

3.3.1. Core

VIM-Core operates homogeneous physical resources located in the city data center. For this role, the project will leverage on the capabilities of OpenStack [10].

3.3.1.1. OpenStack

OpenStack is developed as an open-source project with the goal of being a cloud operating system managing large-scale compute, storage and networking resources. Its logical architecture (Figure 15) is composed of modules, developed as separate projects.

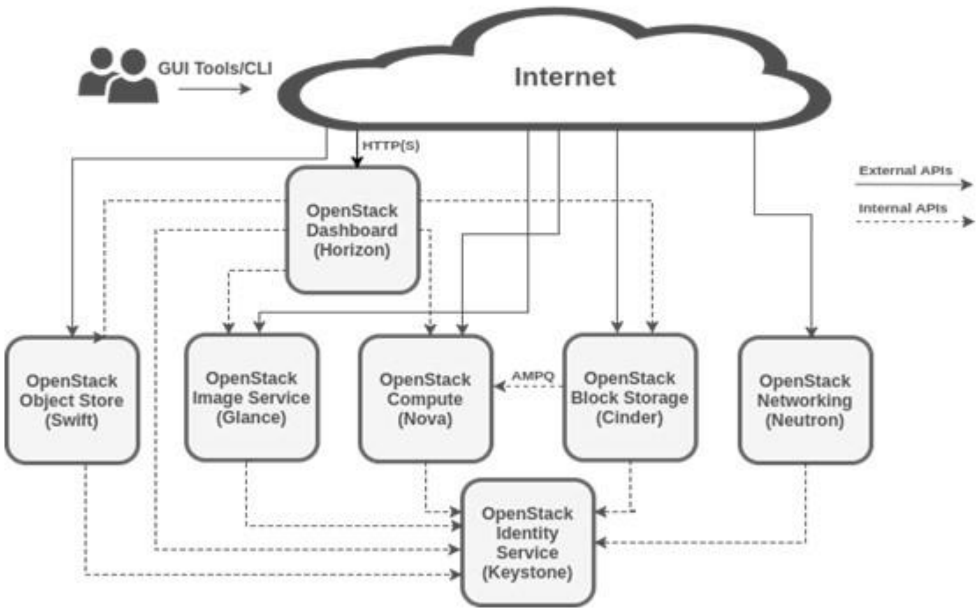

Figure 15: OpenStack architecture

- *Compute* creates and terminates virtual machines' (VMs) instances, tracks the inventory and usage, takes requests from the message queue and determines on which host to run VMs.
- *Networking* gives full control over creation of virtual network resources to tenants in the form of tunnelling protocols.
- *Image* provides users with the capability to upload and discover VM images and metadata definitions.
- *Identity* has the main purpose of authorization and authentication of users.
- *Object Store* is a highly available, distributed, eventually consistent object/blob store.
- *Block Storage* virtualizes the management of block storage devices.
- *Dashboard* provides a graphical interface to access, provision, and automate deployment.

OpenStack is based on an open-source software with an active community. It has the ability to control pools of compute, storage and networking resources and supports multi-tenancy. Its modularity, extendibility and open APIs bring added value to its users. These qualities make him the appropriate match for the role of the VIM-Core in the project.

3.3.2. Edge

VIM-Edge operates a non–homogenous, wide area, resource constrained set of physical resources located in street cabinets across the City. The requirements for software running at the edge is that it be efficient, scalable and distributable. This section presents an investigation done by VOSYS on existing open-source solutions in order to find the most suitable for the role of an edge VIM [11]. The two projects under consideration are OpenStack, presented in section 3.3.1, and OpenVIM [12].

3.3.2.1. OpenVIM

OpenVIM is part of ETSI Open Source MANO (ETSI OSM), the practical implementation of an ETSI MANO stack. It is a lightweight VIM, which follows the NFV principles with interfaces to the compute domain of the NFV Infrastructure and relying on OpenFlow controller (OFC) for the network infrastructure.

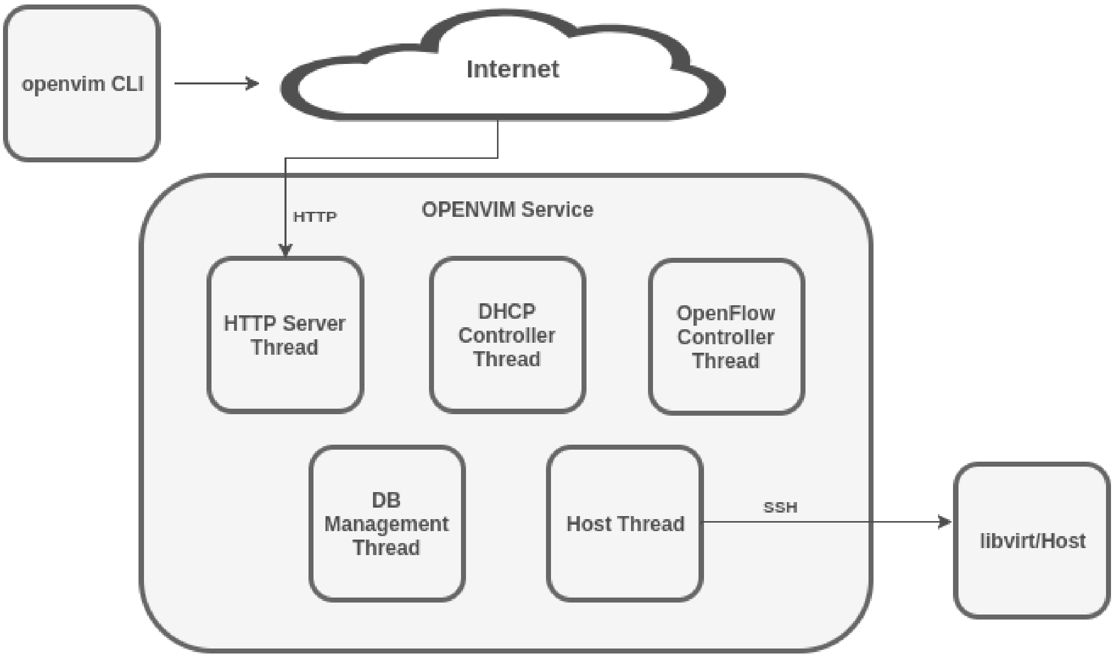

Figure 16: OpenVIM architecture

The OpenVIM service (openvimd) is the core component if its architecture (Figure 16) and is responsible for the creation, deletion and inventory of VMs, networks and images. Each functionality of the service is executed in a separate thread (DB management, HTTP server, Network controllers). OpenVIM relies on libvirt[3] and Kernel-based Virtual Machine (KVM[4]) running on the hosts to provide the virtualization infrastructure.

3.3.2.2. VIMs benchmark

In the conducted research provisioning latency was chosen as the most relevant metric for VIMs evaluation. With the goal of reusing a test platform already supporting OpenStack, we have extended the CloudBench[5] framework by implementing a new cloud adapter for OpenVIM. The following measurements were defined in order to evaluate the provisioning latency of the VIMs:

- *VM provisioning request sent* - the time overhead added by the VIM while collecting pre-provisioning data for images, flavours and existing VMs.
- *VM provisioning request completed* - the time elapsed between submitting the VM provisioning request and the VM status changed to ”running” (start of the operating system boot). It is further split in two stages: request submission and VM instantiation.

[3] Please see: https://libvirt.org/
[4] Please see: https://www.linux-kvm.org/
[5] Details at: https://developer.ibm.com/code/open/projects/cloudbench-cbtool/

The benchmark results are shown in Figure 17.

The OpenStack architecture is meant to be flexible and expandable and allows additional services to be added. The communication between modules is done through messaging which brings overhead to the systems but allows horizontal scaling. OpenVIM is created with the intention to provide the minimal functionalities of a VIM and consists of a single service spawning multiple threads. This leads to a more efficient implementation but with less possibilities for extension. It lacks many of the complementary services of OpenStack – Storage, Identity, Dashboard, Telemetry etc.

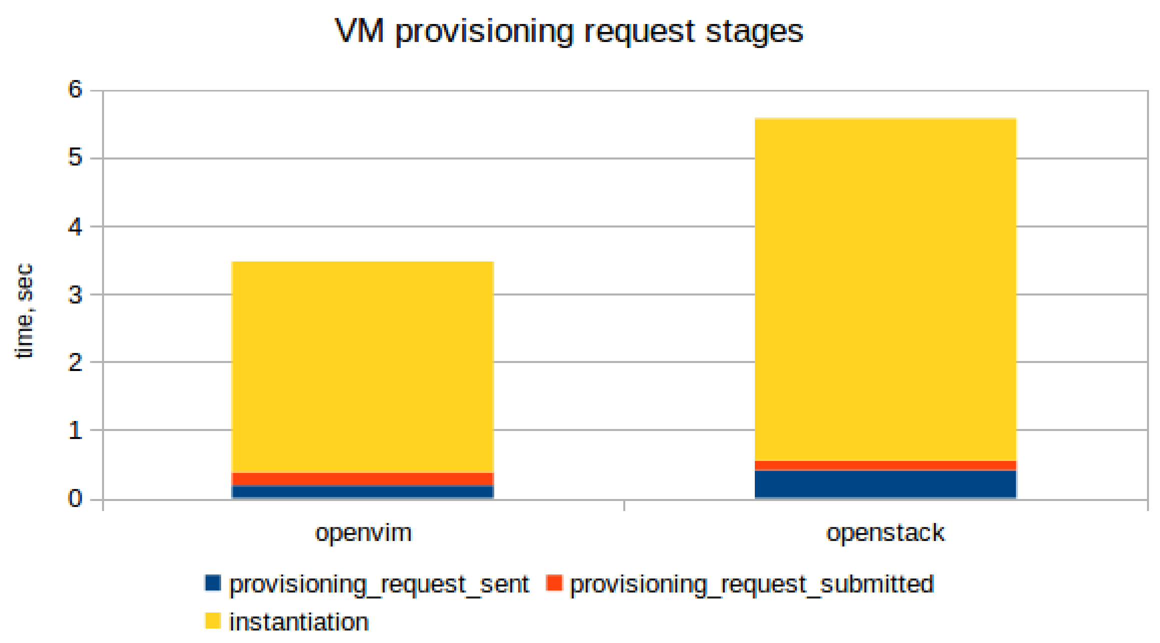

Figure 17: OpenVIM and OpenStack benchmark results

Both proposed solutions can be scaled horizontally by adding additional compute nodes although they lack inherent capabilities for automatic node discovery. The modularity of OpenStack makes the multiplication of the controller node possible with some additional efforts, while OpenVIM does not foresee controller scaling. Both projects are independent from the underlying hardware architecture. However, additional investigation should be done to take into account the heterogeneous resource-constrained devices common for the network edge. OpenStack is capable of running on ARMv8, despite some known issues related to the interaction with the hypervisor and the configuration of devices. OpenVIM has been officially tested only on Xeon E5-based Intel processors and the behaviour on other architectures is not described.

OpenVIM shows better performance that OpenStack and this gives it an advantage in a resource constrained environment such as the edge. However, it lacks support and development activities from the community. Part of its high performance comes at the expense of missing functionalities and authentication and security features. On the other side, although being more complex and general-purpose, OpenStack is massively adopted and updated regularly. Its flexibility allows for the creation of custom solutions for the needs of the edge and makes it the VIM-Edge choice for the project.

3.3.3. Extended Edge

In smart cities, some Use Cases require to provision applications that may need access to sensors/actuators: these class of applications are part of the IoT environment (in this document the word *Extended Edge* is used too). Then these objects can't be treated and deployed like other cloud applications, as they may need access to specific I/O interfaces, network devices or HW accelerators, or they may need to be deployed as close as possible to the device that is the source of the data. In this case, the closest compute node may not be a powerful server but a more constrained device that, for instance, can be placed on a lamppost. Another requirement is that these nodes may have intermittent connectivity, and can be deployed in different steps. This results in the need to discover these new nodes and to deal with low resources in terms of both connectivity and computing.

To address all these requirements the project will leverage on an IaaS software that can harvest computing power from low-end device and expose them to the city infrastructure. There is no need to rely only on an efficient network connection between these small nodes with a fully distributed control plane. All these requirements are addressed using Fog05, which is an infrastructural software with a server-less data-centric architecture that target MEC [13] as well as Fog Computing. Fog05 can manage different types of deployable entities, not only Containers[6] and VMs, but even Native Applications, Unikernels[7] and components of a micro-service framework. It allows also to discover new nodes at runtime and with a very limited configuration as well as discovering the capabilities of these nodes, deploying the application in the node that satisfies the constraints defined in the application definition.

To get all the requirements Fog05 defines a set of abstractions and fully plugin design that allows adding the support for new hypervisors, SDN controllers and, at the end, support for the deployable entity in a fast and easy way. One of the core abstraction of Fog05 is that everything is a resource that can be identified by an URI (Uniform Resource Identifier). Sending an action to a resource is easy as just update some of the fields of that resource. The plugins will implement the state transitions that will derive from an action on a resource. Each resource type has descriptors defined by JSON schemas and are a superset of the one defined by ETSI for VNF and NS: then the descriptors can also be mapped to the TOSCA[8] VNFD/NSD and to the ETSI OSM ones.

