



Project Title	Virtual Presence in Moving Objects through 5G
Project Acronym	PriMO-5G
Grant Agreement No	815191
Instrument	Research and Innovation Action
Topic	The PriMO-5G project addresses the area of “a) Focus on mmWave and super broadband services” in the call “EUK-02-2018: 5G” of the Horizon 2020 Work Program 2018-2020.
Start Date of Project	01.07.2018
Duration of Project	36 Months
Project Website	https://primo-5g.eu/

D4.2 - APIS FOR AI-ASSISTED NETWORK SLICE ORCHESTRATION

Work Package	WP4, AI-assisted Communications
Lead Author (Org)	KCL
Contributing Author(s) (Org)	Abbas Waqar (AALTO), Ozgur Akgul (AALTO), Edward Mutafungwa (AALTO), Jose Costa-Requena (CMC), Zere Ghebretensaé (EAB), Amin Azari (EAB, reviewer), Konstantinos Antonakoglou (KCL), Toktam Mahmoodi (KCL), Xiaolan Liu (KCL, reviewer)
Due Date	31/08/2020
Date	09/09/2020
Version	3.0, submitted

Dissemination Level

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



The work described in this document has been conducted within the project PriMO-5G. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 815191. The project is also supported by the Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2018-0-00170, Virtual Presence in Moving Objects through 5G). The dissemination of results herein reflects only the author's view, and the European Commission, IITP and MSIT are not responsible for any use that may be made of the information it contains.

Versioning and contribution history

Version	Date	Authors	Notes
1.0	24.03.2020	K. Antonakoglou (KCL)	Draft ToC
1.1	21.04.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO)	First consolidated version
1.4	15.06.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB)	Fourth consolidated version
1.5	29.06.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB)	Fifth consolidated version
1.6	14.07.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	6 th consolidated version
1.7	20.07.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	7 th consolidated version
1.8	20.07.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	8 th consolidated version (merged AALTO's content from previous revision)
2.1	03.08.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	11 th consolidated version
2.3	05.08.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	Addition of Acronym's list, introductory content for section 4.2 and expansion of Introduction
2.4	09.08.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	References section fixed
2.5	25.08.2020	K. Antonakoglou (KCL), Jose Costa-Requena (CMC), Mutafungwa Edward (AALTO), Zere Ghebretensaé (EAB), Toktam Mahmoodi (KCL)	Added major comments and resolved minor comments from internal reviews.
2.7	30.08.2020	Abbas Waqar (AALTO), Ozgur Akgul (AALTO), Edward Mutafungwa (AALTO), Jose Costa-Requena (CMC), Zere Ghebretensaé (EAB), Konstantinos Antonakoglou (KCL), Toktam Mahmoodi (KCL)	Consolidated version after first round of updates (Sections 1, 3 and 4) after internal review.

Version	Date	Authors	Notes
2.8	31.08.2020	Abbas Waqar (AALTO), Ozgur Akgul (AALTO), Edward Mutafungwa (AALTO), Jose Costa-Requena (CMC), Zere Ghebretensaé (EAB), Konstantinos Antonakoglou (KCL), Toktam Mahmoodi (KCL)	Consolidated version after second round of updates (Sections 5) after internal review.
2.9	07.09.2020	Abbas Waqar (AALTO), Ozgur Akgul (AALTO), Edward Mutafungwa (AALTO), Jose Costa-Requena (CMC), Zere Ghebretensaé (EAB), Konstantinos Antonakoglou (KCL), Toktam Mahmoodi (KCL)	Final consolidated version
3.0	09.09.2020	Edward Mutafungwa (AALTO)	Final editorial checks prior to submission

Disclaimer

PriMO-5G has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 815191. The project is also supported by the Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2018-0-00170, Virtual Presence in Moving Objects through 5G). The dissemination of results herein reflects only the author's view, and the European Commission, IITP and MSIT are not responsible for any use that may be made of the information it contains.

Table of Contents

Executive Summary.....	7
List of Acronyms.....	8
1 Introduction	11
1.1 Scope of the document.....	11
1.2 Structure of the document.....	11
1.3 Relationship to other project outcomes	11
2 Network Service and Capability Exposure.....	13
2.1 Overview	13
2.2 Open Service Access (OSA).....	13
2.2.1 OSA Framework Architecture.....	13
2.3 Service Exposure in LTE	15
2.3.1 SCEF architecture for service exposure	17
2.4 Network Service and Capability Exposure in 5G	20
2.4.1 5G network architecture.....	20
2.4.2 Network Exposure Function (NEF).....	21
2.4.3 Common API Framework (CAPIF)	23
2.4.4 3GPP Network Exposure for Slice management	25
3 Backhaul and MEC.....	29
3.1 Overview	29
3.2 Mobile backhaul and resource management	29
3.3 MEC optimal placement.....	30
4 Radio Access Network (RAN) Slicing.....	33
4.1 Overview	33
4.2 RAN slice definitions.....	33
4.2.1 3GPP	34
4.2.2 GSMA	34
4.3 RAN Slicing over RAN architecture.....	35
4.3.1 Flexible Function Split as RAN slicing enabler.....	35
4.3.2 Slicing over multiple path fronthaul.....	36
4.4 Open APIs for RAN slice preparation and lifecycle management.....	39
4.4.1 O-RAN Alliance	39

4.4.2	Small Cells Forum 5G FAPI	41
4.4.3	Other open APIs for RAN control	43
4.5	Dynamic RAN slicing	45
4.5.1	Feasibility Analysis	46
4.6	Service continuity	50
4.6.1	Adaptation to the changing channel conditions and tenant strategies	51
4.6.2	Adaptation to the changing traffic mix.....	53
5	Conclusions and Outlooks	55
6	References.....	56

List of Tables

Table 1.1	PriMO-5G scenarios and use cases	12
Table 2.1	Relationship between CAPIF, 5GS and EPS network exposure aspects.....	25
Table 2.2	Exposure of network slice management data for network slice as a service case.....	26
Table 2.3	Exposure of network slice management capability.....	27
Table 4.1	Values standardised by 3GPP TS 23501 [3GPP-23501].....	34
Table 4.2	Fronthaul link and traffic parameters for URLLC and eMBB.....	37
Table 4.3	Comparison between measured time complexities (ms) of feed-forward neural network (FFNN), convolutional neural networks (CNN), the long-short term memory (LSTM) and the optimization-based model (P2).	48
Table 4.4	Performance comparison between different AI blocks (Lower gap values indicates higher performance).....	49

List of Figures

Figure 1.1	PriMO-5G work structure.....	11
Figure 2.1	Open Service Access Architecture	14
Figure 2.2	3GPP architecture reference model for MTC (non-roaming)	16
Figure 2.3	3GPP SCEF architecture for service exposure	17
Figure 2.4	CAPIF functional model representation using service-based interfaces	24
Figure 2.5	NSI related management data exposure to customer	28
Figure 3.1.	Round Trip Time (RTT) results of network congestion impact to URLLC traffic with MBO managed slices.	30
Figure 3.2	MEC platform deployment.....	30

Figure 3.3 Sequence flow of “Procedure for Traffic Influence” used by AF to request support for MEC capabilities. 31

Figure 4.2 GST and NEST in context of the network slice lifecycle (NSC = Network Slice Customer) [GSMA2019] 35

Figure 4.3 Example of envisioned architecture of RAN functional split supporting RAN slicing..... 36

Figure 4.4 Probability of Error in orthogonal sharing, in the case of single path (SP), multiple path with duplication (MPD) and multiple path with encoding (MPC). 38

Figure 4.5 Probability of Error in non-orthogonal sharing, in the case of single path (SP), multiple path with duplication (MPD) and multiple path with encoding (MPC). 38

Figure 4.6. O-RAN Reference Architecture (source: O-RAN) 40

Figure 4.7 Small cell internal architecture..... 43

Figure 4.8. Example interactions via PHY API..... 43

Figure 4.9 FlexRAN framework from [MOS5G20]..... 44

Figure 4.10 5G-EmPOWER framework from [5GEMP20]..... 45

Figure 4.11 Considered 2-step algorithm as proposed in [AMC19] 47

Figure 4.12 Performance comparisons between different algorithms for two different RI, RI=10 TTI (on the left) and RI=100 TTI (on the right)..... 49

Figure 4.13 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)..... 50

Figure 4.14 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)..... 50

Figure 4.15 Performance comparisons between different algorithms (P2 (on the left) and LSTM (on the right)) for RI=50 TTI..... 52

Figure 4.16 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)..... 52

Figure 4.17 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)..... 53

Figure 4.18 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)..... 53

Figure 4.19 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)..... 54

Executive Summary

This document aims to present standardized frameworks and architectural components for exposing network services and capabilities to external network users and application developers. This is aligned with the purposes of PriMO-5G to satisfy the KPIs of the defined use cases. This deliverable presents an overview, definitions as well as an analysis on APIs and the necessary network architecture which allow the management of resources for MEC platform deployments and RAN slices from external network users and application developers.

In this way, applications related to studies already presented within this work package regarding AI-assisted networking and AI-assisted edge computing can be interfaced with exposed network services and capabilities.

The document is organized as follows:

- Section 2 presents an overview of technologies, APIs and architecture elements that allow service exposure in LTE and 5G.
- Section 3 focuses on resource management regarding mobile backhaul and the optimal placement of the MEC. This section describes the network functions defined in 3GPP to expose packet core functionality to external applications.
- Finally, Section 4 provides RAN slicing definitions, an overview and analysis of function splits within RAN, a presentation of open APIs for RAN slicing as well as an analysis for dynamic RAN slicing and service continuity using AI-based solutions.

List of Acronyms

Acronym	Definition
2G	Second Generation Mobile Network
3G	Third Generation Mobile Network
3GPP	3rd Generation Partnership Project
5G	Fifth Generation Mobile Network
5GC	5G Core Network
5GS	5G System
ABF	Analog Beamforming
AI	Artificial Intelligence
AMF	Access and Mobility management Function
API	Application Programming Interface
AS	Application Server
B2B	Business to business
B2B2X	Business to everything
B2C	Business to consumer
CAPEX	Capital Expenditure.
CAPIF	Common API Framework
CIoT	Cellular IoT
CN	Core Network
CNN	Convolutional Neural Network
CP	Communication Pattern
CPU	Central Processing Unit
CSC	Communication Service Customer
CSMF	Communication Service Management Function
CSP	Communication Service Provider
CU	Central Unit
DCSP	Data Centre Service Provider
DiffServ	Differential Services
DNAI	Data Network Access Identifier
DNN	Data Network Name
DU	Distributed Unit
eDRX	Extended idle-mode Discontinuous Reception
eMBB	Enhanced Mobile Broadband
eMBMS	Evolved Multimedia Broadcast/Multicast Services
EPC	Evolved Packet Core
EPS	Evolved Packet System
FAPI	Functional Application Platform Interface
FEU	Frontend Unit
FFNN	Feed-forward Neural Network
FH	Fronthaul
FW	Framework
GERAN	GSM EDGE Radio Access Network
GMLC	UMTS Terrestrial Radio Access Network
GSM	GSM Association
GST	Generic Network Slice Template
HD	High Definition
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IN	Intelligent Network
IN-CSE	Infrastructure Common Services Entity

Acronym	Definition
IoT	Internet of Things
IT	Information Technology
IWF	Inter Working Function
KPI	Key Performance Indicator
LAN	Local Area Network
LSTM	Long-Short Term Memory
LTE	Long Term Evolution
MAC	Medium Access Control
MBO	Mobile Backhaul Orchestrator
MEC	Multi-Access Edge Computing
MIoT	Massive IoT
ML	Machine Learning
MME	Mobility Management Entity
MPC	Multiple Path with Coding
MPD	Multiple Path with Duplication
MTC	Machine Type Communication
NAPS	Northbound APIs for SCEF – SCS/AS Interworking
NEF	Network Exposure Function
NEST	Network Slice Type
NGMN	Next Generation Mobile Networks Alliance
NG-RAN	Next Generation Radio Access Network
NIDD	Non-IP Data Delivery
NMM	Network Monitor Mode
NOP	Network Operator
NRF	Network Repository Function
NSaaS	Network Slice as a Service
NSC	Network Slice Customer
NSMF	Network Slice Management Function
NSP	Network Slice Provider
NSSAI	Network Slice Selection Assistance Information
NSSMF	Network Slice Subnet Management Function
NWDAF	Network Data Analytics Function
OAM	Operations, Administration and Maintenance
OPEX	Operating Expense
ORAN	Open RAN Alliance
OSA	Open Service Access
OTT	Over-The-Top
PCRF	Policy and Charging Rules Function
PDN	Packet Data Network
PDU	Protocol Data Unit
PHY	Physical Layer
PLMN	Public Land Mobile Network
PRB	Physical Resource Blocks
PRI	Priority
PSM	Power Saving Mode
QoS	Quality of Service
RAM	Random Access Memory
RAN	Radio Access Network
RAT	Radio Access Technology
REST	Representational state transfer
RI	Renegotiation Interval

Acronym	Definition
RIC	Radio Intelligent Controller
RRC	Radio Resource Control
RRM	Radio Resource Management
RT	Real-Time
RU	Radio Unit
SBA	Service-based architecture
SBI	Service-based Interfaces
SCEF	Service Capability Exposure Function
SCF	Service Capability Feature
SCS	Service Capability Servers
SD	Slice Differentiator
SDK	Software Development Kit
SDN	Software-defined Network
SGSN	Serving GPRS Support Node
SLA	Service Level Agreement
S-NSSAI	Single NSSAI
SON	Self-organizing Network
SP	Single Path
SST	Slice/Service Type
ToS	Type of Service
TR	Technical Report
TS	Technical Specification
TTI	Transmission Time Interval
UDM	Unified Data Management
UDR	Unified Data Repository
UE	User Equipment
UPF	User Plane Function
URLLC	Ultra-Reliable Low-Latency Communication
UTRAN	UMTS Terrestrial Radio Access Network
V2X	Vehicle-To-Everything
VISP	Virtualization Infrastructure Service Provider
VLAN	Virtual LAN
VN	Virtual Network
VoLTE	Voice over LTE
VPN	Virtual Private Network

1 Introduction

1.1 Scope of the document

The main aim of the PriMO-5G project is to demonstrate an end-to-end 5G system providing immersive video services for moving objects, showcasing application-driven algorithms for autonomous networking with distributed learning-based computations from AI-assisted networking and edge computing research. The title of deliverable 4.2 is “APIs for AI-assisted network slice orchestration” and its purpose is to present and discuss frameworks and architectural components that expose network services and capabilities to external network users and application developers supported by open and standardized APIs. This deliverable is also aligned with Task 4.3 which concerns the design and implementation of APIs for applications to request relevant KPIs, e.g. latency or throughput, from the core network and translate such information into slice or MEC orchestration in order to support multiple verticals with different requirements.