The High-level architecture of Fog05 is described in the Figure 18. It is possible to note that Fog05 was designed to run either as a process on a traditional OS or as a trusted service inside a Trusted Execution environment (e.g. a Unikernel like MirageOS[9]). The TEE can run on the top of a hypervisor or on a bare metal (ideally on the trust zone such as those supported by ARM processor or Intel ACRN, [2], hypervisor).

[6] Please see: https://www.docker.com/what-container

[7] Please see: http://unikernel.org/

[8] Details at: http://docs.oasis-open.org/tosca/tosca-nfv/v1.0/tosca-nfv-v1.0.html

[9] Please see: https://mirage.io/

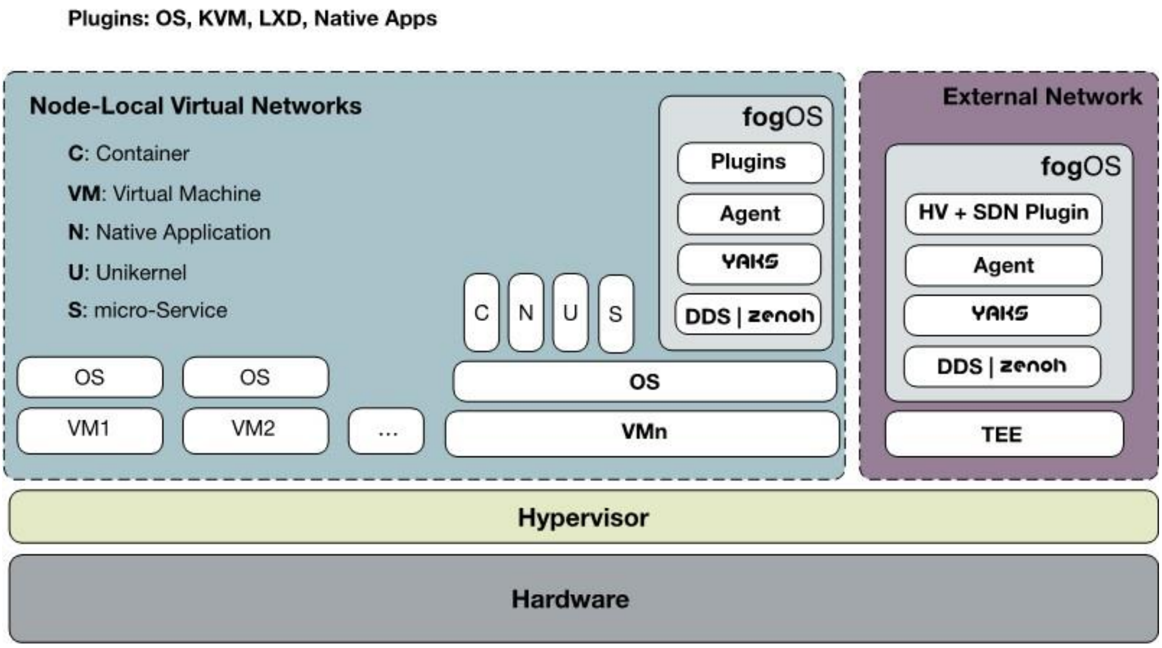

Figure 18: The High-level architecture of Fog05

This deployment allows for all communication with external world through a secure run-time with a very small surface of attack. Internal communication is done using node-wise virtual networks that can cross nodes and will be bridged using a service for the secure part of the system.

Only the software running on the TEE can communicate directly with the external world and will mediates the communication between internal networks and the rest of the world.

The core elements of Fog05 are the agent and the plugins. The agent represents a manageable resource of the system, so there is an agent for each node. Each agent manages the state and the resources, all the dynamic discovery and the distribute control plane, leveraging the YAKS[10] service which is a distributed K, V store that can be implemented on the top of DDS [3] or Zenoh [6] and that takes advantage of their completely distributed and data-centric design. The set of functionalities supported by an agent depends on the loaded plugins, and the agent '*simply'* orchestrated the plugins state transitions.

As mentioned above, Fog05 leverages on the plugins to manage anything from VMs to micro-services components, but even network management (interacting with SDN controllers of the network stack of the underlying hypervisor/operating system). More importantly, the communication with different MANO solutions, in particular in the architecture of 5GCity, Fog05 will interact with OSM which is part of the 5GCity Orchestrator. To allow this interaction in both OSM and Fog05 a plugin was implemented (OSM already uses plugin to interact with OpenStack, AWS, VMWare, etc.) to map an action on OSM to an action on a Fog05 agent. Due to the distributed architecture of Fog05, a service proxy that hides all the complexity of the distributed architecture is in place between OSM and Fog05. This also because OSM and, in general, all MANO stacks are based on a RESTful over HTTP design, but, on the other end, Fog05 uses a data-centric pub/sub communication model through the YAKS services: this proxy will map HTTP actions to YAKS actions and allow the communication within OSM and Fog05.

[10] See http://bit.ly/2xab4EJ "Scale-free Data Sharing" by ADLINK

3.4. SDN controller and agents

5GCity landscape is composed by a wide range of heterogeneous resources, geographically grouped in three main tiers: data-centre resources, edge computing resources, Radio Access resources. Communication among pool of resources are ensured by a network domain, which has been logically divided into two main areas, namely core and edge network, both of them consisting in a mesh of network devices.

The configuration of this network domain has to be done in accordance to the orchestration of the computing resources, being thus able to empower the overall 5GCity infrastructure with full end-to-end slicing capabilities. Network domain has to be enabled with high programmability and flexibility features, together with control plane which is able to provide abstract view on resources from underlying infrastructure.

The solution envisioned in [8] and [9] consists in the utilization of SDN paradigm as a control plane to realize 5GCity data plane configuration in a way which is agnostic to the underlying hardware infrastructure and fully integrated with 5GCity orchestration plane.

A typical SDN framework is composed by the following main functions:

- The SDN controller, as shown in Figure 19 is the core of the SDN Network. It acts as a control point in order to manage flow control to the network devices via southbound interface (SBI), application and business logic via the northbound interface (NBI).
- The NBI of the SDN Controller, in most cases, offers a REST API that aims the interaction between the SDN Controller and the SDN applications providing an abstract view of the underlay network.
- On the southbound, the SDN controller interacts with the network elements, such as switches or routers, SDN agents on the devices, which are in charge of applying to the device specific configuration following control plane indications.

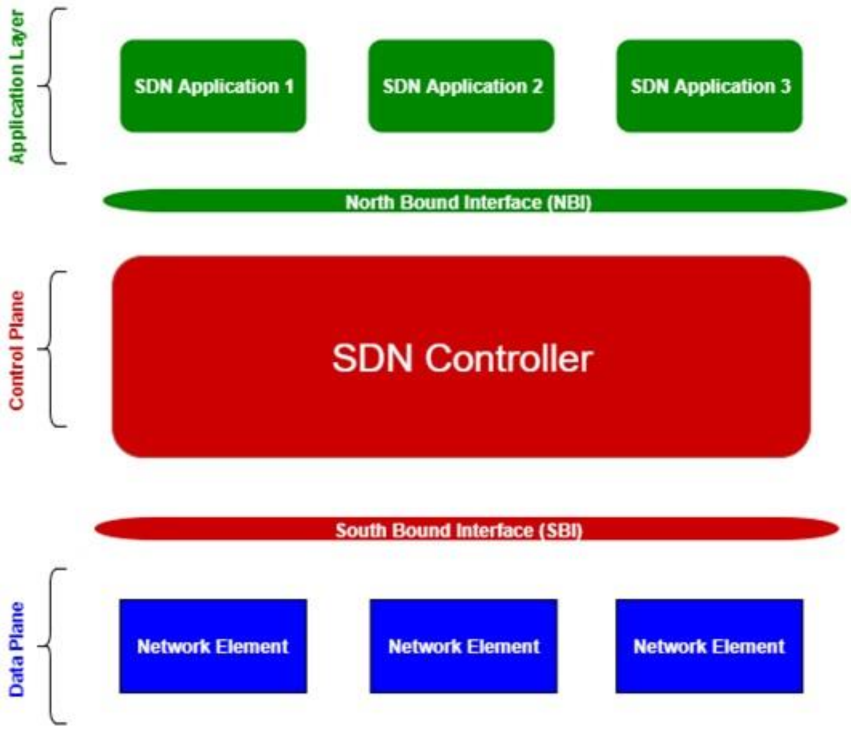

Figure 19: SDN Architecture

In the 5GCity architecture the SDN architecture consists in the following functional elements:

- WAN Resource Manager (SDN Application) is the functional element, which triggers SDN control plane operations. It translates the abstracted view at 5GCity orchestrator level in a more specific, network centric view, ensuring that external link information contained at NFV level is translated in a suitable path which allows proper communication between NFVI PoPs;
- Two different SDN Controllers, first one dedicated to the configuration of the network areas and the second one dedicated to the configuration of the radio access devices;
- A data-plane consisting in Core NFVI, backhaul network, Edge NFVI, fronthaul network, Wi-Fi Access Points and LTE small cells are the network elements and are considered as part of the infrastructure layer.

The interaction between elements in the 5GCity, as shown in the Figure 20, is the following. On the northbound interface, the SDN Controllers interact with the WAN Resource Manager (which acts as SDN application) while on the southbound the SDN Controllers interact with all the components of the infrastructure layer.

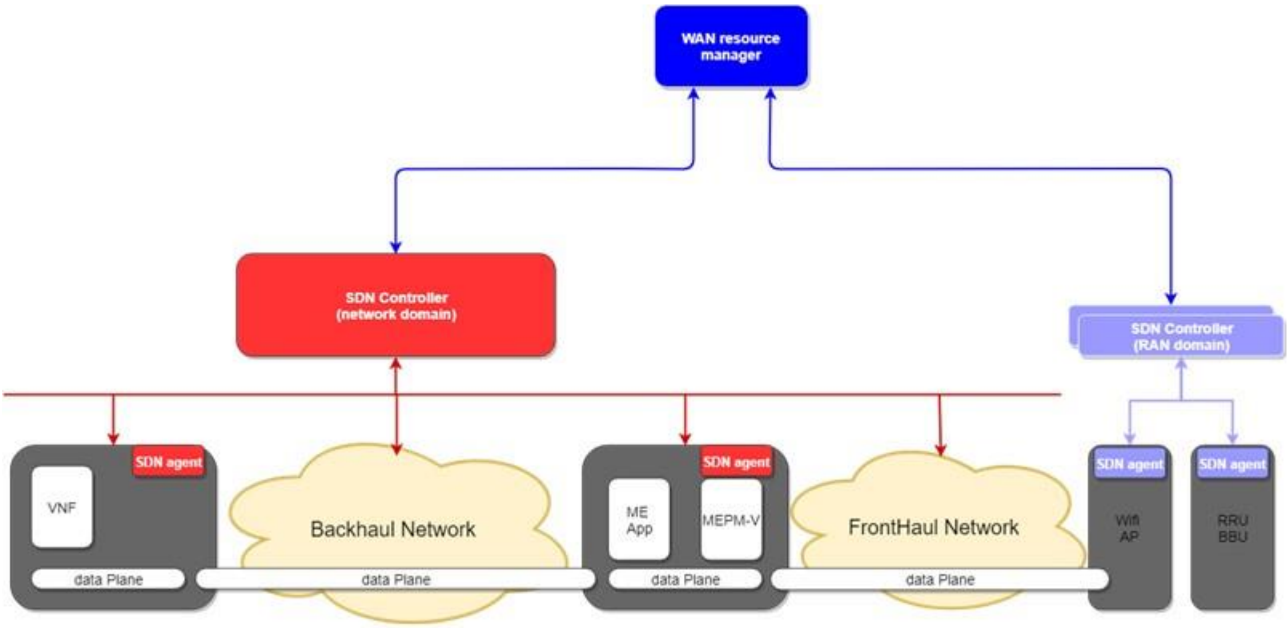

Figure 20: SDN Controllers on 5GCity architecture

In order to handle different type of configurations, two different SDN controllers are used for this purpose.

The first SDN Controller (red box) is used for the configuration of the infrastructure layer. It is in charge of the configuration of the Core and Edge NFVI, through SDN agents running locally on this infrastructure layer and the transport network: configuration of the backhaul network between Core and Edge NFVIs and the configuration of the fronthaul network between Edge NFVI and RAN layer.

In order to perform the configurations, the SDN Controller receives from the WAN Resource Manager a list of parameters:

- BH ingress port;
- BH egress port;
- FH ingress port;
- FH egress port;
- bandwidth ;

- delay;
- encapsulation type;
- tunnel ID.

These parameters are used by the provisioning manager, embedded in SDN controller, to calculate the paths according to certain algorithm (e.g. shortest path) to configure the transport network in both fronthaul e backhaul cases. The configuration of the infrastructure layer is performed in two steps as described in the Figure 21.

In the first step (blue arrows), the SDN controller performs a 'internal' configuration of the Core and Edge NFVI in order to allow traffic from the NFVI Pop to the ingress and egress ports (respectively in the Core NFVI and Edge NFVI in case of the backhaul network). This step, anyway, is not mandatory, since there are deployment scenarios where the SDN Controller is not controlling the NFVI PoP internal network configuration and the network connectivity of the NFVI node is delegated to other VIM entities (i.e. Neutron, in case of the VIM is identified with OpenStack Controller).

The second step (red arrows) consists in the configuration of the backhaul network, which consists in the configuration of the various switches (SW1, SW3 and SW4 in the Figure 21) to allow traffic from the ingress port to the egress port of the backhaul network got from the WAN Resource Manager.

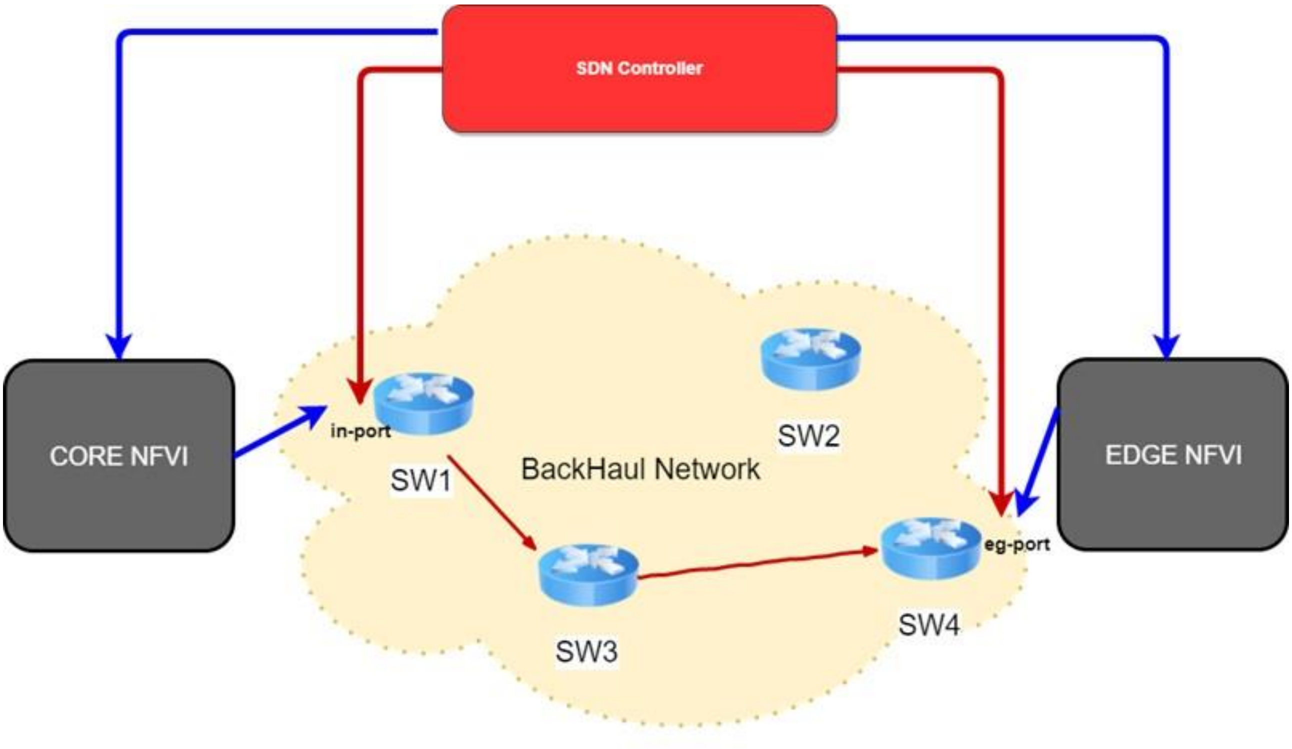

Figure 21: Configuration steps

The WAN manager acting as a SDN application is also in charge of implementing the logic, which triggers and controls this two-step configuration of the networking devices.

The same configuration two-steps strategy is used for the configuration of the fronthaul network.

3.4.1. SDN agents Radio Wi-Fi AP

In order for the Wi-Fi APs to be controllable via the RAN SDN controller, each node runs SDN agents based on OpenvSwitch (OvS) and using the OpenFlow (OF) protocol. Further, the Wi-Fi APs run a NETCONF server used to set up, configure and manage virtual interfaces upon request from the SDN controller (which runs a NETCONF client for this purpose). While the use of NETCONF to configure the RAN interfaces might not be considered in a classic SDN-based approach, the use of this protocol is essential to enable the network slicing in 5GCITY (the details on how NETCONF is used are discussed in [5]).

At bootstrapping, a default set of virtual switches is configured on each Wi-Fi AP to connect with the RAN SDN controller over the wired network infrastructure. Once the virtual switch is hooked up to the control plane, the RAN SDN controller can see the virtual switches representing the elements to which virtual APs can later be hooked up. The virtual wireless interfaces instantiated for a tenant when requesting a slice via NETCONF and whenever RAN presence is requested for a particular Wi-Fi AP, are added to the control plane by hooking them up with the OvS running in the Wi-Fi APs. Figure 22 shows the Yang model used for the Wi-Fi APs (*i2cat-box*), including all configuration parameters.

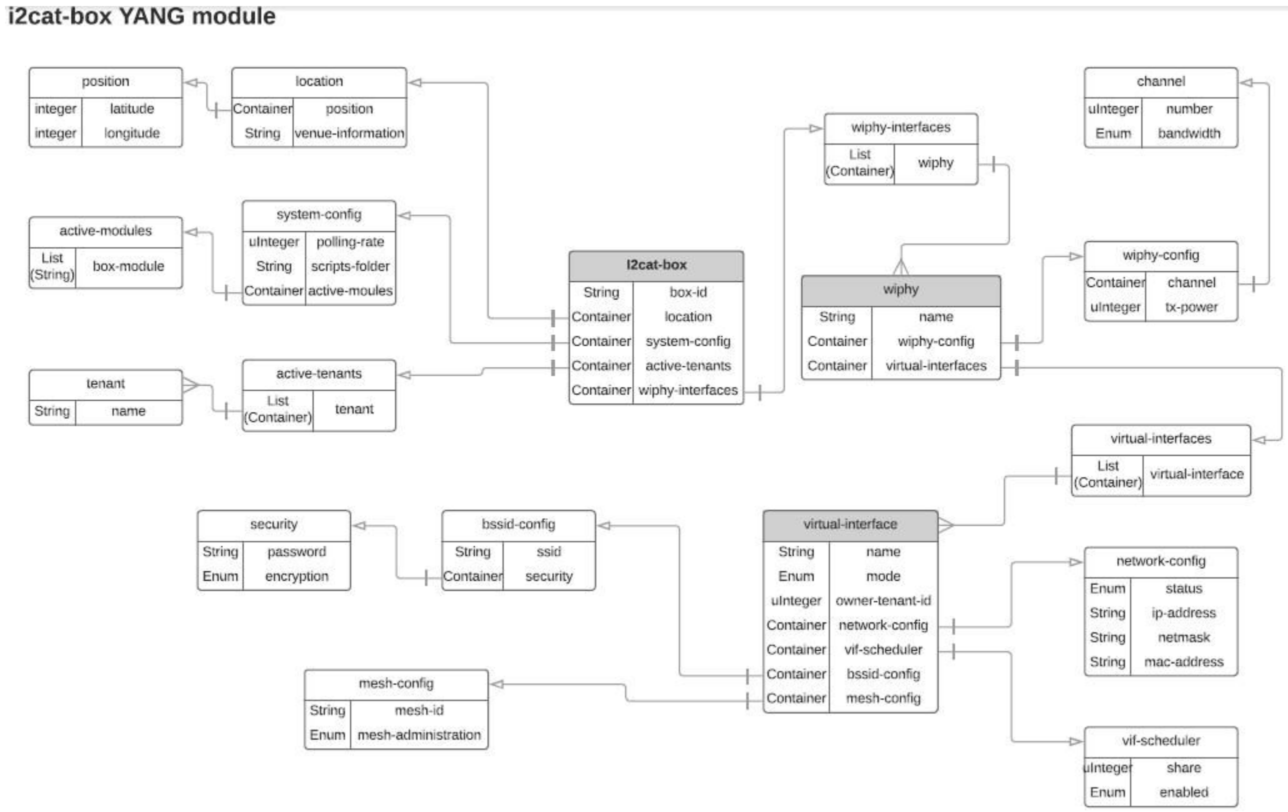

Figure 22: Wi-Fi node Yang control interface model

After a virtual AP has been generated and configured via NETCONF, the SDN controller assigns the virtual AP to a VLAN, which is the basic required to enable the SDN-based slicing for the neutral hosting in 5GCity.

3.4.2. SDN agent LTE SC

The SDN Agent for the LTE Small Cells supports NETCONF based control of network slicing. The agent is implemented using Netopeer and provides an SDN control interface supporting the following functions:

- Initialisation
- Slice Profile Creation
- Slice Profile Modification
- Slice Profile Deletion

SDN control is implemented using the NETCONF protocol via a data-model, which is formally defined in YANG. The Figure 23 illustrates the RAN orchestration model as implemented in NETCONF/YANG.

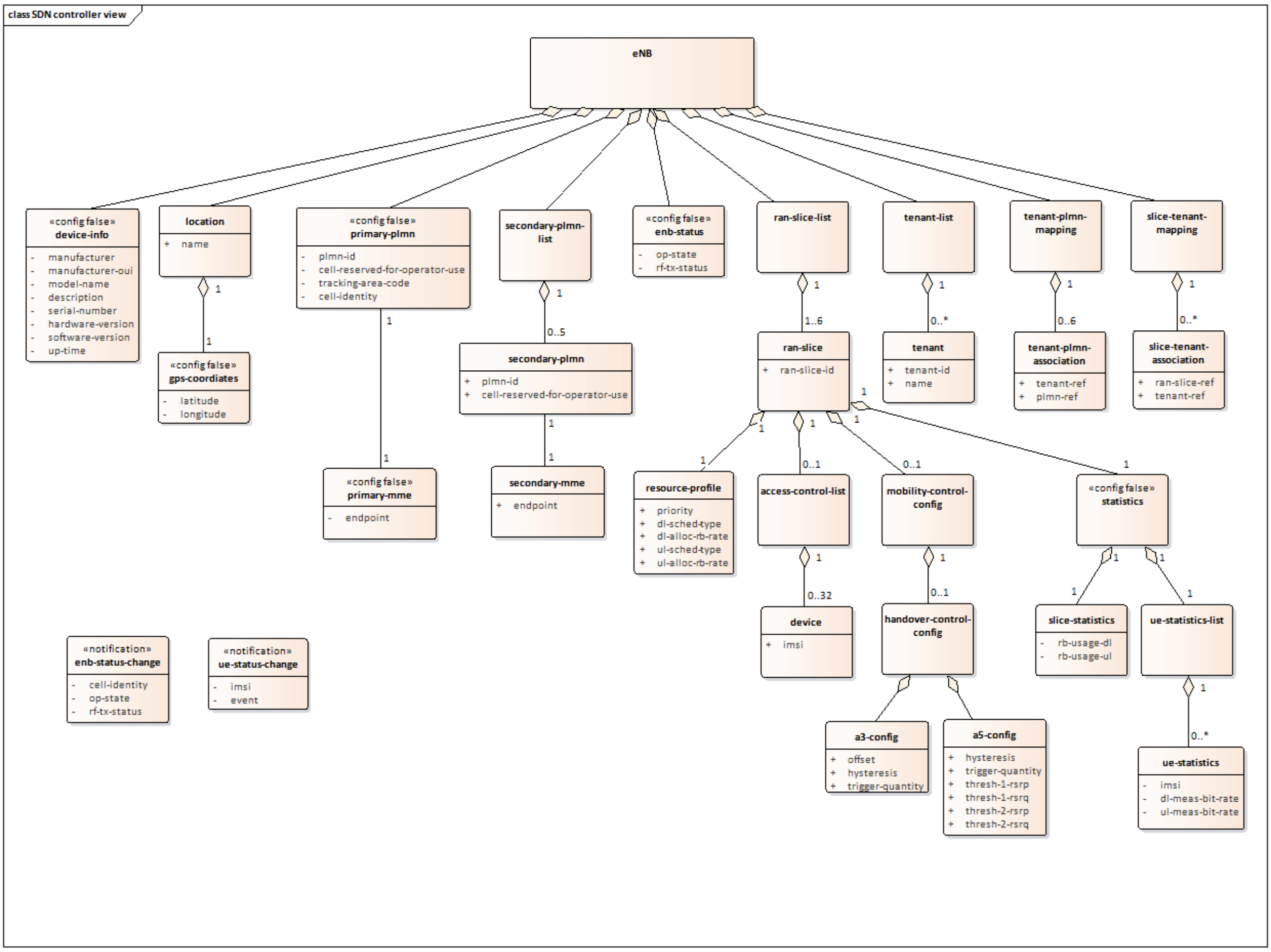

Figure 23: RAN SDN Control Interface Model

The data model has been aligned with the approach taken for Wi-Fi orchestration. Each slice corresponds to a 3GPP PLMN_Id (network identifier) and there can be up to six slices created. Additionally, per slice, an access control list can be instantiated to control which specific users (IMSI identifiers) are permitted access to the slice. Each RAN slice is associated with an MME and EPC via a standard S1 connection. This EPC can be either at Edge or Data Center level.

3.5. Monitoring Framework

The monitoring system for the 5GCity platform must monitor the overall virtualized resources (calculation, storage and network) of the three-tier architecture and, through an appropriate set of parameters, the applications and services running on the 5GCity infrastructure.

Regarding the infrastructure, the monitoring system includes three different resource domains, i.e. 1) NFVI resources; 2) SDN-enabled elements; 3) physical devices that do not belong to the first two categories.