1.2 Structure of the document

Deliverable 4.2 consists of three main sections regarding network service and capability exposure, backhaul and MEC as well as RAN slicing. Section 2 investigates the proposed architectures, APIs and standardisation activities regarding network service and capability exposure in LTE and 5G. Section 3 focuses on mobile backhaul and specifically MEC discussing network functionality exposure for resource management and optimal placement of the MEC. Section 4 concerns RAN slicing, firstly presenting the relevant definitions and APIs and secondly an analysis of dynamic RAN slicing and service continuity.

1.3 Relationship to other project outcomes

The overall work structure of PriMO-5G project is illustrated in Figure 1.1. In this work structure, WP1 specifies the PriMO-5G firefighting use cases that inspired research and technology developments in WP2, WP3, and WP4. Deliverable 4.2 includes input related to the interaction of WP4 with WP1 use cases and the end-to-end 5G system architecture, as well as WP5 regarding the PriMO-5G demonstrations.

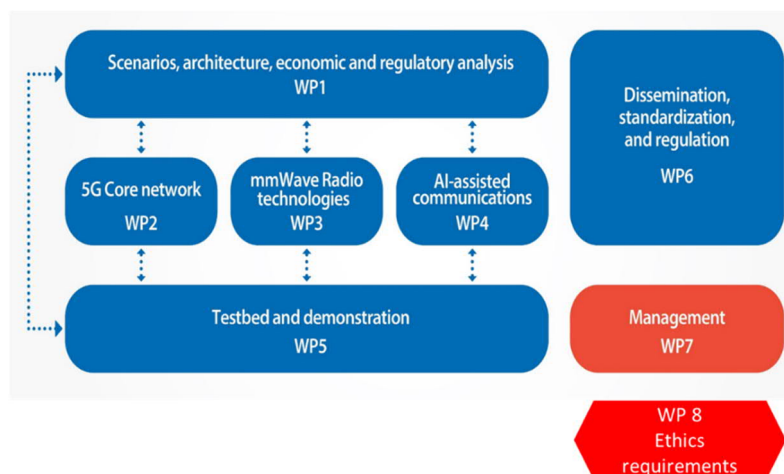
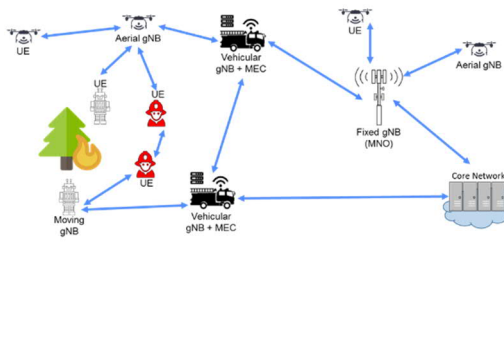
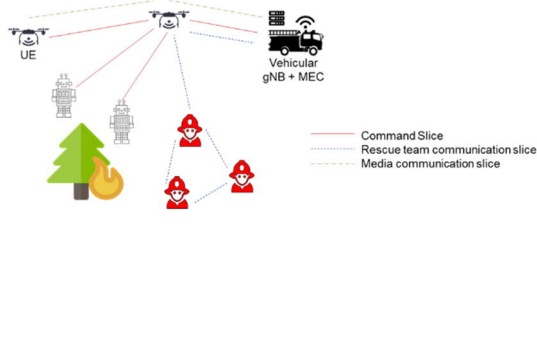
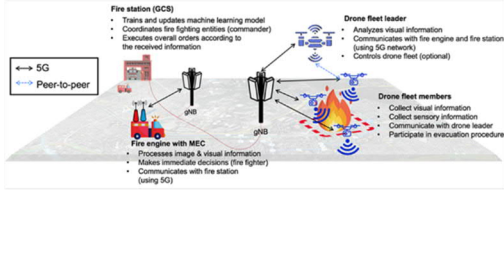


Figure 1.1 PriMO-5G work structure

As noted previously, this present and discuss frameworks and architectural components that expose network services and capabilities to external network users and application developers, focusing on the PriMO-5G firefighting scenarios and use cases described in *D1.1 PriMO-5G Use Case Scenarios* [PRIMO-D11]. These include two scenarios, namely, forest firefighting in rural areas and firefighting in

urban areas, with each of these scenarios having two associated use cases. Table 1.1 provides a summary description of the use cases.

Table 1.1 PriMO-5G scenarios and use cases

Scenarios	Use cases	
Scenario A: Forest firefighting with robots and UAVs	<p><i>Use case A1. Placement of communication and computing for forest firefighting</i></p> 	<p><i>Use case A2. Network slice management for forest firefighting</i></p> 
	Scenario B: Smart firefighting with UAVs in urban area	<p><i>Use case B1. UAV-assisted preparatory measures for smart urban firefighting</i></p> 

2 Network Service and Capability Exposure

2.1 Overview

In the last few decades, mobile technology has evolved through a number of generations and it has now reached an impressive technical maturity transforming our way of communication. The society as a whole has benefitted from increased communication services, while creating huge revenues to network operators. The technical evolution that started with voice optimized mobile network has with the popularity of the Internet evolved to data optimized IP network supporting a variety of services that produce huge amount of data traffic. Over the last decade, however, with the proliferation of video services, the volume of video traffic has exploded exponentially, leaving the operators with huge CAPEX and OPEX, while the revenues from this increased traffic remained flat. On top of that, because of the increased competition from the Over-the-Top (OTT) service providers, that play by different rules and monetize their services through different business models, there is an increased risk of commoditization of the services provided by the operators, leaving them as data pipe providers.

It therefore comes as no surprise to learn that network operators have been studying different ways to diversify their services, including opening up their network functionality for application developers, to increase their customer base. By opening up the network, operators' or third-party applications will be able to access the core network functionalities of the networks by means of open standardized APIs. The goal is to enable new business models where new and innovative applications can be developed by enterprises outside the traditional network operator domain and this opens new sources of revenues for incumbent network operators as well as diversify their customer base.

Over the years 3GPP and other standardization bodies have carried out a number of studies and specifications to incorporate new functional blocks to their legacy network, in an effort of opening up the network services and capabilities to third-party application developers using open standardized APIs.

2.2 Open Service Access (OSA)

One of the early standardization activities on telecommunication network APIs, that had a deep impact the work of 3GPP API standardization was the standardization work of the Parlay group. The Parlay Group [PAR-OSA], is a multi-vendor consortium formed to develop open, technology-independent APIs that enable the development of applications that operate across multiple, networking-platform environments. Parlay integrates Intelligent Network (IN) services with IT applications via a secure, measured, and billable interface. In the early year 2000, the 3GPP and ETSI initiated a joint work on APIs for the network by adopting the Parlay specifications as a basis for their Open Service Access (OSA) API. The aim of OSA is to provide a standardized, extensible and scalable interface that allows for inclusion of new functionality in the network with a minimum impact on the applications using the OSA interface.

The 3GPP OSA specification work started by identifying several requirements of the type of functional features that can be exposed and the necessary security, charging, policy management, event notifications that must be supported by OSA [3GPP-22127]. For example, the OSA charging functionality should be able to allow an application to raise charges for call, session, events and service usage such as online purchases, by enabling applications to control the charge of the call, session and an event. Over the years 3GPP has published number specifications including on Service requirements TS 22.127, OSA architecture TS 23.127, TS 23.198 and OSA APIs TS 29.198 and TR 29.998

2.2.1 OSA Framework Architecture

The OSA specifications define an architecture that enables service application developers to make use of network functionality through an open standardized interface, i.e. the OSA API's [3GPP-23127] [3GPP-23198]. According to this specification the network functionality offered to applications is defined as a set of Service Capability Features (SCFs) in the OSA API supported by different Service Capability

Servers (SCS). These SCFs and OSA open interface architecture provide access to the network capabilities on which the application developers can leverage to rapidly design new innovative applications or enhance already existing ones. The Open Service Access (OSA) functional architecture, shown in Figure 2.1, consists of three components: Applications, Framework and Service Capability Servers (SCSs):

- **Applications** such as call forwarding applications, conferencing, VPNs and location-based services, deployed in standard IT Application Servers (ASs) running in the operator or enterprise domain. The applications use the services capabilities defined in OSA and provided by the SCSs and offered to applications through the OSA APIs.
- **Framework** which may reside within one of the physical entities containing the Service Capability Servers or in a separate physical entity, provides applications with basic mechanisms that enable them to make use of the service capabilities in the network. Examples of framework functions are authentication, registration and discovery services. Service Capability Features made available to applications are registered in the framework and before an application can use these network Service Capability Features, a two-way authentication between the application and framework has to be conducted. After authentication, the discovery function enables the application to find out the Service Capability Features that are provided by the Service Capability Servers., which can then be accessed using the methods defined in the OSA interfaces.
- **Service Capability Servers (SCSs)** provide the applications with Service Capability Features (SCFs), which are abstractions from the underlying network functionality. Examples of SCFs offered by SCSs include Call Control and User Location. The SCSs that provide the OSA interfaces are functional entities that can be distributed across one or more physical nodes. For example, the User Location interfaces, and Call Control interfaces might be implemented on a single physical entity or distributed across different physical entities. Furthermore, a SCS can be implemented on the same physical node as a network functional entity, whose functional capabilities are being exposes by the SCS, or in a separate physical node.

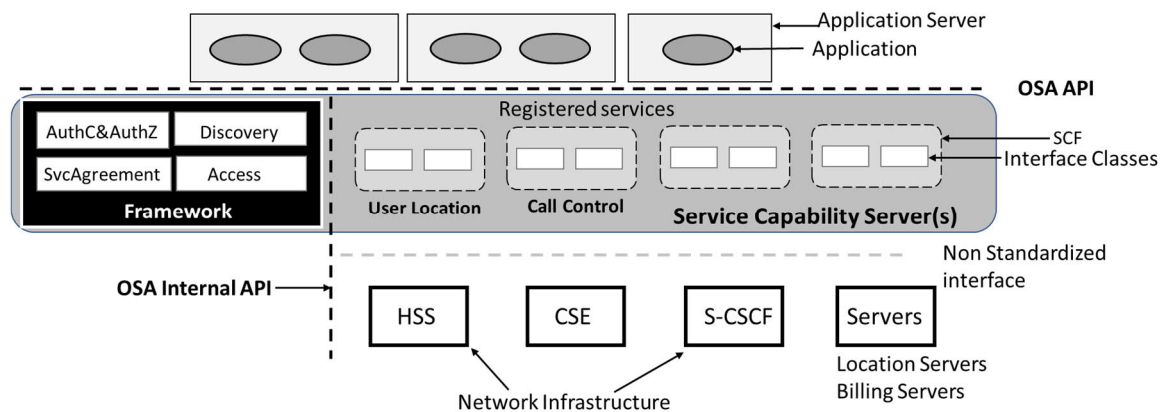


Figure 2.1 Open Service Access Architecture

The OSA API, shown as dashed lines in Figure 2.1, is split into three types of interface classes, Service and Framework [ETSI-OSA].

- Interface classes between the Applications and the Framework (FW) that provide applications with basic mechanisms (e.g. Authentication) that enables them to make use of the service capabilities in the network.

- Interface classes between Applications and SCFs, which are individual services that may be required by the client to enable the running of third-party applications over the interface e.g. Messaging type service
- Interface classes between the Framework (FW) and the SCFs that provide the mechanisms necessary for a multi-vendor environment.

The applications deployed in the Application Servers in Figure 2.1, use the service capabilities specified in the OSA framework and provided by the SCSs through the OSA APIs. This means the OSA SCSs, implement the server side while the applications implement the client side of the OSA APIs. The SCSs are thus the logical entities that implement the API (that is the interface classes of the Service Capability Features) and interact with the core network functional elements such as HSS, MSC, etc. As such the SCSs serv as a proxy or gateway to the core network [MK03]. The applications servers can be in the same business domain as the service SCSs, i.e., both are within the operator domain or in different domain where the Application Servers are provided by a third-party application developer.

2.3 Service Exposure in LTE

The OSA architecture and the APIs described in the previous section were developed to expose the network capability of GERAN and UTRAN, i.e., 2G and 3G mobile networks to third-party service providers. With the evolution towards LTE and 5G systems and the increased demand to support a wide variety of verticals and use cases, however, the one-size-fits-all network paradigm employed in the past mobile networks is no longer optimal. Indeed, this became clear with the deployment of 4G networks and the envisioned traffic from massive Machine Type Communication (MTC) and Internet of Things (IoT) devices, which can potentially end up swamping network, compromising the security and QoS of all the services that share the same network infrastructure. As a result, 3GPP initiated a number of activities in an effort to identify and specify enhancements of the RAN and EPC for MTC and one of the main outcomes from the enhancement of EPC was the definition of Service Capability Exposure Function (SCEF) specified in TS 23.682 release 13 [3GPP-23682]. The SCEF has a central role in exposing the 3GPP services and capabilities to the SCS/AS via a set of APIs and hides the underlying 3GPP network topology from the SCS/AS.