With regard to the applications and services, the monitoring system will include VNFs and service monitoring parameters (metrics, useful also to check SLA compliances).

The monitoring system has to be integrated with the different orchestration layer components, as the Resource Manager, the Slice Manager, the SLA Manager and the OSS/BSS systems to assist in the network system management. It turns out that the 5GCity system monitoring will be able to provide capabilities to monitor both network and cloud infrastructural elements and the related services with a full end-to-end view. In order to have this feature, the monitoring system shall have a global perspective, with a view to the different services composing the overall infrastructure and shall output collected data providing a unique and simple to access view of the system.

4. Infrastructure Layer

4.1. Core NFVI (data center)

5G introduces a series of important innovations that make complete the transition from a "network of equipment and functional entities" to a "network of functions (virtual)", already started with the introduction of the principles of NFV (Network Function Virtualization) and SDN (Software Define Networking), defined by the ETSI (European Standard Telecommunications Institute). The 5G NG Core is based on telecommunication cloud solution, with proven NFV technologies, network slicing and network function virtualized infrastructure (NFVI) platform. In 5GCity architecture cloud and software-defined infrastructure are integrated to offer support for network slicing and automated orchestration. Besides, using this approach is possible to deliver virtual network functions and services in shorter time. At Data Center level, resources for VNFs are dynamically allocated according to the capabilities such as redundancy, availability, throughput, latency, trusted tenant isolation, automatic VM recovery as requested.

5GCity uses Network Function Virtualization (NFV) as a key concept in its architecture design. In particular:

1. The NVF infrastructure (NFVI), which includes physical resources (primarily including computing, storage and networking equipment) and specifies how these can be virtualized.
2. A series of Virtual Network Functions (VNFs), which are software implementations of network functions. Their execution is supported by the NFVI, i.e. they can run over the NFVI.
3. The NFV Management and Orchestration (NFV MANO), which covers the orchestration and lifecycle management of physical and software resources that support the infrastructure virtualization. It also covers the lifecycle management of the VNFs. It is responsible for all virtualization-specific management tasks required by the NFV framework.

The main advantages of this architectural solution are listed below:

- Creation of independent user profiles on a standalone DB
- Separation of mobility and session management functions
- Independence of the CN design from the access network (access agnostic)
- Single reference point for access to the NC
- Authentication functionality integrated into the solution and independent
- Application function integrated into the solution and independent
- Complete scalability in the management of high efficiency control functions
- Standard user traffic management architecture.

In particular we focus our attention on the aspects of the core NFVI at data center level starting from the definition of the network slice: when we consider the concept of "network slice" we mean a complete logical network that encompasses a series of functions (and corresponding resources) necessary to obtain certain network characteristics/capacities. This concept can include both the access component and the core component. Numerous network slices can be developed with the same characteristics but usable by different groups of users according to the specific needs. These capabilities are managed at data center level.

In the DC the NFV and the VIM will be deployed to orchestrate and manage the service and resource pool. In particular, the Network slicing, a complete logical network that comprises of a set of network functions and corresponding resources necessary to provide certain network capabilities and network characteristics: the design, installation, termination sharing and monitoring. Then at DC level we have also to deal with the Network Function discovery and selection and with the Mobility and session management.

It is important to underline that also at DC level the QoS and the Policy managements are included: as indicated in the 5G QoS model, 5GCity supports both QoS flows that require guaranteed flow bit rate and QoS flows that do not require guaranteed flow bit rate.
According to the 5G services classes it could be possible at DC level define the correct parameters for each classes: eMBB slicing, uRLLC slicing, mMTC slicing.

The main DC will be able to execute different Service VNFs hosting the controller which performs decisions for cope with the heterogeneous access available technologies, like 5G, LTE, Wi-Fi, as requested by the service-driven architecture proposed by 5G paradigm (Figure 24).

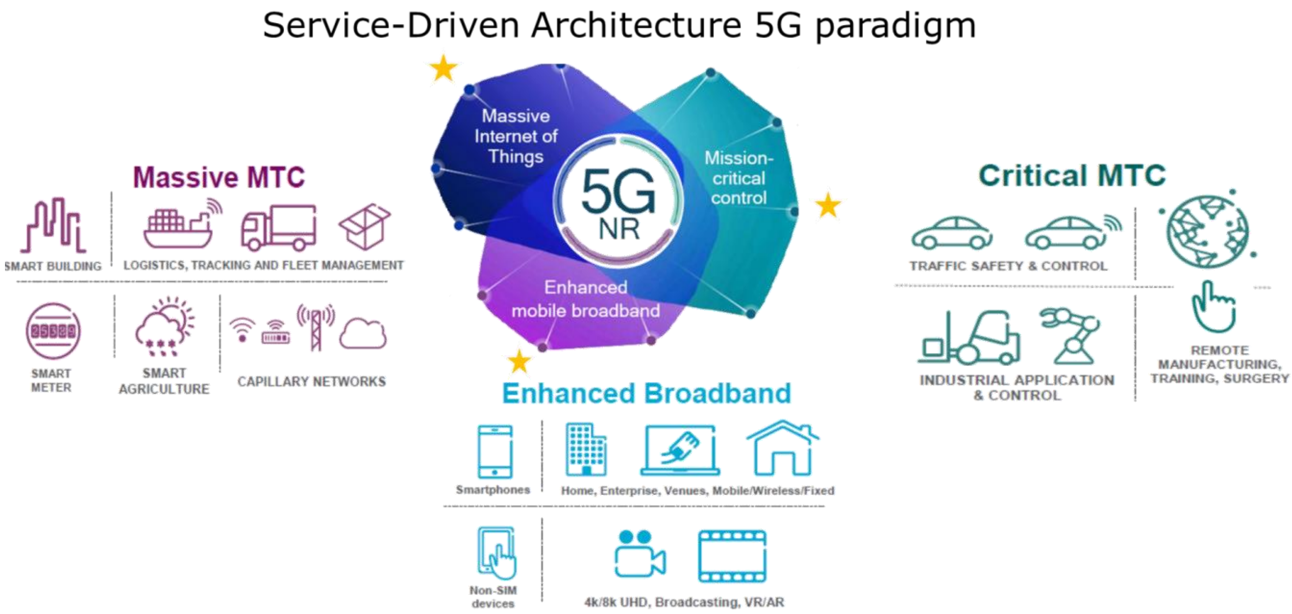

Figure 24: Service-Driven Architecture 5G paradigm

4.2. Backhaul Network

In this section, we are going to describe the relevance of the Backhaul scenarios in the 5GCity architectural design, for 5G networks.
The collecting data networks have been designated as the Backhaul. There is a huge amount of data that needs to be carried from the data centers to the Cloud or the Operator CPD and this makes the backhaul to be carefully dimensioned. Commonly this is an IP/MPLS network and the expected data rates for 5G systems are not less than 10Gbps.
That part will allow the virtualization layer to manage the network resources like switches and routers to support the data transportation of the deployed services and network functions.
The 5GCity project has a clear market orientation focusing in a technical development of great value and supporting a series of relevant reference scenarios within the Neutral Host Use Case.

Scenario 1. Neutral Host with own network

Service providers provide the needed infrastructure and networks to offer their own services or their infrastructure to other operators. These operators are characterized by:

- Having own networks to serve services
- Having enough purchase capabilities to acquire and maintain the telecommunications systems
- Following new business models in front of the margin reduction in the traditional services
- Desiring the use of these infrastructures to hold new services and thus incorporating new technologies like NFV and SDN.

As a result, it will complement its capabilities allowing it to orchestrate the allocation of resources and services to the operators in an agile, dynamic and efficient.

Scenario 2. Virtual Mobile Operator

Smallest operators often have many difficulties in offering value-added services. On the one hand, its services replicate those of the largest operators, whose networks they depend on. On the other hand, the large costs they incurred make innovation and differentiation very difficult. The results of this project will make the service component of virtual operators being more independent and productive while reducing their associated operational costs, so that they can provide their own added value services.

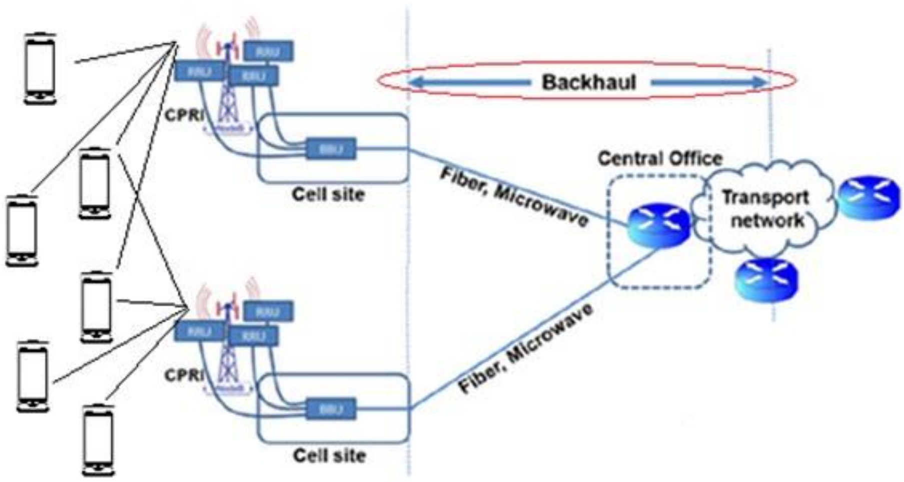

Figure 25: Backhaul network

In 5G scenarios, the existence of multiple virtualized network functions and network slicing mechanisms makes the network to be able to support all these functionalities alongside the delivery of such a huge amount of content. The use of all these functions will allow the consumption of a higher bandwidth and in consequence a higher data rate. All this capacity coming from the access networks must be managed by the backhaul - see Figure 25 - and with the minimum time, it must be delivered in one way to the end users and in the other way to the data center, in order to be processed.

There is a set of key requirements for any network wanting to meet 5G service demands. This network will have to be reliable, operational and efficient for the following capabilities:

- More capacity per device (ultra-high capacity per end-device)
- Mew types of devices (introduction of IoT and M2M services)
- More devices (exponential growth)
- New services (Augmented and virtual reality, autonomous car, etc.)

The case that 5G is slicing the network and by function virtualization is adapting the bandwidth of each small cell leads to a very high data consumption as the spare bandwidth can be harnessed by the end users who need higher consume while current technologies (3G and 4G) does not harnesses this spare bandwidth.

In the case of the 5GCity project and the Barcelona scenario, see Figure 26, the Barcelona consortium is planning to deploy two links network with a capacity between 10Gbps and 1Gbps in the lowest link. With estimations to have in the near future a whole 10Gbps capacity network.

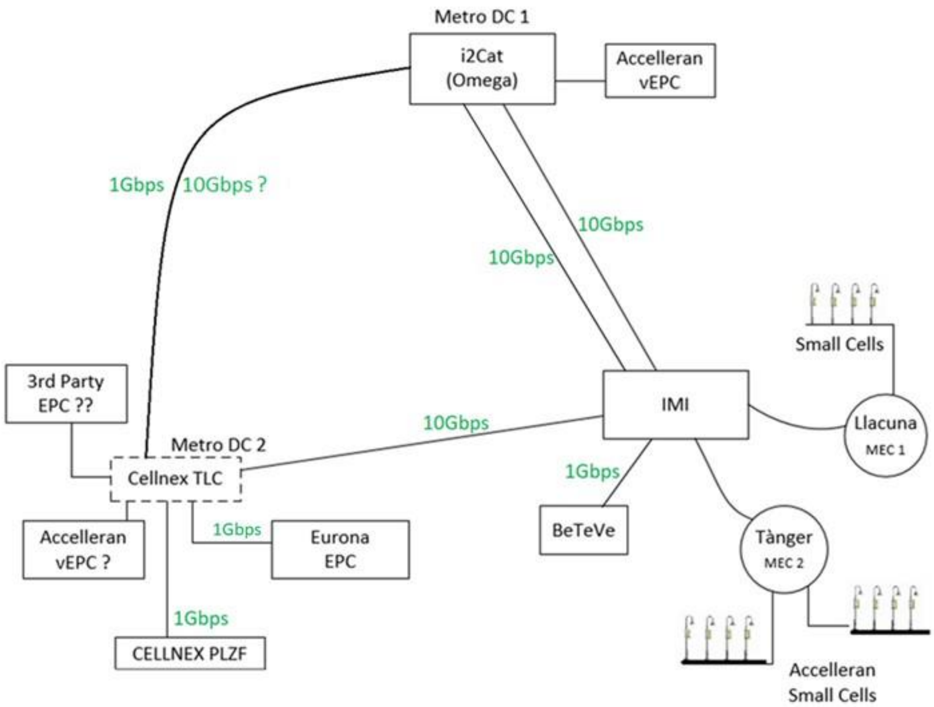

Figure 26: 5GCity Barcelona scenario

4.3. Edge NFVI

The Edge NFVI supports the middle tier of the three-tier 5GCity architecture and provides distributed compute and storage resources locally at the network edge (close to the access node) to support NFV instances which implement components of network slice service chains (especially at RAN level) and to provide an environment for Multi Access Edge Computing (MEC) applications. The following sections describe the edge networking architecture (itself implemented through NFV deployments) and the NFVI architecture for orchestrating and executing both networking and MEC application VNFs.