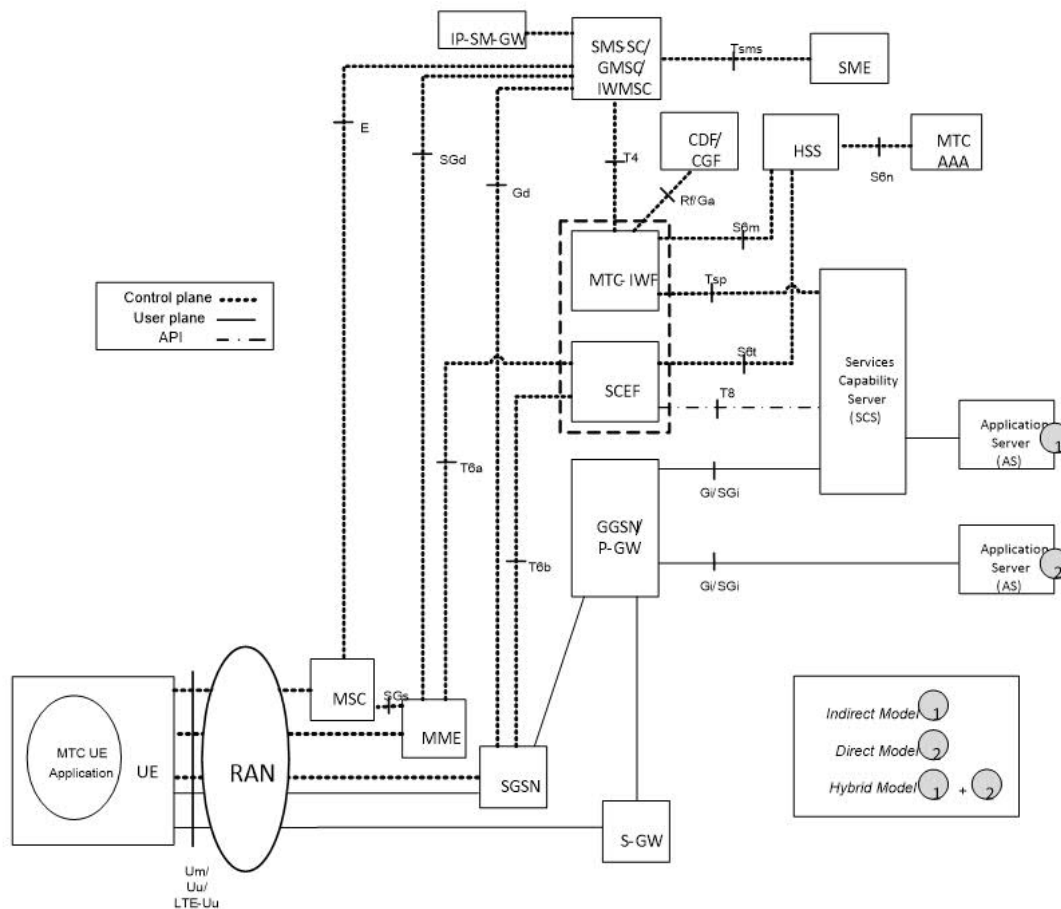


Figure 2.2 3GPP architecture reference model for MTC (non-roaming)

Figure 2.2, from TS 23.682 Release 19, shows the non-roaming 3GPP reference architecture for MTC, with three new functional entities MTC-Inter Working Function (MTC-IWF), Service Capability Servers (SCS), Service Capability Exposure Function (SCEF) and the corresponding interfaces between MTC-IWF and SCS and the API between SCEF and SCS.

The MTC-IWF is responsible for the termination of the Tsp interface towards the SCS and authentication and authorization of the SCS before communication establishment with the 3GPP network. It is also responsible for authorization of the control plane requests from the SCS and for diverse MTC device trigger related functions.

The SCS is an entity which connects to the 3GPP network to communicate with UEs used for MTC and the MTC-IWF and/or SCEF in the HPLMN. The SCS offers capabilities for use by one or multiple MTC Applications. A UE can host one or multiple MTC Applications. The corresponding MTC Applications in the external network are hosted on one or multiple ASs.

Depending on whether the MTC Service Provider (MTC-SP) or the Network Operator is providing the SCS, the 3GPP architecture reference model supports three types of communication models - direct, indirect and hybrid models, for communication of MTC traffic between the Application Server (AS) and the 3GPP system. In the direct model, the AS establishes a direct user plane communication, without external SCS. In the indirect model the AS accesses the network functionality via the SCS to, utilize additional value-added services provided by the SCS which can be owned by the operator or the MTC-SP. And in the hybrid model the AS accesses the network functionality using a mixture of both, i.e., the direct and indirect models.

2.3.1 SCEF architecture for service exposure

The network functions and interfaces of SCEF were first standardized in 3GPP TS 23.682 release 13. But release 13 only specified the southbound interface, i.e., the interface that connects the SCEF to internal core network nodes, i.e., the interfaces to MME, SGSN, HSS etc. The architectural definition and the northbound APIs for SCEF was standardized in TS 23.868 Release 15, as part Northbound APIs for SCEF – SCS/AS Interworking (NAPS) work and was given the name T8. The protocol definition of the Northbound APIs was defined in a new specification, TS 29.122 [3GPP-29122], which binds the T8 APIs to Hypertext Transfer Protocol (HTTP). Furthermore, a new specification TS 23.222 [3GPP-23222a] was created, which defines a common API framework (CAPIF) that specifies API aspects that are common to all Northbound API interfaces, such as registration, discovery, identity management, etc.

The SCEF exposes 3GPP services and capabilities to the SCS/AS via a set of APIs and hides the underlying 3GPP network topology from the SCS/AS. It provides secure exposure of the services and capabilities provided by 3GPP network interfaces and a means for the discovery of the exposed services and capabilities. It also abstracts the services from the underlying 3GPP network interfaces and protocols and provides access to network capabilities through homogenous network API defined over T8 interface. The co-location of MTC-IWF and SCEF deployment was added so that triggering services available via MTC-IWF can be provided over T8 using common API. Figure 2.3 shows SCEF architecture for service exposure and the interfaces to the network functional nodes, whose services and functionalities are exposed by the SCEF to the SCS/ASs.

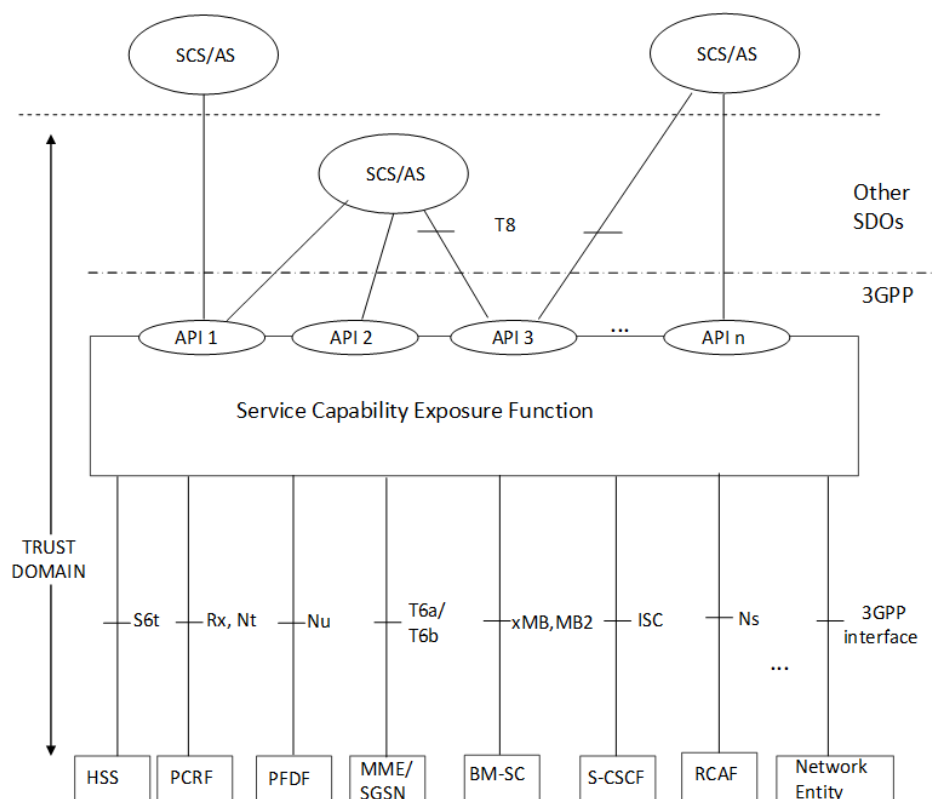


Figure 2.3 3GPP SCEF architecture for service exposure

Broadly speaking, the network services and functionalities that are exposed to the SCS/AS over the T8 APIs in TS 23.682 can be categorized into device Triggering and Event monitoring, Group Message

Delivery, UE and network service configuration, Efficient delivery of small payload data, Session management and Network resource optimization and Group Message Delivery.

The network services and capabilities exposed by the SCEF will be used in different use-cases, supported by types of terminals including MTC devices, IoT devices, UEs, etc. in the following, depending on the context the terms of MTC devices, IoT devices and UEs are used interchangeably

2.3.1.1 Device Triggering

Many IoT use cases demand IoT devices to be deployed once and remain unattended for their lifetime. In many cases these devices are powered by batteries therefore need to use their power very efficiently. One such method that can dramatically extend the battery life of IoT devices is the power saving mode (PSM). This mode is similar to power-off, but the device remains registered with the network and there is no need to re-attach or re-establish PDN connections. A device in PSM is still able to perform machine originated transmission without delay, therefore PSM is suitable for devices that can accept the corresponding triggering latency for mobile terminated communication. A UE that wants to use the PSM requests an Active Time value during every Attach and Tracking Area Update/ Routing Area update (TAU/RAU) procedures. If the network supports PSM and accepts that the UE uses PSM, the network confirms usage of PSM by allocating an Active Time value to the UE. In order to wake up or alert a UE in PSM, the network needs to support a device triggering service. Device triggering is subscription-based service that is exposed by SCEF over the T8 in which the AS/SCS sends information to the UE, using unique external identifiers, via the 3GPP network to trigger the UE to perform application specific actions that include initiating communication with the AS/SCS for the indirect model or with the AS in the network for the hybrid model. When device triggers are delivered via mobile terminated SMS the serving nodes MME, SGSN and MSC provide the service towards a specific UE based on the UE's subscription for mobile terminated SMS and other subscription parameters affecting SMS service provision.

2.3.1.2 Event monitoring

Similar to the device triggering service, the event monitoring feature is intended to monitor specific events in the network and make this information available to the SAS/AS. The monitoring features exposed by the SCEF over the T8 APIs for monitoring of specific events in the network, provide a means for identification of network elements suitable for configuring specific events, event detection and event reporting to authorized users. If such an event is detected, the network might be configured to simply report the occurrence of the event or perform special actions, e.g. limit the UE access. The type of UE related configuration and reporting of monitoring events that are supported by SCEF over the T8 include UE reachability, location and change of location, loss of connectivity, roaming status, number of UEs present in a specific geographical area, PDN connectivity Status. etc.

2.3.1.3 UE and network service configuration

In general, most of the MTC/IoT UEs are low cost, power constrained simple devices which require substantial support from the network, to perform their services. The 3GPP core network can support to configure UEs to use different features such Extended idle-mode Discontinuous Reception (eDRX), PSM, configure a IoT UE's use of the Enhanced Coverage feature, provide information to the RAN to help the RAN minimize UE state transitions and buffer downlink packets that are sent towards UEs that cannot be paged because they are in deep sleep. These parameters can, of course, be statically configured in the UEs subscription but instead, the features are exposed by the SCEF over the T8 APIs enabling the AS/SCS to dynamically configure different UE features and network parameters. For example, the AS/SCS can use the API to inform the network the maximum acceptable delay between the UE's reachability occasions and how long the UE needs to be available for mobile terminated data after it becomes reachable. The network can then use this information to configure the UE's DRX cycles, Power Saving Mode, and Tracking Area Update timer. The SCS/AS can also use this API to inform the network the number of downlink packets the network should buffer for the UE when it is sleeping and when a device is expected to communicate. The core network can use this information to create assistance information for the RAN. The RAN may then use the assistance information to minimize UE state transitions. Alternatively, the SCS/AS may instead request a one-time or a time-window, UE

reachability notification when a UE wakes up from its power saving state and sends downlink data to the UE when the UE is reachable, which may be suitable for infrequent mobile terminated communication.

2.3.1.4 Efficient delivery of small payload data

For power constrained IoT devices which need to transmit small payload traffic, utilizing power hungry IP protocol stack for these small data packets is very inefficient both in terms of protocol overhead and power consumption. Non-IP data delivery (NIDD) specified in TS 23.401 [3GPP-23401], as part of cellular IoT EPS optimizations, may be used to handle mobile originated and mobile terminated communication of unstructured or non-IP data traffic. The SCEF exposes APIs that allow the SCS/AS to exchange data with the UE using two different control plane delivery methods - SMS based and NAS-based Non-IP Data delivery. The SMS based service is always supported for cellular IoT EPS Optimizations and can be used simultaneously with Non-IP and IP data. For SMS Delivery, the SCEF exposes APIs that allow the SCS/AS to exchange data packets with UE hosted applications via SMS. When the UE hosts multiple SMS-based applications, the Port IDs are used on the API interface to multiplex traffic from multiple SMS-based applications. The UE uses the Application Port ID field of the SMS header to determine the receiving application and indicate the sending application. For data delivery over NAS, the SCEF exposes APIs that allow the SCS/AS to exchange data packets with the UE. The NAS data packets are associated with a PDN connection between the UE and the SCEF. When the UE hosts multiple applications that need to exchange small data packets with the SCEF, the data from the applications can be multiplexed onto separate PDN connections assuming that each UE hosted application has been provisioned with an Access Point Name (APN) that can be used to establish the application's PDN connection. 3GPP TS 24.250 [3GPP-24250] has also specified a standard for a Reliable Data Service protocol that can be used when exchanging data packets between the UE and SCEF. This protocol, which requires a 1 to 3-byte header on the non-IP data packet can be used to multiplex traffic from multiple applications onto the same PDN connection instead of provisioning separate APNs. Besides being used to multiplex traffic from multiple applications, the protocol can be used to simplify the UE hosted applications because it can be configured to provide functionality such as acknowledged and in sequence delivery.