4.3.1. 3GPP RAN Network Slicing and Virtualized Edge Architecture

The RAN architecture for 5GCity (Figure 27) includes several innovations in the area of RAN functional disaggregation, RAN and Network Slicing with SDN control and RAN function virtualization.

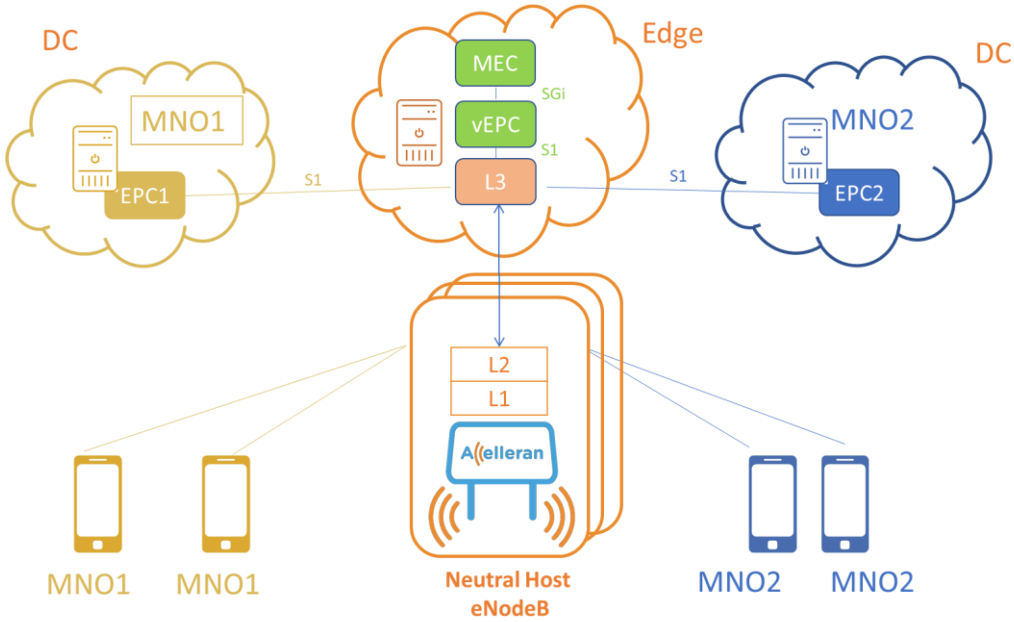

Figure 27: 5GCity RAN Network architecture overview

Network functions in the architecture are:

- LTE Layer 1: The physical layer functions of the LTE air interface require specialised processing acceleration for DSP functions such as FFT, Turbo coding, etc. and execute in each radio head on specialised DSP silicon. This is not a virtualised function in 5GCity and is common to all network slices.
- LTE Layer 2: The RLC and MAC functions of the LTE air interface are required to meet real-time schedules and are closely coupled to the LTE physical layer 1 implementation. These functions also execute in each radio head. RLC/MAC are not virtualised functions in 5GCity and are common to all network slices.
- LTE Layer 3: The layer 3 (control plane) function of the LTE air interface is implemented as a virtual network function that runs in the Edge NFVI. 5GCity L3 supports network slicing and connection to multiple EPC (MME) instances (one per slice).
- vEPC: EPC (packet core network) is deployed at the network edge to support MEC access and low latency applications. Each instance of vEPC supports a network slice offering MEC application access.
- Data center EPC: The RAN functions support connectivity to EPC functions implemented at the data center. The Neutral Host use case assumes multiple EPCs and network slices to support access provision to multiple tenant service providers.

4.3.2. Mobility Management at the Edge

Mobility between cells at the edge is managed in accordance with standard 3GPPP procedures and interface. There are two cases to consider:

- Mobility between cells connected to the same edge node and vEPC (Figure 28): In this case mobility is handled within the RAN function with handover between the local eNBs and MEC connectivity maintained via the local vEPC

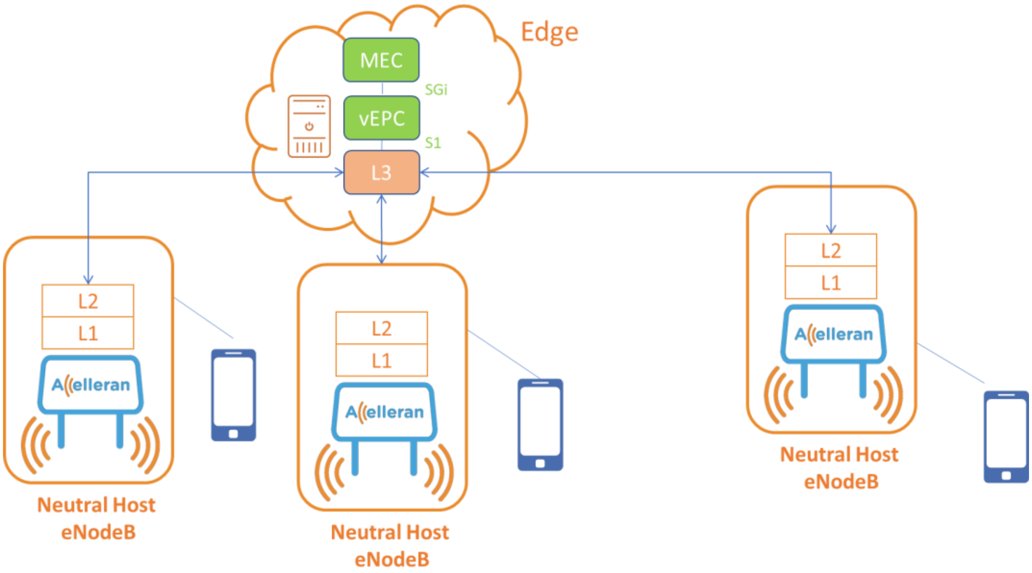

Figure 28: Intra vEPC mobility

- Mobility between cells connected to different edge nodes (Figure 29): In this case, mobility involves inter-EPC procedures with service continuity to the MEC application maintained. It is for the MEC application to manage service migration between edge nodes should this be desired.

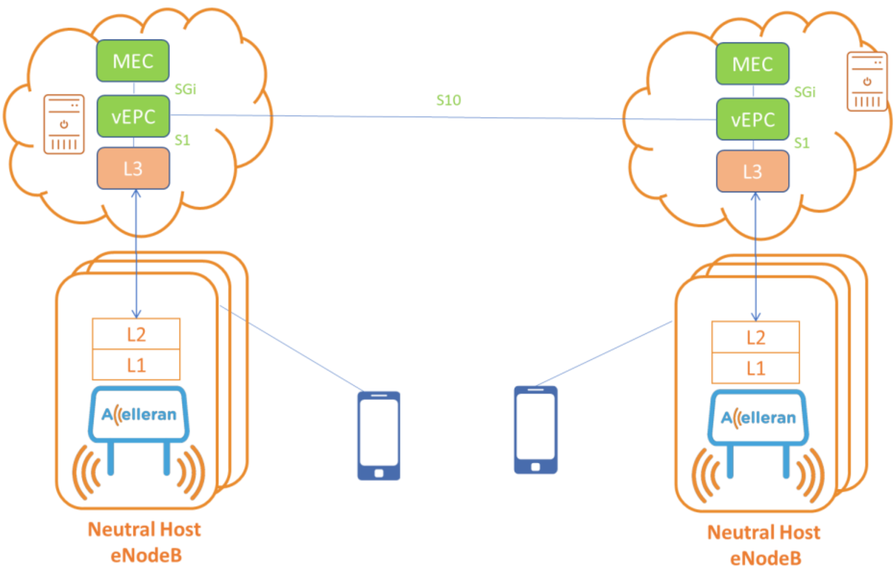

Figure 29: Inter vEPC mobility

4.3.3.Management and Orchestration of RAN VNFs

About Management and Orchestration of RAN VNFs, in the following are shortly described the eNB (FCAPS and SDN) and vEPC features.

- eNB FCAPS: RAN layer 3 is deployed as a Docker container. General RAN FCAPS management is supported via multiple interfaces including BBF TR-069 [14], webGUI and CLI. One of these mechanisms is required to initially configure each cell for service in the network.
- eNB SDN control: RAN L3 provides an SDN control interface supporting the following functions
    - Initialisation
    - Slice Profile Creation
    - Slice Profile Modification
    - Slice Profile Deletion

  SDN control is implemented using the NETCONF protocol via a data-model which is formally defined in YANG. The Figure 23 (reported in Figure 30) illustrates the RAN orchestration model as implemented in NETCONF/YANG.

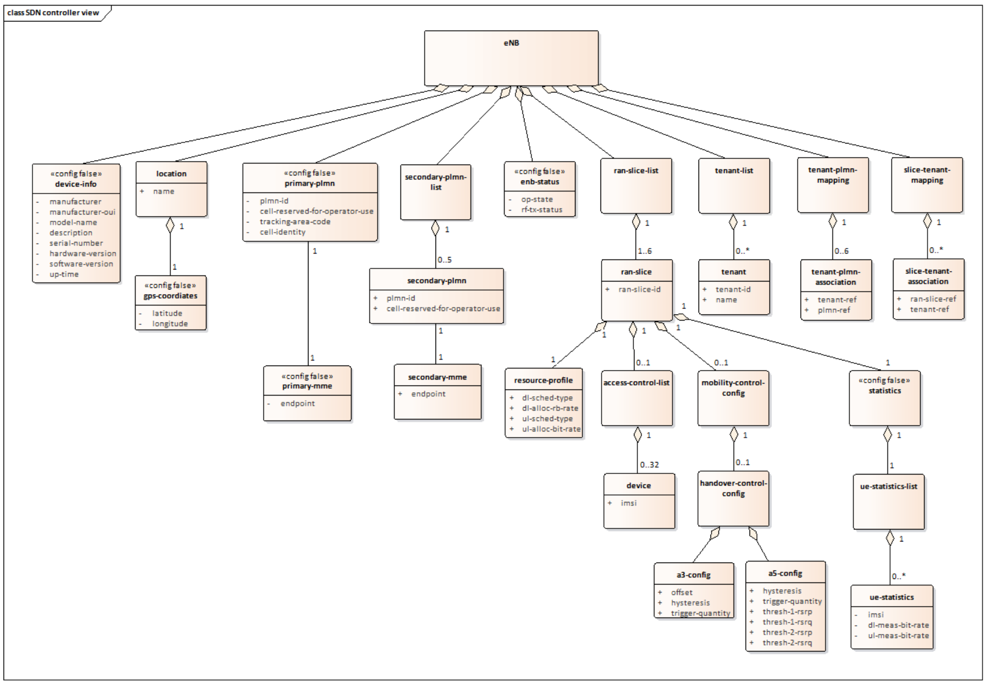

Figure 30: RAN SDN Control Interface Model

- vEPC: The vEPC for edge deployment provides WebGUI and/or CLI based configuration interfaces.

4.4. Front-haul Network

In this paragraph, we describe the relevance of the front-haul components in the 5GCity architectural design, for cellular networks and small cells combining the potentialities of both.

From a private and neutral perspective, an infrastructure operator has different options to tackle the network deployment. Taking into account the business level the infrastructure operator would like to achieve, it should decide between different deployments (among several options) that would lead to different business. From this point of view, the equipment can be moved from the tower-cell to the cloud (data center) presenting different scenarios as it is moving along the route:

- In the beginning the equipment is placed in the tower-cell or in the cell-site cabinet where the base band unit (BBU) and the Remote Radio Unit (RRU) are both in the tower. There is no impact in the front-haul as it is not used in that scenario, all the needed infrastructure is deployed in the tower. Now, to get the benefits from the distributed architecture (C-RAN and V-RAN), the RRU and BBU will be split and moved from the tower to the data center.
- C-RAN The equipment is divided between the tower-cell and the data center. The RRU remains in the tower-cell while the BBU is moved to a data center (Metro DC). In that scenario the front-haul takes relevance as different tower-cells can be connected to the same baseband unit. The back-haul is not far from the tower-cell and this could be an advantage for the data transportation.
- V-RAN scenario. The physical architecture is the same than for C-RAN but the software is virtualized. The new computational techniques such as SDN and VFN makes it possible transforming the traditional two-tier front-haul architecture which uses a point-to-point connection between the BBU and the RRU to a three-tier multipoint-to-multipoint network that is empowered by software.
- MEC for particular scenarios. The BBU is moved nearer to the cloud and the front-haul gets the most relevance as a single MEC can manage several tower-cells providing content and all the managing information. The back-haul is far from the tower-cell and data traffic is supported by the front-haul.

We can consider the 4G LTE (Figure 31) protocol architecture as the “reference” to include the small cells functions in the deployment context as VNFs (vBBU), and extend the model to also include small cell capabilities in our architecture.