2.3.1.5 Session management and Network resource optimization

The SCEF provides a number of APIs over the T8 interface that can be used to optimize the network resource management and session configuration. For example, the SCEF exposes APIs that allow the third-party SCS/AS, to request a session to be configured with specific QoS and priority handling. When the SCEF receives such request the SCEF acts as an Application Function as per TS 23.203 specifications and transfers the request to the PCRF via the Rx interface, between SCEF and PCRF, to retrieve the traffic profile. The SCS/AS may also provide the Communication Pattern (CP) of a UE to the SCEF in order to enable network resource optimizations for such UE(s). Communication patterns such as whether the UE is stationary or mobile, communication frequency between UE and the AS communication period, etc. such communication pattern information can be provided by third-party to the 3GPP network in order to tune the parameters and the MME could use such information when it deciding to send the information to the eNB side. The SCEF filters the CP parameters and forwards them to the HSS, which provides them to the MME. The MME may use the CP parameters as input to derive the CN assisted eNB parameters as described in TS 23.401.

The Network Status Reporting API allows the SCS/AS to receive reports from the 3GPP Network about congestion levels in a given area. Since IoT traffic is often time tolerant and IoT devices are often known to be in particular areas, this information can be used by the SCS/AS to delay interaction with UEs (e.g., polling) that are in a congested area until the congestion situation has subsided as illustrated in the following example from [SSW18]. Oftentimes, an SCS/AS may know that it needs to exchange data with a number of UEs in a geographical area, e.g., the SCS/AS may know that it needs to send a 10 Mbyte software upgrade to 100 sensors in the next 24 hours or that it needs to collect 1 Mbyte of data from 100 sensors sometime in the next 24 hours. In such scenarios, the Background Data Transfer API allows the SCS/AS to provide the network with the requirements of the data transfer (e.g. number of

UEs, how much data per UE, and time constraints). The network's policy engine (i.e., the Policy and Charging Rules Function/PCRF) is able to provide the SCS/ AS with a set of policy(s) that indicate the best time(s) to perform the data transfer. The SCS/AS can then activate the transfer policy for the UEs that are going to be involved in the data transfer. Thus, this API gives the network operator the ability to influence the SCS/AS to delay its activity until time periods where the network is typically underutilized and, in return, SCS/AS service providers may receive better charging rates.

2.3.1.6 Group Message Delivery

One of the main challenges of introducing MTC devices in the network is to be able to support numerous numbers of MTC devices without impacting the security and quality of service of the existing services that share the same network infrastructure. The Group Message Delivery is one of the many MTC related features exposed by the CSEF that can be used by the SCS/AS to request data delivery to group of devices using the evolved Multimedia Broadcast/Multicast Services (eMBMS) or via the unicast transmission of non-IP data delivery (NIDD). Delivery over the eMBMS has limited applicability since it cannot be used by UEs that do not support eMBMS or by UEs located in areas where eMBMS is not deployed. The group message delivery via the unicast mobile terminated NIDD is used for UEs which belong to the same External Group Identifier. When the SCS/AC needs to send non-IP data to a group of UEs, it submits a request which includes the external group identifier, maximum latency, PDN connection establishment option and other relevant identifiers to the SCEF. Based on the local policies the SCEF uses the SCS/AS Identifier and the External Group Identifier to determine the APN that will be used to send the non-IP data to the group member UEs. Therefore, in order for the non-IP data to reach each group member UE it assumes that each group member UE must have a PDN connection established to the APN and the SCS/AS must have performed an NIDD Configuration Procedure for the External Group Identifier. When the SCEF receives a Group MT NIDD request from the SCS/AS, the SCEF queries the HSS to resolve the group members and forks the message by sending it in a unicast manner to all of the individual UEs that are associated with the External Group Identifier.

2.4 Network Service and Capability Exposure in 5G

Similar to the previous generations, network exposure in 5G deals with the opening up of the services and capabilities of the network internally within the operator network or externally to third-party applications. Unlike the previous 'one size fits all' 3GPP network systems, however, 5G system was designed to provide optimized support for a variety of different communication services expanding the application space of the mobile network to support enhanced mobile broadband (eMBB) services, ultra-reliable low-latency communications (URLLC) and massive machine type communication (mMTC) applications, which can be operated by third parties. Therefore, 5G system was designed not only to support the stringent KPIs for latency, reliability, throughput, etc. required by these applications, but also to be able to expose the selected capabilities of the core network to external application functions (AF).

2.4.1 5G network architecture

Compared to LTE EPC architecture, 5G core network architecture can be described in two different ways. In its first incarnation, which is similar to LTE EPC architecture, it shows the way the different network functions are connected using the traditional point-to-point reference points or protocol interfaces. The protocol interfaces show the interactions between pairs of network functions and the main value of this type of representation is that it unambiguously shows, which network functions specifically consume the functions or services of which other network functions. In the second incarnation, the 5G Control plane system architecture assumes a service-based architecture (SBA), which means that the network functions of the core network offer their services to other network functions via interfaces of a common framework called Service Based Interfaces (SBI) to any of the network functions that are authorized to make use of these provided services. Therefore, in the SBA, the network functions assume two distinct roles, a service consumer role, when using the service

offered by other network functions and a service producer role when offering their services to other network functions. The communication method defined for SBA of the 5G core relies on the widely used HTTP REST paradigm that defines how the web communication technologies access services from distributed applications using APIs. One of the most important network functions of relevance to service and capability exposure identified in the SBA architecture as specified in TS 23.501 [3GPP-23501] is the Network Repository Function (NRF). The NRF supports service registration and discovery function by maintaining a set of network function profiles of the available network function instances, allowing every network functions function to discover the services offered by the other network functions. Before a service can be discovered, however, each network function must be provisioned with the address of the NRF and the network function that supports the service must register it to the NRF. This way, the NRF keeps track of all the services that are registered in the network. In general, a service offered by a network function can be requested on demand or it can be subscribed from the network function that offers the service. Another important network service and capability exposure related network function specified in the SBA architecture is the Network Exposure Function (NEF). The NEF is a new network function that supports interaction with external applications by exposing selected network capabilities for opening up new business opportunities enabling more advanced services to be offered by third-party application providers.

2.4.2 Network Exposure Function (NEF)

The Network Exposure Function (NEF) is a functional element responsible for exposing the services and capabilities within and outside the 5G core network. It supports secure exposure of network capabilities and events provided by 3GPP network functions to Application Functions (AF) and is a means for the AFs to securely provide information to 3GPP network. It may also authenticate, authorize and assist in throttling the AF and translate the information exchanged between the AF and the internal network functions. E.g., it translates between an AF-Service-Identifier and internal 5G Core information such as Data Network Name (DNN) and Single Network Slice Selection Assistance Information (S-NSSAI). The NEF also handles masking of network and user sensitive information to external AF's according to the network policy. In general, the NEF can receive information from the other network functions, stores the received information as structured data using a standardized interface to a Unified Data Repository (UDR) and then re-expose the data to other network functions and AFs used for other purposes such as analytics.

The Northbound interface, between the NEF and the AF specifies RESTful APIs that allow the AF to access the services and capabilities provided by 3GPP network entities and securely exposed by the NEF supports procedures for: monitoring, device triggering, management of background data transfer, CP Parameters Provisioning, packet flow description management, traffic Influence, i.e., changing the chargeable party at session set up or during the session and for setting up an AF session with required QoS. In general, the network services and capabilities exposed by the NEF over the northbound APIs can be categorized into Exposure of event monitoring and device triggering, Secure provision of information from external AF, Policy and charging control, Exposure of NWDAF analytics and retrieval of data by NWDAF and Support of Non-IP Data Delivery (NIDD).

2.4.2.1 Exposure of event monitoring and device triggering

This feature is used for monitoring of specific events in 3GPP system and securely exposing such monitoring events information by the NEF to third-party, Application Functions (AFs), Edge Computing, etc. It comprises of means that allow network functions in 5G system for configuring the specific events, the event detection, and the event reporting to the requested party. Depending on the operator deployment, certain AFs can be allowed to interact directly with the core network functions, with which they need to interact, while other AFs need to use the external exposure framework via the NEF. Typical examples of the monitoring events specified in TS 23.502 [3GPP-23502] and the network functions that detect these events include: Loss of connectivity of UE (AMF), UE reachability (AMF, UDM), Location

Report (AMF, GMLC), Roaming Status (UDM), communication failure (AMF), Number of UEs present in a geographical area (AMF), UE reachability for SMS delivery (UDM). In these cases, the AF may request to be informed about the network status, in a specific geographical area or for a specific UE, by sending a one-time network status request or to be continuously informed about the network status. Similarly, the application triggering service can be invoked by sending trigger delivery request to the NEF by the AF. An application trigger message contains information that allows the network to route the message to the appropriate UE and the UE to route the message to the appropriate application. The application in the UE may perform actions indicated by the Trigger payload when the Triggered payload is received at the UE. For example, initiation of immediate or later communication with the application server based on the information contained in the Trigger payload, which includes the PDU Session Establishment procedure if the related PDU Session is not already established

2.4.2.2 Secure provision of information from external AF

The provisioning capability of the NEF allows an external party to provision information, such as expected UE behaviour, 5G Virtual Network (5G VN) group information and Service Specific Information, in which case the NEF may also authenticate and authorize and assist in throttling the AFs. The provisioning of the expected UE behavioural information consists of information on expected UE movement and communication characteristics. Typically, these parameters consist of the expected UE trajectory, Stationary indication, Communication duration time, Periodic time, scheduled communication time, battery indication, traffic profile and scheduled communication type.

A 5G Virtual Network (5G VN) group consists of a set of UEs using private communication for 5GLAN-type services. The 5GLAN Group Management Function in the NEF may store the 5GLAN group information in the UDR via UDM as described in TS 23.502. This information of 5G VN group is provided by the AF to the NEF and is stored in the UDR, by using the NEF service operations information flow procedure. Typical 5G VN group information provisioning parameters consist of Data Network Name (DNN), S-NSSAI, PDU Session Type and Application descriptor. The service specific information consists of information to support the specific service in 5G system. The provisioned data can be used by the other NFs.

2.4.2.3 Policy and charging control

Policy and Charging allow external applications to control various aspects of data sessions. One example is the ability to influence traffic routing, for example to influence what and how local breakout and routing should be applied for certain devices. Also, QoS and charging policies can be controlled and enforced from external applications via the NEF providing information about the data traffic. The NEF interacts with the PCF for this, which in turn determines the QoS and charging information based on the application information provided by the AF/NEF. Other mechanisms include the support of negotiations about the transfer policies about future background data transfer and NEF support for allowing external applications to define the templates used by the UPF to detect that certain traffic is related to specific external application traffic. The NEF interacts with the SMF to achieve this and the SMF forwards these templates to the UPF. In this context the authentication and authorization of the external applications that want to interact with the NEF is very important in order to protect both the network and the devices from malicious interference or unauthorized information gathering.

2.4.2.4 Exposure of NWDAF analytics and retrieval of data by NWDAF

The NWDAF analytics can be accessed by any operator internal service consumers including network functions and OAM. But it may also be securely exposed externally to AFs by the NEF by using analytics subscription to NWDAF as specified in TS 23.288 [3GPP-23288]. The AF is configured, e.g., via static

OAM configuration, with the appropriated NEF to subscribe to analytics information, the allowed Analytics ID(s), and with allowed inbound parameters and parameter values for requesting each Analytics ID. The NEF controls the analytics exposure mapping among the AF identifier with allowed Analytics ID, and associated inbound restrictions (i.e., applied to subscription of the Analytics ID for an AF) and/or outbound restrictions (i.e., applied to notification of Analytics ID to an AF). Besides the exposure of the analytics the NWDAF can also retrieve data from external party. The Data Collection feature permits the NWDAF to retrieve data from various sources (e.g. network functions such as AMF, SMF, PCF, and AF and OAM), as a basis of the computation of network analytics. For this purpose, the NWDAF may directly perform data collection from the network functions or indirectly via the NEF from external AFs as specified in TS 23.288, where the NEF handles and forwards requests and notifications between NWDAF and AFs.

2.4.2.5 Support of Non-IP Data Delivery (NIDD)

For low power Cellular IoT (CIoT) devices which need to communicate with the network on a frequent or infrequent basis, utilizing power hungry IP protocol stack for data delivery provides a suboptimal solution. In 5G the delivery of Mobile Originated and Mobile Terminated communication of unstructured data (also referred to as Non-IP) is accomplished either by using UPF via a Point-to-Point N6 tunnel or by using the NIDD API. For this purpose, similar to the SCEF in LTE, the NEF provides a means for management of NIDD configuration and delivery of unstructured data by exposing the NIDD APIs on the N33/Nnef reference point. When a UE performs a PDU Session establishment with PDU Session type of "Unstructured" and the subscription information corresponding to the UE requested Data Network Name (DNN) includes the "NEF Identity for NIDD" (NEF ID), then the SMF initiates an SMF-NEF Connection establishment procedure towards the NEF corresponding to the "NEF ID" for that DNN/S-NSSAI Combination. This means, whether or not the NIDD API shall be invoked for a PDU session is determined by the presence of a "NEF Identity for NIDD" for the DNN/S-NSSAI combination in the subscription. If the subscription includes a "NEF Identity for NIDD" corresponding with the DNN and S-NSSAI information, then the SMF selects that NEF and uses the NIDD API for that PDU session. The NEF ID for a given DNN and S-NSSAI in the subscription can be updated by using the NIDD configuration procedure.

2.4.3 Common API Framework (CAPIF)

As shown in the previous sections 3GPP has developed multiple northbound API-related specifications over the years, despite the long list of specification work however, there is no authoritative standard for the capability exposure of the different 3GPP architectures. Therefore, in order to avoid duplication and inconsistency of approaches between different API specifications and to specify common services, 3GPP has with the specifications TS 23.222 [3GPP-23222b] and TS 29.222 [3GPP-29222] developed a common API framework (CAPIF). As shown in Figure 2.4, CAPIF specifies API aspects that are common services to all Northbound API interfaces that includes onboarding and offboarding of Application Functions, service discovery and management, event subscription and notification, security and charging, etc. The functional model for the common API framework (CAPIF) is organized into five functional entities of CAPIF-core functions, API Invoker, API exposing function, API publishing function and API management function that describe a functional architecture which enables an API invoker to access and invoke service APIs. The CAPIF functional model can be adopted by any 3GPP functionality providing service APIs and may be deployed in centralized and distributed fashion.