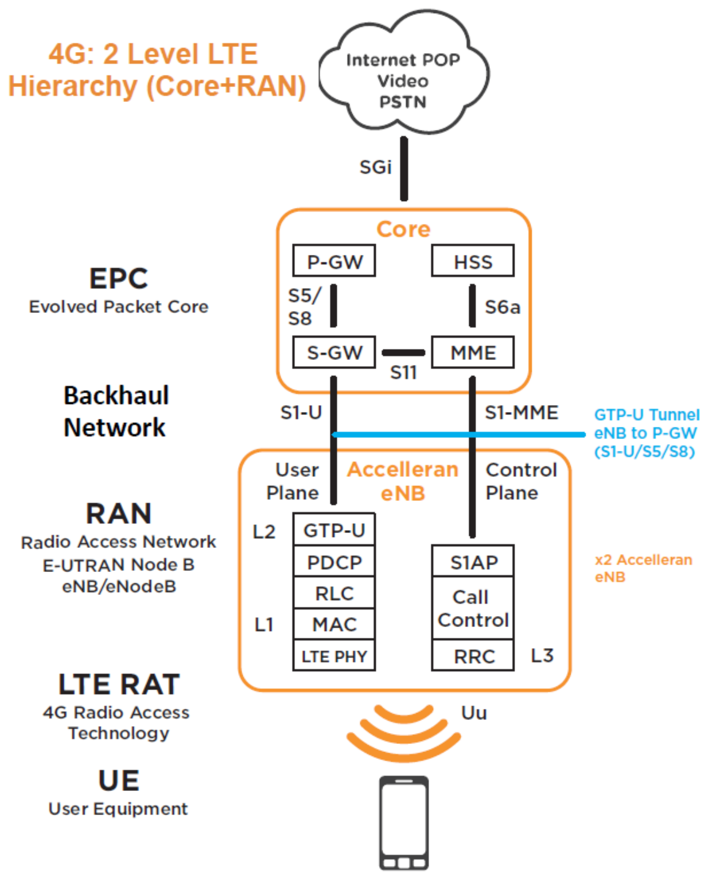

Figure 31: 4G - 2 Level LTE Hierarchy (Core+RAN)

Starting from the two levels LTE hierarchy model including Core and RAN in 5GCity architecture we have to consider a three levels schema (Figure 32).

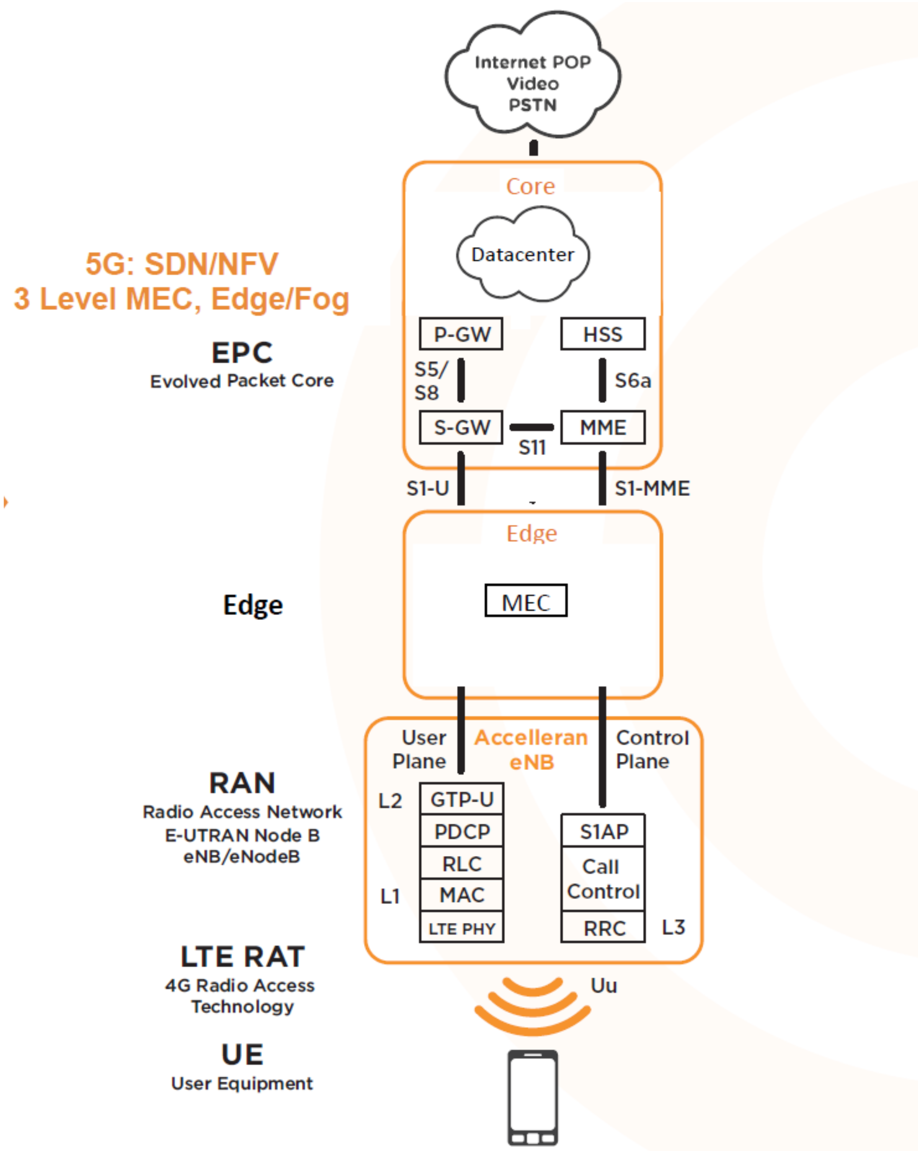

Figure 32: 5G - SDN/NFV 3 Level MEC, Edge/Fog

According to this approach, some capabilities will be realized as virtualized functions while the additional capacity will be made available for running mobile service instances in the edge area. The idea is to endorse the deployment of small cells with some virtualized functions, the inclusion of mobile edge computing will improve the quality of the service, in particular the end users' experience.

Radio resource management control has to deal with the scheduler, the congestion control and other mechanisms. Different RAN clusters have to be considered.
In particular, in case of the LTE systems, the scheduler allocates Physical Resource Blocks based on the traffic type according to the QoS E2E requested; in the new architecture of 5GCity it is requested that the solution will provide an effective radio resource management for all the virtualised RANs components with the

capacity to virtualise various components. The virtualisation of all or portions of baseband processing of wireless stack can be centralised to enable the pooling of radio resources (Figure 33) minimizing the provisioning of the different requested resources according to the real requests of bandwidth.

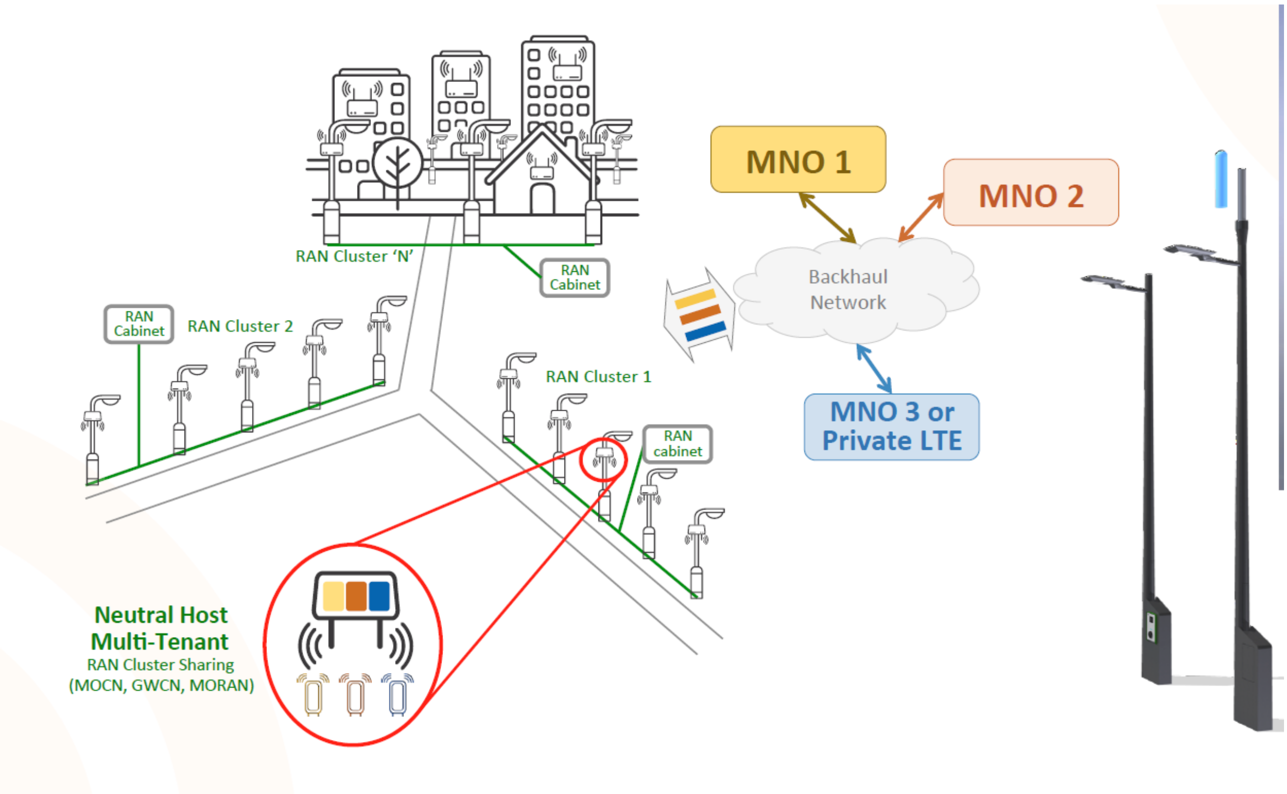

Figure 33: Multi-Tenant view in the Neutral Host

As far as concern the small cells front-haul, in the general schema we consider the opportunity to have remote small cells, where functions are non-virtualized and a central small cell, where functions are virtualized and are executed on a pool of shared computation, storage and networking resources. A central small cell will serve multiple remote small cells: the central and remote small cells are physically connected through the front-haul link. The bandwidth will be dimensioned according to the requirement of the deployed services in the area.

For the 5GCity architecture the model of the vRAN is suggested according to the schema in Figure 34:

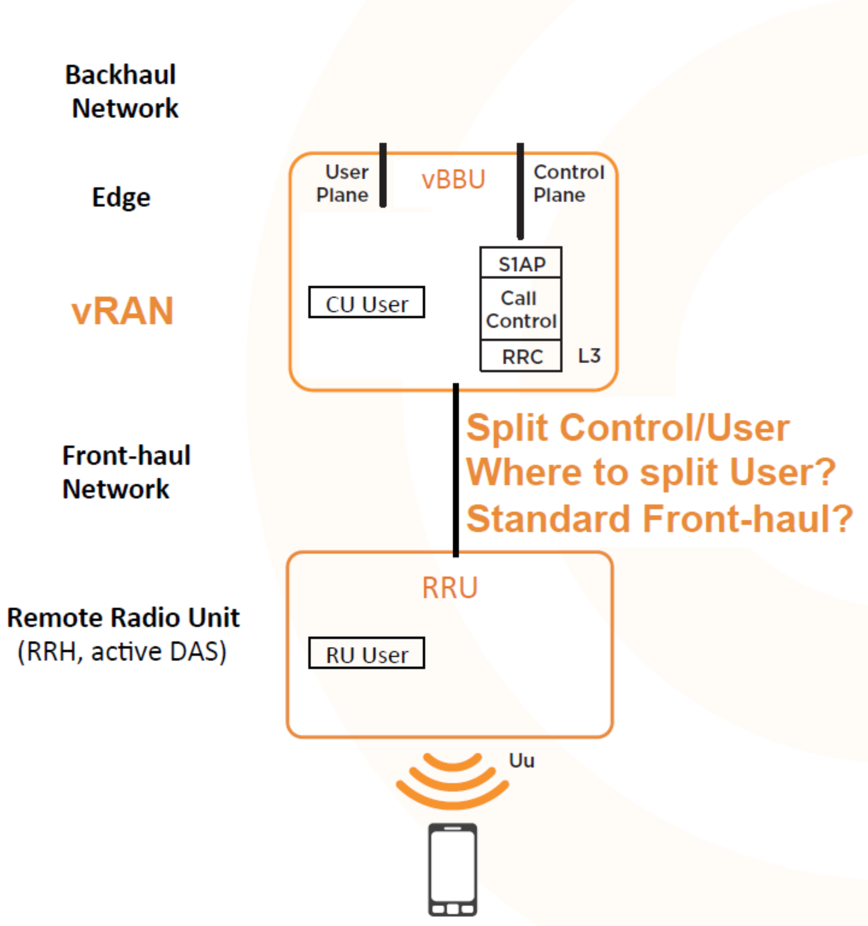

Figure 34: 5GCity vRAN Model

The front-haul standard level (see Figure 35) has to be improved to be able to allow the integration of the different radio access technologies in the neutral host context

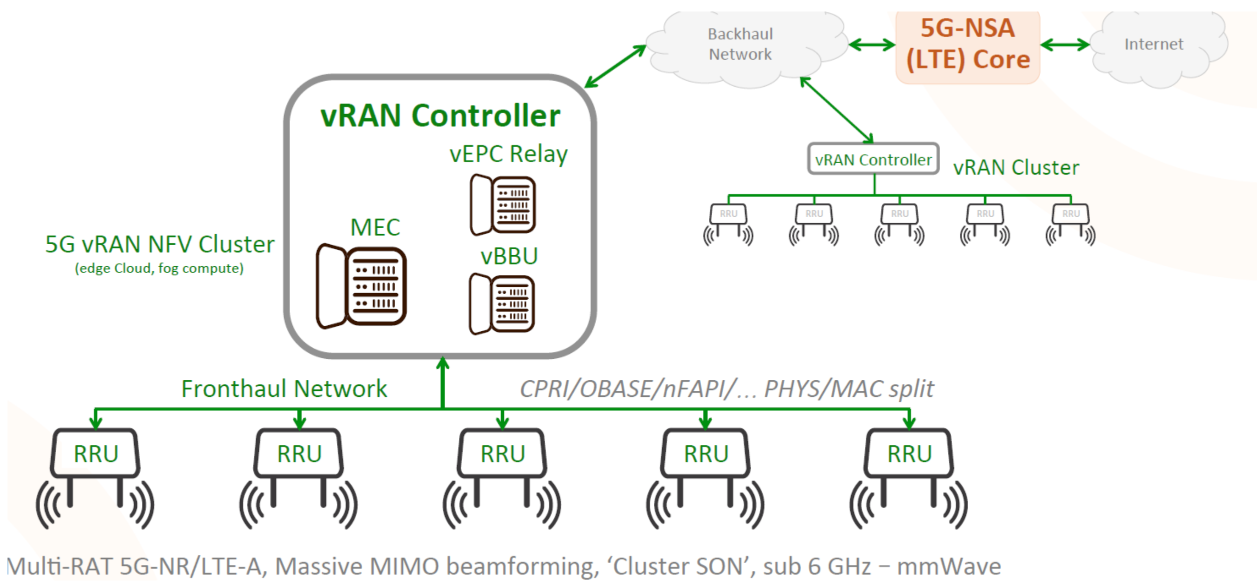

Figure 35: 5GCity front-haul level

In general, from the architectural point of view it will be relevant to describe the front-haul resources.

For LTE

- #Transceivers
- Per transceiver
  - Band (3GPP Band number, i.e. B3, B7, B42, etc.)
  - Bandwidth (5, 10, 15, 20 MHz)
  - Max. Output Power Capability

For Wi-Fi

- #Transceivers
- Per transceiver
  - Band (2.4, 5GHz)
  - Standard (n, ac, ...)
  - Max. Output Power Capability

For small cells

- #Transceivers
- Per transceiver
  - Band (2.6 GHz,)
  - Max. Output Power Capability

Allowing the centralised resource control, it is possible to guarantee not only the target 5G performance KPIs but also to reduce costs (both OPEX and CAPEX).

4.5. Extended Edge NFVI

There is no support for NFVI on the small cell platform planned, due to the compute resource constraints on the platform.

4.6. Radio Element (for Small Cells and/or Wi-Fi)

In 5GCity, two types of radio elements are integrated into the architecture: small cells (SCs) and Wi-Fi APs. On these radio elements tenants can ask for presence when setting up their network slices, whenever wireless connectivity is required in a particular area of the city. Presence in the RAN can be implemented by dedicated PLMN-IDs in the case of SCs and by virtual APs in the case of Wi-Fi. Since these instances are SDN-enabled, they can be integrated dynamically with other elements of the architecture, such as VNFs and the wired back-haul, so an architecture-wide slice can be generated.

Apart from requiring presence, a tenant may also be interested in offering a certain QoS to the users connecting over the wireless medium. Such requirements are translated into assigning the virtual presence of a tenant in a SC or Wi-Fi dedicated radio resources. This can be a dedicated carrier (in the case of a SC) or airtime, e.g. a minimum share of time during which a tenant can actively use the radio channel.

In this context, NFVI support for small cells is not planned. This does not impact the functionality and the potentiality of 5GCity, as the industry trend towards virtual RAN will migrate more and more functionality from the small cells (RAN) to be executable in the edge server environment with restricted functionality in the embedded small cell (RRU).

4.6.1. Small Cell Radio Element

The small cell radio element provided is the E1000 outdoor small cell (see Figure 36 and Figure 37). This is a compact, single carrier LTE eNB supporting 2x2 MIMO and up to 64 simultaneously connected UE's. The unit is weatherproof (IP67) and designed for operation in a range of environments.

Back-haul connectivity is provided by a single Gigabit Ethernet physical connection, which also provides power to the unit (PoE+) that is a single cable providing both connectivity and power.

Antennas are fitted externally allowing the type of antenna to be chosen to suit the deployment.

In the 5GCity deployment, the small cell units will be configured to operate as L2/L1 remote radio heads which are slaved to the virtualized eNB control plane (L3) which runs at the edge.

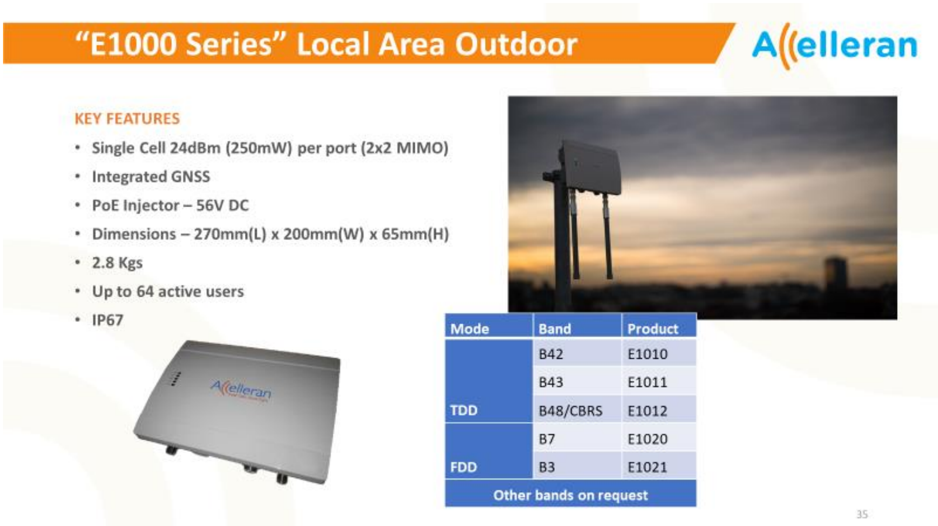

Figure 36: Single Carrier Local Area E1000

Figure 37: Single Carrier Local Area E1000

5. Interfaces and Workflow

5.1. Interfaces

In this paragraph the 5GCity external interfaces are listed with a focus on their description and major references.

- **OSS/BSS - Dashboard** (DHB.E1): Os-Ma-NFVO reference point is used for information exchanges between OSS/BSS and the NFV Orchestrator (NFVO). In the 5GCity architecture this is the interface to be used between the OSS/BSS and the Dashboard
- **5G Public App Catalogue - Dashboard** (DHB.E2): This interface is under definition, but follows ETSI GS SOL 005 specificationsand related NS and VNF descriptors (ETSI GS SOL 001, ETSI GS SOL 004)
- **User - Dashboard** (DHB.E3)**:** browser
- **User -SDK** (SDH.U): browser
- **SDK- Public App Catalogue (**SDK.E1): 5GCity SDK External interface, allows communication between SDK Toolkit and the Public catalogue platform.
- **5GCity Orchestrator (Infrastructure abstraction) - VIMs** (INFA.V1): It will be used an '*Or-Vi*' like interface
- **VIMs - Infrastructure Layer VI** (VII.E1): It will be used an '*Nf-Vi*' like interface
- **5GCity Orchestrator (WAN Resource Manager) - SDN ctrl /Fronthaul/BackHaul** (WAN:E1): This interface is under study
- **SDN controller - SDN agent** (SDN:E1): This interface is under study, it is likely to use an 'SDN CDPI (Ctrl data plane i/f)' like interface

The interfaces will be deeply detailed in the further iterations of this specification which will be collected in deliverable D2.3 and D.2.4.

5.2. Workflows

As mentioned earlier, the neutral hosting concept enables an ICT Infrastructure owner (IO) to logically segment its infrastructure in to slices and lease them to different users over the same physical infrastructure. That is, each slice user will have access to its part of the infrastructure resources (where infrastructure may include compute, storage and network resources) which are logically bundled into a slice. The slice user will have the flexibility to choose a network service from a catalogue and deploy, run/operate and manage the network service in the slice assigned to it.

5GCity project addresses all the necessary pieces of the puzzle for the neutral hosting ecosystem that are, SDK for network service development and 5GCity platform for converting an IO in to a neutral host along with infrastructure slicing and service provisioning.

This section describes high-level 5GCity workflows from network service creation to its deployment and operation in a multi-tenant SDN/NFV based edge infrastructure.

5.2.1. Slice User Registration

Slice user registration is a mandatory step that allows a slice user entity (cf. Section 1.2) to request an infrastructure slice from the neutral host and later on provision services within its (Figure 38).

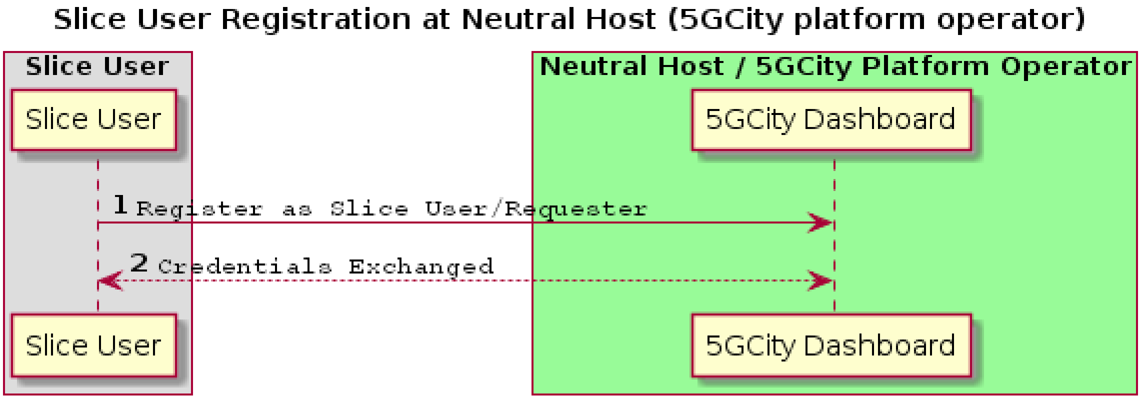

Figure 38: Slice User Registration at Neutral Host (5GCity platform operator)

Similar registration is required for service developers as well.

5.2.2. Service creation

Service creation (Figure 39) refers to the development of a service (a network service or a vertical oriented service). This service can be deployed (instantiated) by the slice user in one of its existing slices, i.e., once the slice is created and provisioned. The SDK pulls the services from a 3rd party service catalogue that are available for a Software Developer only after she/he has logged into the system, while the storing of a new service will also be performed on her/his account. It should be kept in mind that the SDK will be potentially downloaded and installed by the Software Developer, although it could be, in principle, offered as a Cloud service as well.

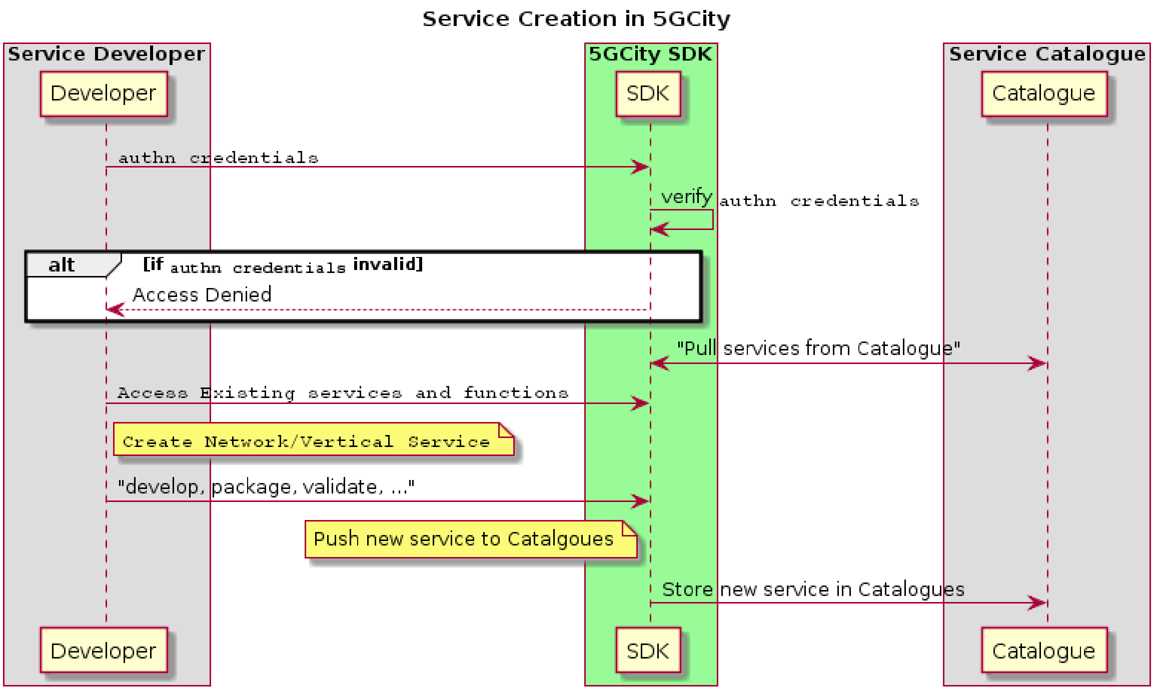

Figure 39: Service Creation in 5GCity

5.2.3. Slice creation

Slice creation (Figure 40) is performed whenever a slice user entity (cf. Section 1.2) requests some infrastructure resources managed by the neutral host through the Dashboard.