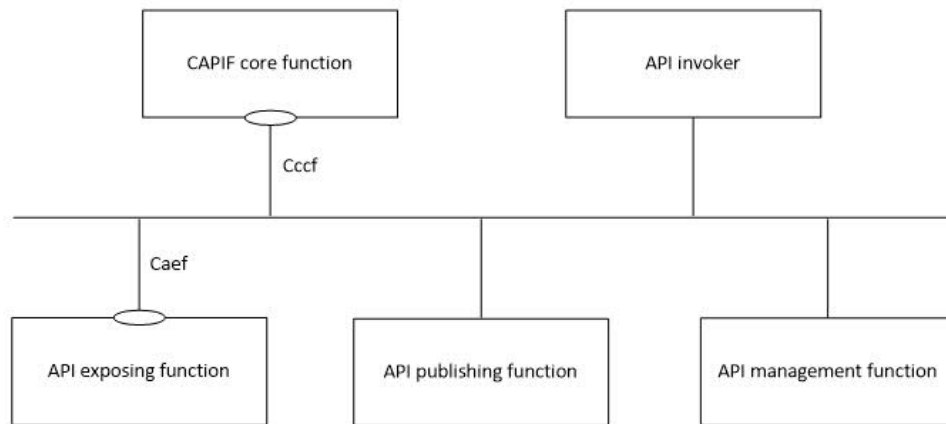


Figure 2.4 CAPIF functional model representation using service-based interfaces

2.4.3.1 API invoker

The API invoker is typically provided by a third-party application provider who has service agreement with PLMN operator. The API invoker may reside within the same trust domain as the PLMN operator network and supports the discovery of the service API and mutual authentication with CAPIF before obtaining authorization to access the service API.

2.4.3.2 CAPIF core function

As the name suggests this entity supports the core functions consisting of authentication and authorization of the API invoker, publishing, storing and supporting the discovery of service APIs information; controlling the service API access based on operator configured policies, storing the logs for the service API invocations and making them available to authorized entities, charging based on the logs of the service API invocations, monitoring the service API invocations, onboarding/offboarding of API invoker, storing policy configurations related to CAPIF and service APIs.

2.4.3.3 API exposing function

The API exposing function is the provider of the service APIs and is also the service communication entry point of the service API to the API invokers. Its functions consist of: authenticating the API invoker based on the identity and other information provided by the CAPIF core function, validating the authorization provided by the CAPIF core function and logging the service API invocations at the CAPIF core function.

2.4.3.4 API publishing function

The API publishing function enables the API provider to publish the service APIs information in order to enable the discovery of service APIs by the API invoker. The API publishing function consists of Publishing the service API information of the API provider to the CAPIF core function.

2.4.3.5 API management function

The API management function enables the API provider to perform administration of the service APIs and consists of the following capabilities: auditing the service API invocation logs received from the CAPIF core function, monitoring the status of the APIs and the events reported by the CAPIF core function, configuring the API provider policies to the CAPIF core function, onboarding the new API invokers and offboarding API invokers and registering and maintaining registration information of the API provider domain functions on the CAPIF core function.

Table 2.1, summarizes the relationship between CAPIF, 5GS and EPS network exposure aspects. The details of SCEF and NEF and their role in exposing network capabilities of 5Gs and EPS to third-party

applications are specified in 3GPP TS 23.501, TS 23.502 and 23.682.

Table 2.1 Relationship between CAPIF, 5GS and EPS network exposure aspects

Aspects	CAPIF	5GS network exposure	EPS network exposure
Entity providing the APIs to external or third-party applications	AEF	NEF	SCEF
Entity providing framework related services to the applications (discovery, authentication, authorization, etc)	CAPIF core function	NEF (Not specified yet)	SCEF
Entity representing the external or third-party applications	API invoker	AF	SCS/AS
Entity providing framework related services to support the APIs operation and management (publish, policy enforcements, charging)	CAPIF core function	NEF (Not specified yet)	SCEF
Interface/Reference point for exposing network capabilities as APIs	CAPIF-2 and CAPIF-2e (Do not include the service specific aspects)	Nnef	T8
Interface/Reference point for exposing framework services as APIs to the applications	CAPIF-1 and CAPIF-1e	Nnef (Not specified yet)	Not specified. (May be via T8)
Interface/Reference point for framework services to support the APIs operation and management	CAPIF-3, CAPIF-4 and CAPIF-5	Internal to NEF	Internal to SCEF

2.4.4 3GPP Network Exposure for Slice management

5G network will support different types of communication services with different requirements using network slicing. A Network Slice Instance (NSI) is a managed entity in the operator's network with a lifecycle independent of the lifecycle of the supported service instance. 3GPP specifications TS 28.530 [3GPP-28530] and TR 28.801 [3GPP-28801] identify the following high-level business roles related to the operation of network slices. Communication Service Customer (CSC) which uses communication services, provided by a Communication Service Provider (CSP). The CSP provides the communication services by designing, building and operating its communication services, with or without network slices, on top of the network service provided by the Network Operator (NOP). And the NOP designs, builds and operates its networks to offer such network services using virtualized infrastructure service provided by Virtualization Infrastructure Service Provider (VISP). The VISP may in its turn use the network function virtualization infrastructure provided by the NFVI supplier and data center resources provided by the Data Center Service Provider (DCSP). The communication service provided by the CSP to its customers include Business to Consumer (B2C) services, such as mobile web browsing, VoLTE, etc. Business to business (B2B) services such as Internet access, LAN interconnection, etc and Business to Business to everything (B2B2X) services offered to other Communication Service Providers, such as international roaming, RAN sharing, etc. or to verticals e.g., eMBB, etc. offering themselves communication services to their own customers. A communication service offered by CSPs can include a bundle of specific B2C, B2B or B2B2X type of services.

According to the above 3GPP specifications network slices can be offered as NOP internals or as Network Slice as a service (NSaaS). In the Network Slices as NOP internals model, network slices are not part of the CSP service offering and hence are not visible to the CSCs. However, in order to provide support to communication services, the NOP may decide to deploy network slices, e.g. for internal

network optimization purposes. This model allows CSC to use the network as the end user or optionally allows CSC to monitor the service status (assurance of the SLA associated with the internally offered network slice). The CSP should be able to provide the service status information (e.g. service performance, fault information, traffic data, etc.) to CSC via the management exposure interface.

In the Network Slice as a Service (NSaaS) model, the CSP may offer a network Slice Instance (NSI) to its CSCs in the form of a service. This service allows CSC to use the network slice instance as the end user or optionally allows CSC to manage the network slice instance as manager via management interface exposed by the CSP. The CSC can in its turn play the role of CSP and offer its own services (e.g. communication services) on top of the network slice instance obtained from the CSP. Depending on service offering, a CSP offering a NSaaS may impose limits on the NSaaS management capabilities exposure to the CSC and the CSC can manage the network slice instance according to NSaaS management capabilities exposed and agreed upon limited level of management by the CSP. Tables Table 2.2 and Table 2.3, from TS 28.530 show high level use cases for Exposure of network slice management data for NSaaS and Exposure of network slice management capability.

Table 2.2 Exposure of network slice management data for network slice as a service case

Use case stage	Evolution/Specification
Goal	To expose network slice management data to a Communication Service Provider (CSP) consuming Network Slice as a Service (NSaaS) based on mutual agreement.
Actors and Roles	A Communication Service Provider (CSP) provides limited management data to a Communication Service Customer (CSC)
Telecom resources	3GPP management system
Assumptions	Network slice management data of NSI can be exposed to the CSP consuming NSaaS according to the pre-defined agreements.
Pre-conditions	1. NSaaS level exposure has been agreed upon and the CSP offering the NSaaS is aware of it. 2. An NSI used for NSaaS is created.
Begins when	The CSP consuming NSaaS wants to get the management data of the network slice instance.
Step 1 (M)	The CSP consuming NSaaS sends requests to the 3GPP management system for the exposure management data of network slice instance.
Step 2 (M)	The 3GPP management system provides the CSP consuming NSaaS of exposed management data for the NSaaS scenario.
Ends when	The network slice management data is provided.
Exceptions	One of the steps identified above fails.
Post-conditions	The CSP consuming NSaaS is aware of the management data of the network slice instance.
Traceability	REQ-3GPPMS-CON-27

Table 2.3 Exposure of network slice management capability

Use case stage	Evolution/Specification
Goal	To expose limited network slice management capability to a Communication Service Customer (CSC) consuming Network Slice as a Service (NSaaS) based on mutual agreement.
Actors and Roles	A Communication Service Provider (CSP) provides limited management capability to a Communication Service Customer (CSC)
Telecom resources	3GPP management system
Assumptions	Network slice management capability of 3GPP management system can be partially exposed to the CSC consuming NSaaS according to the pre-defined agreements.
Pre-conditions	Level of management exposure has been agreed upon and the CSP offering the NSaaS service is aware of it.
Begins when	The CSC consuming NSaaS wants to get certain management capability to manage the network slice instance, e.g., PM, FM, CM, based on the mutual agreement between CSC and CSP.
Step 1 (M)	The CSC consuming NSaaS sends requests to the 3GPP management system for the exposure of management capability of network slice instance.
Step 2 (M)	The 3GPP management system provides the CSC consuming NSaaS with the requested capability via appropriate methods, e.g., exposing network slice management service to the CSC.
Ends when	The network slice management capability is provided.
Exceptions	One of the steps identified above fails.
Post-conditions	The limited network slice management capability has been exposed to the CSC consuming NSaaS.
Traceability	REQ-3GPPMS -CON-30, REQ-3GPPMS -CON-31, REQ-3GPPMS -CON-32

TR 28.801 also identifies the following management functions needed to manage NSIs to support communications services:

- Communication Service Management Function (CSMF) which is responsible for translating the communication service-related requirements to network slice related requirements and to communicate with Network Slice Management Function
- Network Slice Management Function (NSMF): Responsible for management and orchestration of NSI and derive network slice subnet related requirements from network slice related requirements. It also communicates with the Network Slice Subnet Management Function (NSSMF) and CSMF.
- Network Slice Subnet Management Function (NSSMF): which is responsible for management and orchestration of NSSI and to communicate with the NSMF.

As illustrated in Figure 2.5, when the NSI is offered as service to customer, the customer needs to access the management data (performance measurements, alarm information, etc.) related to NSI by the following way:

1. Customer requests to collect the management data related to NSI from Communication Service Management Function (CSMF).
2. Per the request from customer, CSMF requests to collect the management data related to NSI from Network Slice Management Function (NSMF).
3. NSMF collects the management data related to NSI and informs the CSMF about the availability of the management data.
4. CSMF gets the management data from NSMF.
5. CSMF informs the customer about the availability of the management data.

6. Customer gets the management data from CSMF.

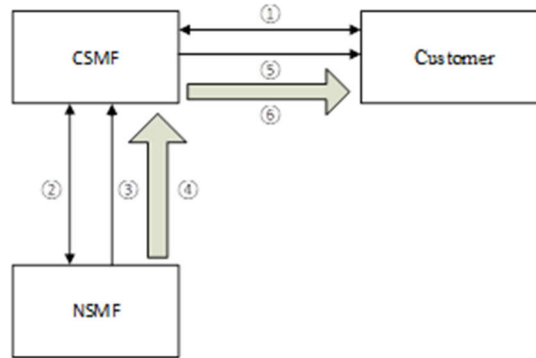


Figure 2.5 NSI related management data exposure to customer

3 Backhaul and MEC

3.1 Overview

The mobile backhaul consists of the infrastructure between the Radio Access Network (RAN) and the mobile Core. In 5G the addition of small cells that are normally installed after the macro cells to extend the coverage or increase the bandwidth when using higher frequencies than macro cells. The small cells require a new transport that connect to the microcell named fronthaul. However, backhaul applies also to backhaul as the transport between RAN (i.e. small and macro cells) and the mobile core. The mobile core as described in the 5G architecture provides the connection between mobile backhaul and data networks (DN) where services and applications accessible from the mobile devices are deployed. The DN can be either public such as Internet or private when connecting the mobile networks to private Local Area Network (LAN).

The Multi-Access Edge Computing (MEC) are running in DN and provide computing capabilities to the services and applications accessible from the mobile devices. It is an evolution of cloud computing that pushes applications from centralized data centres to the network edge near the end-users. MEC is indeed one of the key pillars for meeting the demanding Key Performance Indicators (KPIs) of 5G, especially as far as low latency and bandwidth efficiency are concerned. 5G system provides a set of new functionalities that serves as enablers for edge computing. These enablers are essential for integrated MEC deployments in 5G networks.

MEC allows external applications to run closer to the end devices, thus reducing latency. In this context MEC would support the firefighting se case where video processing to be deployed close to the incident.

3.2 Mobile backhaul and resource management

Generally, IP based networks used as transport for mobile backhaul operate on a best effort, which means that all traffic has equal priority and equal probabilities of being delivered. Similarly, with best effort when a network becomes congested, all traffic has an equal probability of being delayed or dropped in the worst case. Quality of Service (QoS) selects network traffic, prioritizes the traffic according to its relative importance. The QoS can be enforced in the packets at link layer 2 or transport layer 3. The port/node level QoS classification at layer 2 is based on VLAN priority (PRI) field. This means that all tagged traffic is classified based on a VLAN PRI and untagged traffic classified as best effort (BE). The QoS classification at layer 3 can be done using Type of Service (ToS) if IP transport is used or packet tagging if MPLS is used.

The deployment of QoS in mobile backhaul networks is vendor specific. Normally, the usage of QCI is associated with DiffServ based QoS. Thus, the eNB associates each QCI at the radio bearer with different DSCP packet marking at the network site. Thus, the network switches after the eNB, will check every incoming packet for different parameters in the IP header such as source IP address, destination IP address, type of traffic, etc. and assign it to a specific queue based on DSCP class value.

Machine learning can be used to develop a control logic on the network behavior. It uses several flow-level features such as end-to-end latency, packet count, link capacity, network delay, hop count, flow count etc., to classify the traffic.

A Mobile Backhaul Orchestrator (MBO) is designed using SDN to deploy and manage network resources. The SDN controller is integrated with module that implements machine learning algorithms to analyze different flow parameters and identify new routing policies to guarantee QoS requirements for selected traffic going through mobile backhaul. The MBO design includes the following modules 1) Network monitoring to check available resources and 2) Machine Learning engine that will apply different traffic engineering to calculate optimal rules to be set in the network switches though SDN controller to optimize the network resources in the backhaul.

The MBO includes machine learning function that, based on user data collected through LLDP from

SDN switches, will calculate disjoint paths using the Dijkstra algorithm. The MBO takes into use the disjoint path when congestion or link break is detected to deliver reliable and low latency communication for selected users i.e. IMSI/TEIDs. The plot in Figure 3.1 shows that the 5GC architecture integrated with SDN MBO, including NRF and NSSF for network slicing, delivers Ultra Reliable Low Latency Communications despite broken links or network congestion, which is required to support reliable connectivity for the firefighting use cases presented in Section 111.3. In Figure 3.1, the network is congested after 10 seconds which affects considerably to the best effort slice compared to the URLLC slice which maintains the delay regardless of congestion.

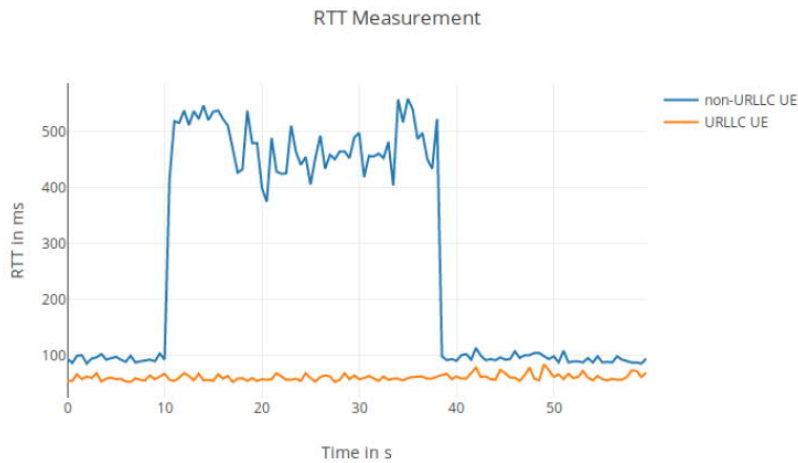


Figure 3.1. Round Trip Time (RTT) results of network congestion impact to URLLC traffic with MBO managed slices.

3.3 MEC optimal placement

An MEC platform is normally deployed in the DN after the mobile core but to reduce latency it could be located right after the RAN in the backhaul. The MBO based on the delays monitored from the backhaul can determine the optimal placement of the MEC as shown in Figure 3.2.

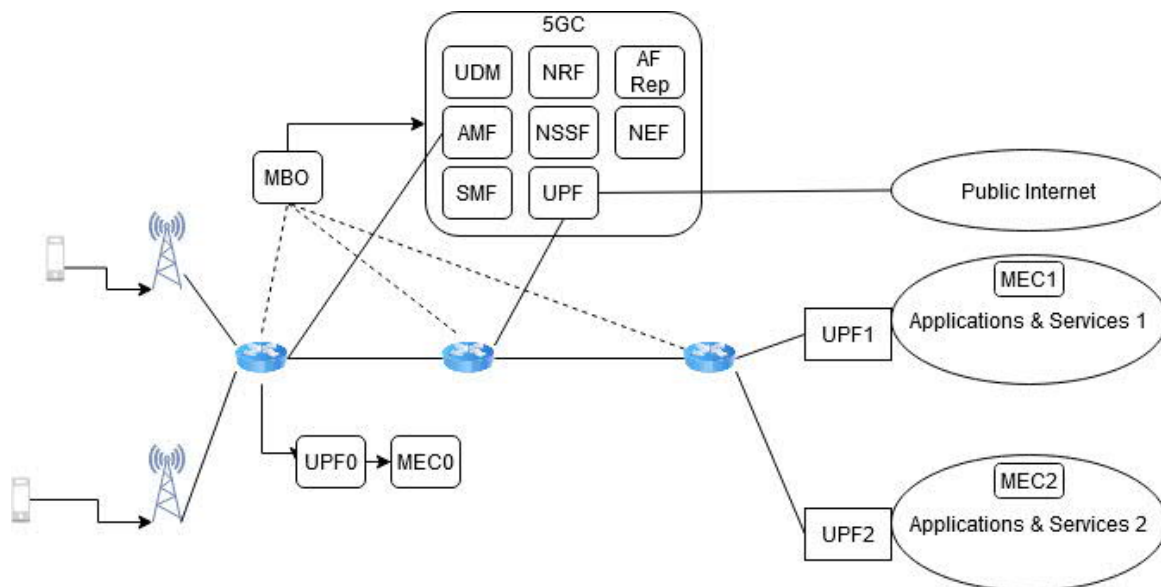


Figure 3.2 MEC platform deployment

3GPP has defined the Network Exposure Function (NEF) to expose the services and capabilities of mobile networks to external applications. Thus, NEF could be used to expose MEC capabilities to external applications. The NEF exposes a set of interfaces so applications can request network resources to be deployed as MEC applications inside the mobile network [3GPP-29522]. 3GPP has defined NEF Northbound interface is between the NEF and the Application Function (AF). This Northbound interface consists of a RESTful APIs so the AF can access the services and capabilities provided by 3GPP mobile network.

In this case the AF could use an existing procedure in the NEF Northbound interface named “Procedure for Traffic Influence”. This procedure allows the AF to request specific traffic handling for selected UE or group of UEs, which in this case would consist of running MEC application for those UEs.

To initiate the process, the AF shall send an HTTP POST message to the NEF with the resource “Traffic Influence Subscription”. According to 3GPP [3GPP-29522] the body of the HTTP POST message may include the AF Service Identifier, external Group Identifier, external Identifier, any UE Indication, the UE IP address, GPSI, DNN, S-NSSAI, Application Identifier or traffic filtering information, Subscribed Event, Notification destination address, a list of geographic zone identifier(s), AF Transaction Identifier, a list of DNAI(s), routing profile ID(s) or N6 traffic routing information, Indication of application relocation possibility, type of notifications, Temporal and spatial validity conditions. The Notification destination address shall be included if the Subscribed Event is included in the HTTP request message.

The AF can request from the NEF to instantiate a Traffic Influence for the UE(s) and in case of requesting MEC support the AF should specify the required CPU or memory requirements for running the AF and the expected round trip delay. The NEF will communicate with the mobile network management functions e.g. MBO to determine whether has MEC capabilities and network resources to fulfil the requirements. If the NEF responds that successfully can fulfil the AF requirements it will return the repository where the AF can be uploaded, and the DNS name associated to the AF. There are still open items such as the format of the AF to run in the virtualized platform provided by the mobile network.

Figure 3.3 shows the sequence flow how the AF can request the support for MEC capabilities to run for selected UE(s).

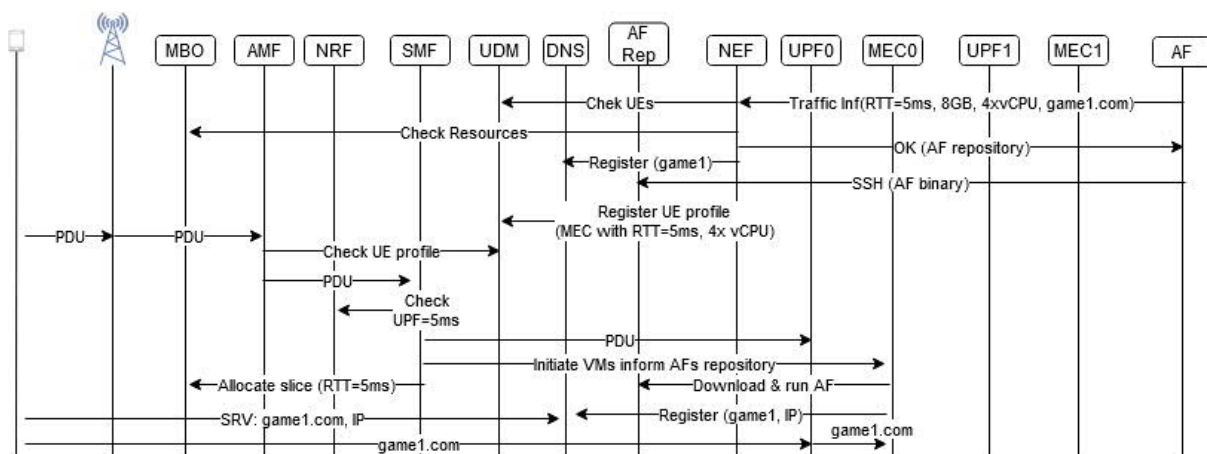


Figure 3.3 Sequence flow of “Procedure for Traffic Influence” used by AF to request support for MEC capabilities.

The MBO includes a monitoring system to check the available network resources and an AI/ML module that can dynamically modify the routing policies of the policies in the backhaul [MOH19].

The NEF provides an interface to the AF to request network resources and indicate the expected performance in terms of delay and bandwidth. The NEF will communicate those requirements to the MBO and PCRF that will confirm whether they can be fulfilled. If the AF requirements can be fulfilled, the NEF will trigger the process to allocate the necessary computing i.e. MEC and networking

resources to deploy and run the AF in the specific location where the requested requirements can be fulfilled.

Besides NEF, 5G architecture has defined the NWDAF which, as indicated in previous the section, collects analytics which can be accessible for other functions. Thus, the NWDAF collects information about the network resources which can be used by AI/ML applications to subscribe to receive information on the load level of a network slice. The NWDAF can be integrated with the MBO to deliver slice information to external applications to perform analysis for optimal usage of available network slices. The MBO will hide all the low-level physical network specifics and will provide NWDAF the operator specified information that AI/ML applications can utilize for their analysis. The MBO should also integrate a MDAF that will allow the AI/ML applications to interact with the low-level network management to request allocation of additional resources to meet their requirements.

4 Radio Access Network (RAN) Slicing

4.1 Overview

The PriMO-5G firefighting use cases described in Section 1.3 noted that in such emergency situations there are various types of communication services with disparate requirements, and they have to be allocated resources commensurate with individual needs of each service. For instance, command messages from the incident commander to drones and robots require URLLC, whereas, HD video transfer requires higher bandwidth and low latency, but is more tolerant to certain delay compared to the command messages. Network slicing may play a key role in order to satisfy different requirements for the communication services during firefighting event. To that end, network slicing can be used to create slices cater to different service functionality requirements (e.g. priority, policy control, security, and mobility), heterogeneous service performance requirements (e.g. throughput, latency, availability, reliability etc.), or vary performance or priorities according user type (e.g. firefighting robots, drones, firefighters, ground control, general public etc.). The PriMO-5G has formulated a use case (see Use Case A2 in Section 1.3) that seeks to further investigate and validate the potential of network slicing in a firefighting context.

In the 3GPP 5G system architecture specifications, the network slicing can be carried out end-to-end, implying that a network slice could be able to provide the functionality of a complete network, that is the Next Generation Radio Access Network (NG RAN) functions and 5G Core Network (5GC) functions (and possibly transport network functions) [3GPP-23501]. This includes specification of network slice identifiers and procedures or network slice provisioning for the different segments and across different operator domains.

In the case of NG-RAN slicing, 3GPP has specified the following key principles for support of network slicing [3GPP-38300]:

- RAN awareness of slices
- Selection of RAN part of the network slice
- Resource management between slices
- Support of QoS
- RAN selection of CN entity
- Resource isolation between slices
- Access control
- Slice Availability (some slices may be available only in part of the network).
- Support for UE associating with multiple network slices simultaneously
- Granularity of slice awareness
- Validation of the UE rights to access a network slice

4.2 RAN slice definitions

Several proposals for the definition of slices (including RAN slices) have appeared in both standards and research literature. In subsections we provide a brief overview of prevalent definitions driven from standardization.

4.2.1 3GPP

In 3GPP, each network slice is uniquely identified by a S-NSSAI (Single Network Slice Selection Assistance Information) [3GPP-23501].

NSSAI (Network Slice Selection Assistance Information) includes one or a list of S-NSSAIs (up to 8 S-NSSAIs supported) where a S-NSSAI is a combination of:

- Mandatory SST (Slice/Service Type) field, which identifies the slice type (see Table 4.1)
- Optional SD (Slice Differentiator) field, which allows for differentiation among slices with same SST value.

Table 4.1 Values standardised by 3GPP TS 23501 [3GPP-23501]

Slice/Service type	SST value	Characteristics
eMBB	1	Slice suitable for the handling of 5G enhanced Mobile Broadband.
URLLC	2	Slice suitable for the handling of ultra-reliable low latency communications.
MIoT	3	Slice suitable for the handling of massive IoT.
V2X	4	Slice suitable for the handling of vehicle-to-everything (V2X) connectivity services.

The standardized SST values of Table 4.1 provide a way for ensuring global interoperability for slicing so that different PLMNs can support the roaming use cases more efficiently for the most commonly used SSTs. An S-NSSAI may also utilize non-standard SST values (used together with SD value or alone) when used to identify a network slice only confined within a single PLMN. The combination of SST and SD values provides flexibility for instantiation of multiple network slices to according to different service types and their specific requirements and priorities.

The SST values are currently defined in general for all slice subnets (CN, RAN, transport etc.). The separate definition of differentiators for RAN slicing still flexible. For instance, some of the considered approaches for RAN slice differentiation include some [SAM20]:

- Slice-aware selection of RAN NFs and CN NFs (e.g. UPF, AMF) that impact RAN performance
- Flexible control of common radio resources (typically Physical Resource Blocks or PRBs) for different slices through dynamic scheduling etc.
- Control of priorities across different RAN slices
- Slice-aware partitioning of RAN according to traffic separation needs of certain RAN slices

4.2.2 GSMA

A GSMA initiative on slice definition is motivated by need to enhance the commercial viability of network slicing. GSMA defines the Generic Network Slice Template (GST) as a set of *attributes* that can be used to characterise a type of network slice [GSMA2019]. A GST is generic and is not tied to any specific network deployment.