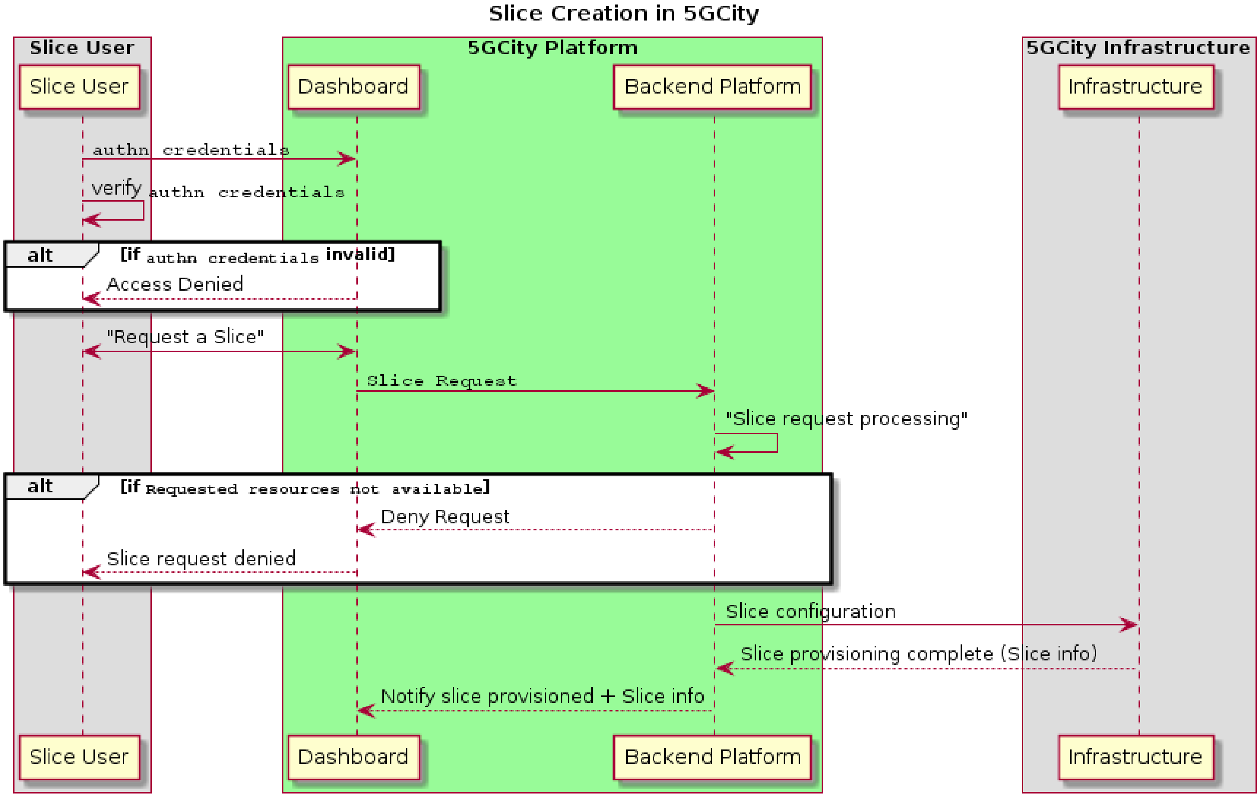

Figure 40: Slice Creation in 5GCity

5.2.4. Service Provisioning

Service provisioning (Figure 41) practically corresponds with the deployment of an instance of a previously created network/vertical service upon a previously commissioned slice.

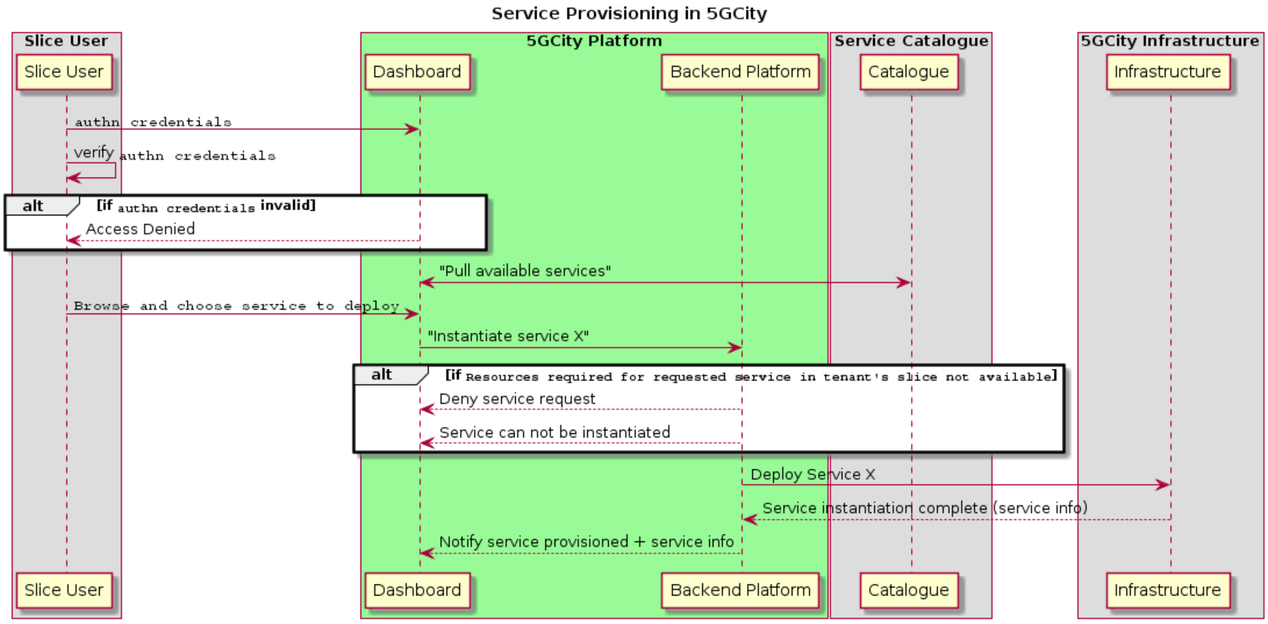

Figure 41: Service Provisioning in 5GCity

6. Conclusion

In this deliverable, we presented the first iteration of the 5GCity architecture built upon the pillars of SDN, network functions virtualisation, MEC and network slicing over the three-tier domains. The main objective was to design an architecture allowing us to deploy and to demonstrate, in operational conditions, a distributed cloud and radio platform for municipalities and infrastructure owners acting as 5G neutral hosts. The inputs for starting the design were based on the Use Cases definition and requirements provided by the deliverable D2.1.

We presented the overall architecture that includes the innovative components designed in order to address the neutral host model requirements and to enable the deployment of the 5GCity verticals in Barcelona, Bristol and Lucca. The 5GCity platform, including the Dashboard, the core block of the Orchestration & Control layer and the SDK, the core block of the Service layer, manage the interaction between the 5GCity infrastructure and the different roles envisaged in the business model. In order to address also the security requirements in accessing the 5GCity resources, they provide also the right level of privileges associated to each role. In the context of the NFVI, as innovation aspects, 5GCity architecture integrates two types of radio elements: Small Cells and Wi-Fi access points. The concept of resource slicing is extended up to the radio elements, enabling the inclusion of wireless connectivity in the slice. As the radio parts are managed by SDN, they can be integrated dynamically with other virtualized resources, such as VNFs and the wired backhaul, so an end-to-end slice can be generated. Furthermore, in the last section of the deliverable, we described the high-level 5GCity workflows from network service creation to its deployment and operation in a multi-tenant SDN/NFV based edge infrastructure.

The previous sections depict the high-level overall architecture of the 5GCity platform, including interfaces, workflows and the initial architectural vision for all the subsystems. Using an iterative approach, the feedback received from the detailed subsystems design and specification as well as from the following phases of implementation will help to refine and amend the overall architecture providing a second iteration on M19 and a final version on M25.

7. References

[1] 5GCity Project Deliverable D2.1: “5GCity System Requirements and Use Cases”, H2020 5GCity project, December 2017
[2] Project ACRN, “Project ACRN documentation”, online at https://projectacrn.github.io/
[3] Object Management Group (OMG), “Data Distribution Service (DDS) specification”, online at: https://www.omg.org/spec/DDS/1.4/
[4] 5GCity Project Deliverable D4.1: “Orchestrator design, service programming and machine learning models”, H2020 5GCity project, May 2018
[5] 5GCity Project Deliverable D3.1: “5GCity Edge Virtualization Infrastructure Design”, H2020 5GCity project, May 2018
[6] Angelo Corsaro, Erik Boasson, “Zero network over-head”, http://zenoh.io/download/pdf/2018.04.23-zenoh.pdf
[7] ETSI, “NFV Management and Orchestration; Network Service Templates Specification”, ETSI GS NFV IFA 014, online available at http://www.etsi.org/deliver/etsi_gs/NFV-IFA/001_099/014/02.01.01_60/gs_NFV-IFA014v020101p.pdf
[8] ETSI, Network Functions Virtualisation (NFV); Terminology for Main Concepts in NFV, ETSI GS NFV 003
[9] ETSI, “Network Functions Virtualisation (NFV); Ecosystem; Report on SDN Usage in NFV Architectural Framework”, ETSI GS NFV-EVE 005 V1.1.1
[10] OpenStack, online at: https://wiki.openstack.org/wiki/Main_Page
[11] T. Sechkova, M. Paolino and D. Raho, “Virtualized Infrastructure Managers for edge computing: OpenVIM and OpenStack comparison”, IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (IEEE BMSB 2018), Valencia, Spain, June 2018
[12] OpenVIM documentation, online at: https://osm.etsi.org/wikipub/index.php/OpenVIM_installation_(Release_THREE)
[13] ETSI, “Mobile Edge Computing (MEC) Framework and Reference Architecture”, ETSI GS MEC 003 V1.1.1, (2016-03)
[14] Broadband Forum, “TR-069: CPE WAN Management Protocol”, online at https://www.broadband-forum.org/technical/download/TR-069_Amendment-6.pdf

Abbreviations and Definitions

7.1. Abbreviations

3GPP	3rd Generation Partnership Project
5G-PPP	5G Infrastructure Public Private Partnership
AR	Augmented Reality
BBU	Baseband Unit
CAPEX	Capital Expenditure
CCAM	Cooperative Connected and Automated Mobility
CDVS	Compact Descriptor for Visual Search
CP	Control Plane
CPE	Customer Premise Equipment
C-RAN	Cloud-RAN
DC	Data Center
DDS	Data Distribution Service for Real Time Systems
DOF	Degree of Freedom
eMBB	Enhanced Mobile Ultra Broadband
eNB	Evolved Node B
E2E	End to End
EPC	Evolved Packet Core
FCAPS	Fault, Configuration, Accounting, Performance and Security
FDD	Frequency Division Duplex
FFT	Fast Fourier Transform
FoV	Field of View
GPS	Global Positioning System
H24	24 Hours a day operation
HD-SDI	High Definition Serial Digital Interface
HDMI	High Definition Multimedia Interface
HR	High Resolution
IaaS	Infrastructure as a Service
ICT	Information Communication Technology
IoT	Internet of Things
IP	Internet Protocol
ITS	Intelligent Transportation System
ITU-T	International Telecommunication Union – Telecommunication Standardization Bureau
LTE	Long Term Evolution
LTE-A	Long Term Evolution Advanced
LPWA	Low Power, Wide Area (network)
mMTC	Massive Machine Type Communication
MAC	Medium Access Control
M2M	Machine to Machine
M&E	Media & Entertainment
MEC	Multi access Edge Computing (formerly Mobile Edge Computing)
MR	Mixed Reality
MNO	Mobile Network Operator
MOCN	Multi Operator Core Network
MORAN	Multi-Operator Radio Access Network

MVNO	Mobile Virtual Network Operator
NGMN	Next Generation Mobile Networks
NFV	Network Function Virtualization
NFVI	Network Function Virtualization Infrastructure
NFVO	Network Function Virtualization Orchestration
NS	Network Service
OPEX	Operative Expense
OTT	Over-The-Top Player
PaaS	Platform as a Service
PoP	Point of Presence
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RAT	Radio Access Technology
RBAC	Role Base Access Control
RF	Radio Frequency
RLC	Radio Link Control
RO	Resource Orchestrator
RRH	Remote Radio Head
RU	Radio Unit
SaaS	Software as a Service
SD-SDI	Standard Definition Serial digital interface
SDK	Software Development Kit
SDN	Software Defined Network
SLA	Service Level Agreement
SME	Small Medium Enterprise
TAG	Text and Graphics
TDD	Time Division Duplex
TEE	Trusted Execution Environment
uRLLC/uMTC	Ultra reliable communication/Low latency Communication
UHD	Ultra High Definition
UHDTV	Ultra High Definition Television
UP	User Plane
URI	Uniform Resource Identifier
vBBU	virtual Baseband Unit
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Network
V2P	Vehicle-to-Pedestrian
V-RAN	Virtual Radio Access Network
VIM	Virtual Infrastructure Manager
VM	Virtual Machine
VoD	Video on Demand
VR	Virtual Reality
VNF	Virtual Network Function
VNFM	Virtual Network Function Manager

<END OF DOCUMENT>