- GST attributes derived from different 3GPP specifications
 - Mandatory – the attribute's value must be present
 - Conditional – the attribute's value is mandatory if a certain condition exists
 - Optional – the attribute's value doesn't need to be present

- The NEtwork Slice Type (NEST) is a GST filled with attribute values.

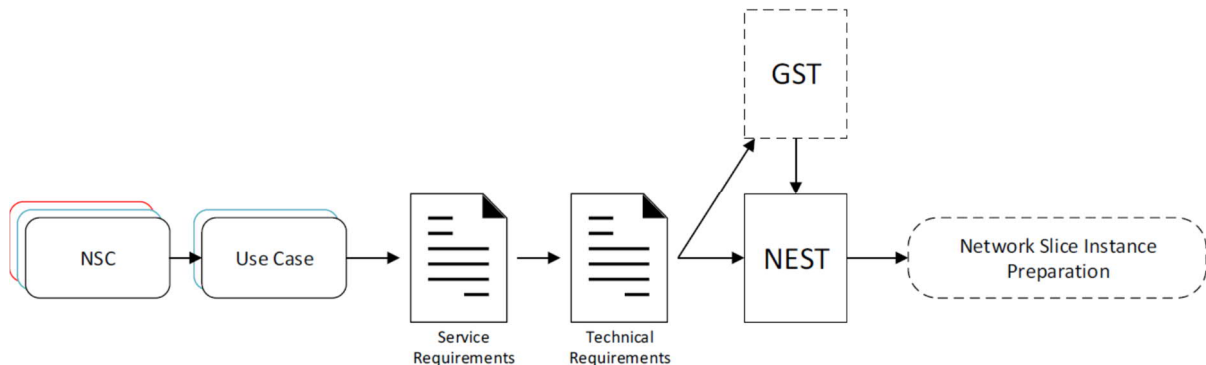


Figure 4.1 GST and NEST in context of the network slice lifecycle (NSC = Network Slice Customer) [GSMA2019]

The GST attributes can either a character attribute or scalability attribute, but not both:

- Character attributes - characterize a slice (e.g. throughput, latency, (APIs), etc.) and are independent of the Network Slice Customer (NSC) and the Network Slice Provider (NSP).
- Scalability attributes - provide information about the scalability of the slice (e.g. number of terminals, etc.) and are specific for the NSC and the NSP.

Character attributes can be further tagged. Tags are used as labels attached to the character attributes to give additional information about the nature of each attribute.

- Performance related tags (KPIs supported by a slice)
- Functional related tags (e.g. prediction, positioning)
- Operational related tags (means provided to the NSC in order to operate the slice)

3GPP (SA2#134 meeting) recently agreed a new study item for Rel.17 [S2-1908583] to identify the gaps in the currently defined 5GS system procedures defined in 3GPP to support of GST attributes and to study potential solutions that may address these gaps.

4.3 RAN Slicing over RAN architecture

As defined by the NGMN and the ORAN alliance, the RAN could be split to three entities of RU (or O-RU as defined by ORAN), DU (or O-DU) and the CU, which then will interface as seen in Figure 4.2 . How this split is shaped over functionalities of RAN could bring a new definition to the network slices that are now over RAN architecture. Through experimental analysis, we have demonstrated that different split can benefit different 5G use case [MCM+17]. With the advances in for example microservices deployment, there is a possibility of having higher granularity of slices through architecture, i.e. multiple slices that split the RAN differently across RU, DU and CU while operating along each other.

The resource sharing for slicing within the disaggregated RAN architecture is an important aspect that we elaborate further in the following.

4.3.1 Flexible Function Split as RAN slicing enabler

There have been eight different split options foreseen for RAN by the 3GPP, offering separation of RU at different level of protocol stack. In general, the lower the split point, the greater the level of

centralization, the higher is the required interface data rate and the more stringent is the latency requirement on the fronthaul. On the other, the lower layer splits can serve applications with higher latency demand. This trade-off motivates functionality split to be part of the RAN slices.

Hence, to flexibly meet different requirements of 5G services, different split between CU and DU can be configured. In this context, each functional split may belong to a slice. Accordingly, the requirements provided by functional split can be guaranteed by the slice, thus features provided differ from slice to slice. A specific type of service is dedicated to an appropriate slice according to the application need. Each slice might be assigned a set of resources to guarantee slice isolation. Nevertheless, slicing should allow efficient resource utilization. In general, there is a trade-off between isolation and effective radio resource utilization.

Figure 4.2 shows an example of functional split deploying RAN slicing. For instance, smart cars will require low latency communication while smart cities would typically require high bandwidth and as such different slices may be configured to offer different requirements for latency and FH throughput helping to serve different types of services. This architecture provides Cloud-RAN the flexibility needed to support the diverse requirements of 5G services.

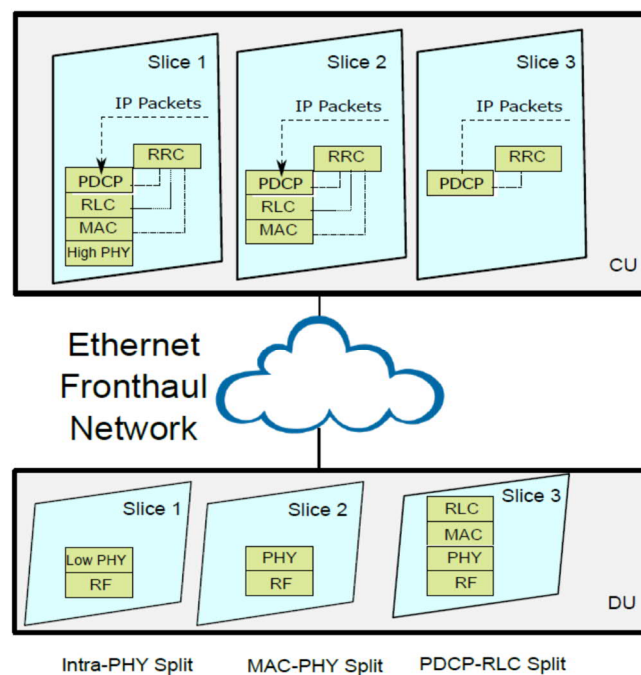


Figure 4.2 Example of envisioned architecture of RAN functional split supporting RAN slicing

4.3.2 Slicing over multiple path fronthaul

Fronthaul resource sharing can be categorized into two primary categories: non-orthogonal sharing and orthogonal sharing. In the formal, all FH resources are shared among the services. In the case of orthogonal sharing, a dedicated amount of resources is allocated to services. For instance, a percentage of the bandwidth or a percentage of available path is allocated to a service to guaranty the required KPIs. This type of orthogonality enables the service slicing in the coexistence of the heterogeneous service to offer the requirement needed according to the specific service. The traffic is

steered to the appropriate slice using for example the concept of identification. The packets, hence, are enqueued into the specific fronthaul path associated to a slice identity. Each slice offers a set of reserved KPIs such as delay and packet loss.

4.3.2.1 Simulation Analysis of multiple path fronthaul

The multiple path fronthaul is simulated in both cases of orthogonal and non-orthogonal sharing and with different slices having different latency-reliability trade-off. Hence, traffic over fronthaul is sent repeatedly over multiple path for improved reliability, and also encoded and sent over multiple path for improved reliability with less compromising on the latency. The slices are considered to serve URLLC and eMBB traffic.

The simulation model consists of Cloud-RAN with a single CU and a single DU connected with multiple fronthaul paths (n different paths), where each path i has a capacity C_i . Packets of size B bits are arrived to the system with exponential inter-arrival periods. These parameters are summarized in Table 4.2. We assume that the fronthaul links are identical. Each link is modelled as a single queue with exponential service.

Table 4.2 Fronthaul link and traffic parameters for URLLC and eMBB

Parameters	Values
Ethernet capacity	100 Mbps
N	10
Packet arrival rates (packet/ms)	8 eMBB, 24 URLLC
IP packet size (Bytes)	1500 eMBB, 500 URLLC

Three schemes are compared with each other: multiple path with duplication (MPD), multiple path with coding (MPC) in which fountain coding is used and receiving k out of n packets are sufficient for decoding of the source packet, and single path transmission (SP). These three are then compared with the case of no slicing and simple sharing the bandwidth.

Figure 4.3 shows the probability of error as a function of the latency when orthogonal bandwidth allocation with four fifth of the bandwidth allocated to URLLC and one fifth to eMBB. MPC with $k = 2$ can reduce latency by 0.3 ms in eMBB and 0.0075 ms in URLLC, with respect to MPD at the error probability of 10^{-5} . Furthermore, dedicating a large bandwidth to URLLC can significantly enhance the probability of error as compared to shared FH, but this happens at the cost of a larger latency for the eMBB service.

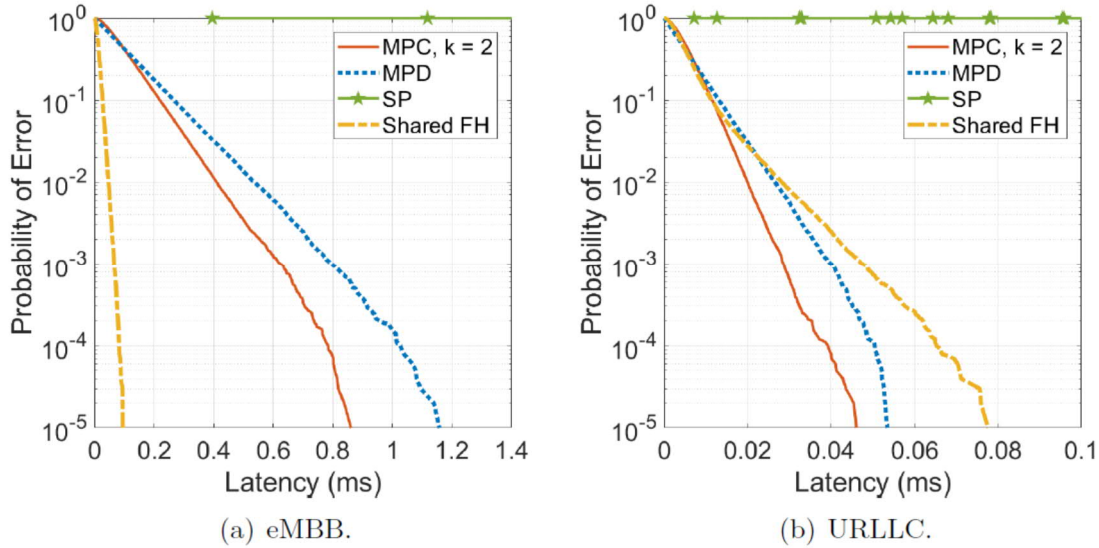


Figure 4.3 Probability of Error in orthogonal sharing, in the case of single path (SP), multiple path with duplication (MPD) and multiple path with encoding (MPC).

Figure 4.4 shows the probability of error obtained with path split where eight out of ten paths are allocated to the URLLC. For the URLLC, MPC can manage to reduce latency by 0.023 ms as compared to the MPD at the error probability of 10^{-5} . Moreover, the probability of error of path split is improved by 60% as compared to shared fronthaul.

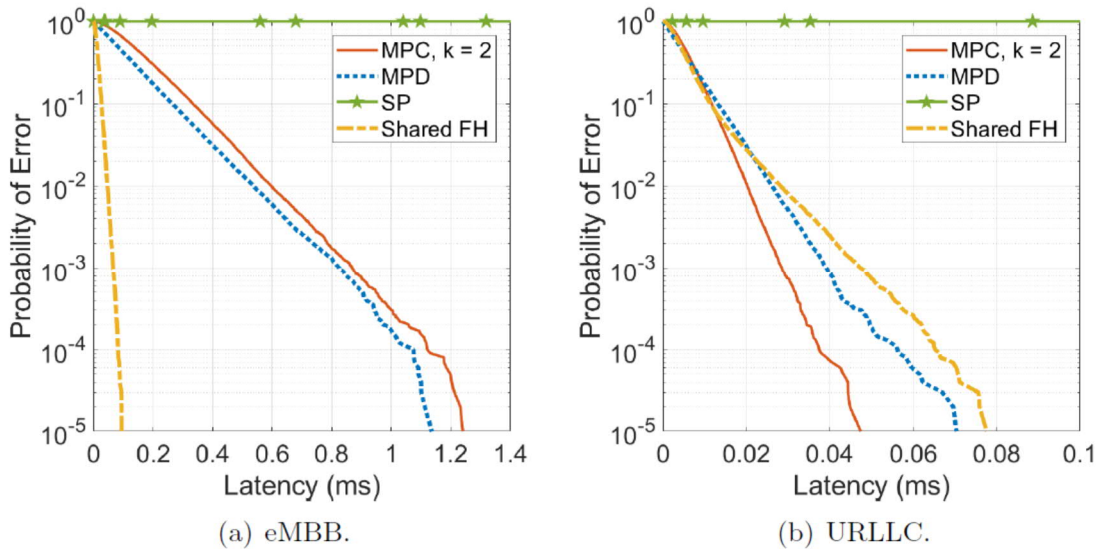


Figure 4.4 Probability of Error in non-orthogonal sharing, in the case of single path (SP), multiple path with duplication (MPD) and multiple path with encoding (MPC).

In summary, by slicing the fronthaul resources adequately via orthogonal allocation transmission, the error probability can be reduced compared to non-orthogonal fronthaul allocation. It can also be

observed that the error probability is improved in the case of MPC as compared to MPD but with the cost of extra processing, which could be justified for the slices with the need of lower latency.

4.4 Open APIs for RAN slice preparation and lifecycle management

In a typical network an operator may need to flexibly support large number of networks slices, which underlines the need for automation throughout the processes of slice preparation and lifecycle management [3GPP-28530]. In the preparation phase, slices templates or definitions that meet customer requirements may be created from an operator's existing catalogue templates, and thereafter slice creation and activation is commenced. Subsequent process in slice lifecycle management include the supervision, performance reporting, modification, de-activation and termination of a network slice instance. An emerging scenario is of these processes of slice preparation and other slice lifecycle management aspects being handled by the customer, external applications or third parties via appropriate open APIs. In the section we review some of the open APIs that are specified for RAN slice preparation and lifecycle management.

4.4.1 O-RAN Alliance

The O-RAN Alliance is an operator-led industry alliance that constitutes diverse range of members and contributors from the telecommunications ecosystem. This includes telecom vendors with an objective of reducing vendor lock-in, enhancing interoperability, increasing automation and boosting innovation in the RANs by leveraging virtualization, standardized interfaces and minimizing dependence on proprietary hardware through use of white-box hardware [ORAN19].

To that end, the O-RAN alliance is driven by two core-principles:

1. *Openness*: Specifying open interfaces in the RAN to enable operators and third parties to produce more agile and innovative services or easily customize the RAN to suit unique needs of particular use cases or deployment scenarios. Furthermore, open interfaces allow operators to have multi-vendor deployments in the RAN that mixes best-of-breed approach. Moreover, the open interfaces bring economy of scale benefits of the cloud to the RAN through resource pooling, allowing multiple services or customers to be served in multi-tenant model with different physical and virtual resources (e.g. radio resources, processor pools etc.) dynamically assigned and reassigned according to demand.
2. *Intelligence*: Increase intelligence in the deployment, optimization and operation of increasingly complex RANs with advent of 5G NR, increased densification and more diverse use cases and deployment scenarios. This intelligence is enabled by gradual shift from human-intensive approaches towards more automated approaches with emerging AI/ML-based solutions, at both component and network levels. The combination of these approaches with open standardized southbound interfaces provides a wide new range of possibilities of network automation and programmability, particularly for network slicing preparation and lifecycle management.

4.4.1.1 O-RAN reference architecture

In pursuing the aforementioned vision of future RAN infrastructure, the O-RAN Alliance has specified an O-RAN Reference Architecture (see Figure 4.5) that leverages the principles of openness and intelligence[ref]. The reference architecture is considered to be complementary to the standards specified by 3GPP, as well as other standards development organizations (SDOs) and industry alliances. The controller functionality (radio intelligent controller or RIC) introduced in O-RAN architecture is split between non-Real Time (non-RT) and near-Real Time (near-RT) control functions that occur in control loops longer than 1s and shorter than 1s (typically 10ms-1s), respectively. The description of these controllers and their new interfaces are summarized below:

- a) *RAN Intelligent Controller (RIC) non-Real Time (non-RT) layer*: The Non-RT RIC resides in the network player that provides orchestration and management, and its control functions include service and policy management, as well as, RAN analytics and model-training for the underlying Near-RT RIC functions. The non-RT functions rely on the data gathered by the Near-RT RIC layer from the underlying RAN infrastructure. The O-RAN architecture introduces a standardized A1 interface to facilitate the aforementioned interaction between non-RT RIC and near-RT RIC functions.
- b) *RAN Intelligent Controller (RIC) near-Real Time (near-RT) layer*: This is a controller layer introduced within the gNB to enable enhanced radio resource management (RRM) that may replace or compliment legacy RRM. The enhanced RRM may improve existing operational aspects (e.g. load balancing, interference mitigation etc.) but also provide new functionality enabled by embedded intelligence including the flexibility to meet needs of customers or third-party applications interacting with the controller in scalable and secure manner (while also being guided by policies from non-RT RIC). Near-RT RIC functions utilize a database referred to as the Radio-Network Information Base (R-NIB) that captures the near real-time state of the underlying network. Moreover, a standardized open interface is introduced, namely, the E2 interface between the near-RT RIC layer and the underlying central unit (CU) and distributed unit (DU) protocol stacks. The interface originates from the interface between RRM and RRC in legacy systems, and it provides the means for sending RRM configuration commands from the Near-RT RIC to the CU/DU and collecting measurement data from RAN infrastructure to the R-NIB.

Additionally, the O-RAN Reference Architecture introduces multiple radio access technology (multi-RAT) protocol stack for CU to enable control across different RATs, both 3GPP (4G, 5G) and non-3GPP. Moreover, O-RAN Alliance is further working on providing a detailed specification of Open Fronthaul Interface between the DU and radio units (RU), to ensure realization of multi-vendor interoperability, through interworking of RUs and distributed units DUs from different vendors.

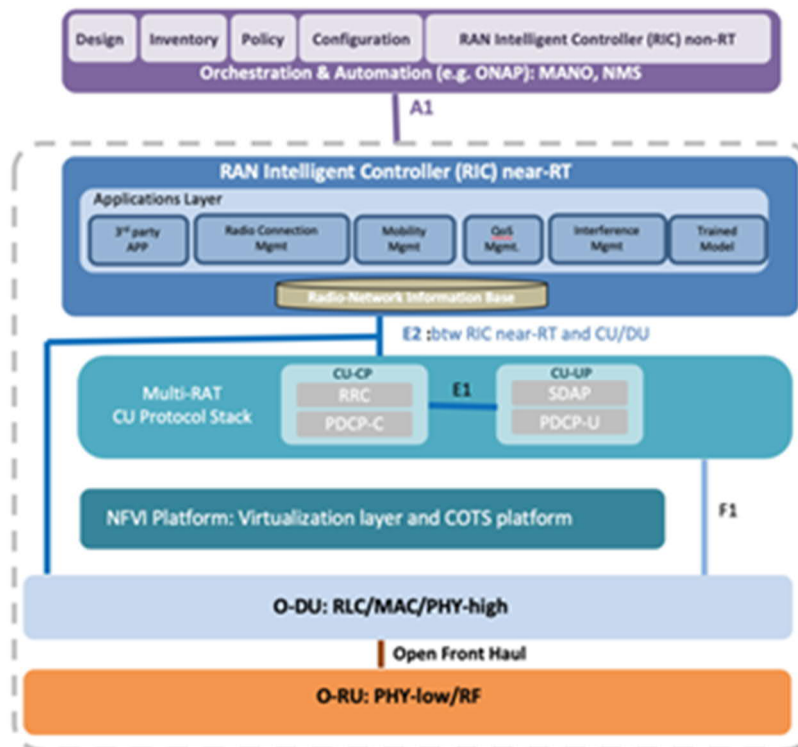


Figure 4.5. O-RAN Reference Architecture (source: O-RAN)

4.4.1.2 O-RAN exemplary use cases

The O-RAN Alliance is currently studying a set of exemplary use cases and deployment scenarios to illustrate the benefits of the openness and intelligence introduced by the O-RAN architecture [ORAN20]. A selection of these use cases that are also of close interest to the PriMO-5G use cases are summarized below.

- a. *RAN slice SLA assurance*: In RAN slicing there is a need for automated mechanisms for continuously ensuring slice Service Level Agreements (SLAs) between mobile operator and business customers or end users, to prevent its possible violations. The O-RAN's standardized open interfaces and intelligent controllers can facilitate these challenging RAN SLA assurance mechanisms. The Non-RT RIC and Near-RT RIC enable dynamic fine-tuning RAN behavior to assure RAN slice SLAs according on RAN-specific slice SLA requirements (e.g. latency, data rate, service availability etc.). Specifically, the Non-RT RIC monitors long-term trends and patterns in the RAN and trains learning models and policies to be utilized by the Near-RT RIC for slice SLA assurance.
- b. *Flight path based dynamic drone resource allocation*: 5G NR is considered a suitable candidate for connecting drones flying at a low altitude, replacing traditional point-to-point connections between the drone and ground control stations. However, the deployed 5G networks were not originally optimized for providing continuous coverage for aerial devices. In an O-RAN architecture, the Non-RT RIC may collect data from RAN measurements, as well as, flight related data (e.g. on no-fly zones), weather data and so on, and use this data to train learning models. The models will be utilized by the Near-RT RIC to perform radio resource allocation for on-demand coverage of the drones whilst taking into consideration flight path information, radio channel conditions and so on.
- c. *Radio resource allocation for drone applications*: Rotor winged drones with mounted cameras and sensors have found many useful applications in monitoring and surveillance use cases. In a typical 5G connected drone scenario, multiple user plane data traffic streams are produced, including uplink high-definition video and sensor data, and uplink/downlink control data. This mixed traffic raises requirements for optimizing resource allocation of the gNB according to individual stream/terminal needs. The O-RAN architecture enables Near-RT RIC to translate radio resource requirements for different terminals in configuration commands for CU/DU to meet individual needs.

4.4.2 Small Cells Forum 5G FAPI

The Small Cells Forum (SCF) is an industry alliance that focuses on development of technical and commercial enablers for accelerating adoption of small cells and driving wide-scale dense deployment of small cells in enterprises, industries, urban and rural scenarios [SCF20]. To that end, the SCF defines a small cell as “a radio access point with low RF power output, footprint and range. It is operator-controlled, and can be deployed indoors or outdoors, and in licensed, shared or unlicensed spectrum. Small cells complement the macro network to improve coverage, add targeted capacity, and support new services and user experiences. There are various types of small cell, with varying range, power level and form factor, according to use case.” The SCF has driven the standardization of APIs and key elements of small cell technology, providing solutions such as:

- Disaggregation of 5G small cells
- Planning, management and automation of small cells
- Neutral host and multi-operator small cell deployments
- Coexistence of private and public networks
- End-to-end orchestration
- Edge computing with small cells

These solutions enable small cells to be deployed as open multivendor platforms boosting innovation

within the ecosystem and lowering barriers to densification for all possible stakeholders (not just traditional operators).

In the PriMO-5G firefighting scenarios of deliverable D1.1, small cells play a key role in providing rapid on-demand coverage in firefighting scenes. This includes following deployments in rural or urban settings:

- I. Stationary small cells deployed temporarily e.g. on fire trucks
- II. Nomadic small cells carried on backpacks of firefighters
- III. Aerial small cells carried on firefighting drones
- IV. Permanently deployed small cells part of existing commercial or public safety networks

The control and programmability of these small cell platforms is useful in configuring or optimising the local RAN to meet continuously changing service needs in a firefighting context. To that end, in this section the functional application platform interface (FAPI) initiative of the SCF is reviewed in brief.

The FAPI initiative targets to specify common APIs for small cells to enable interoperability and enhance innovation among suppliers of small cell platform software, hardware and applications. Through these APIs, different solution vendors are able to have lowered engineering barrier for providing new innovations in software and silicon hardware for small cells platforms from different vendors. In the case of 5G small cells, the FAPI initiative provides three main API specifications whose interfaces are within a gNB as shown in Figure 4.6. These specifications are summarised below.

1. *5G FAPI PHY API (interfaces P5 and P7)* [SCF-PHY]: Provides interfaces for exchange of control-plane and user-plane (data-plane) information between L2/L3 application software and L1 hardware platforms. As noted in the example of Figure 4.6, different L2/L3 protocol layers will interact with the PHY API, whereby, PHY control entity provides configuration procedures via P5 interface, whilst, the MAC layer allows for exchange of user-plane messages via the P7 interface. In this example, a PHY control entity is responsible for configuration procedures (P5). The MAC layer is responsible for the exchange of data-plane messages with the PHY (P7).
2. *5G FAPI RF and Digital Front End Control API (interface P19)* [SCF-FAPI]: In the 5G FAPI initiative, the frontend unit (FEU) is a logical concept that constitutes the RF, digital front end (DFE) and the analog beamforming (ABF) blocks. The shortened TTI and symbol durations in 5G necessitated tighter control of the FEU by L2/L3 software. Hence the RF and DFE Control API was introduced via the P19 interface dedicated control and configuration of the FEU (separate but complementary to the P5/P7 interfaces of the PHY API). In addition to enabling fast dynamic control of the FEU, the P19 interface also allows for disaggregation of the FEU blocks and their interoperability when provided by different vendors.
3. *Network Monitor Mode API (interface P4)* [SCF-API]: The 5G FAPI Network Monitor Mode (NMM) API is specified to utilize the P4 interface to configure and operate network monitoring functions within the capability of the PHY. Specifically, the NMM specification defines the procedures, messages, and structures to implement NMM functions for 5G NR as well as legacy pre-5G RATs. The NMM enables small cells to listen to the radio environment within the close vicinity of the deployed small cell to support Self Organizing Network (SON) functions for optimizing the radio parameters of the small cell accordingly, thus reducing planning overhead, minimizing interference, enable dynamic spectrum sharing and so on. This network monitoring or listening capability involves the small cell acting as passive UE receiver that is able to search and decode system parameters from nearby cells (5G or earlier RATs).

Small cell internal architecture

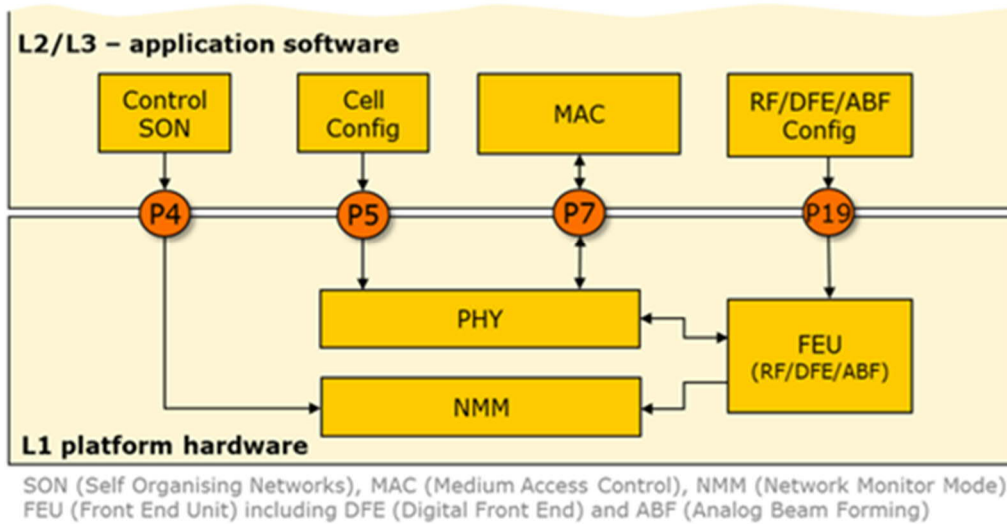


Figure 4.6 Small cell internal architecture

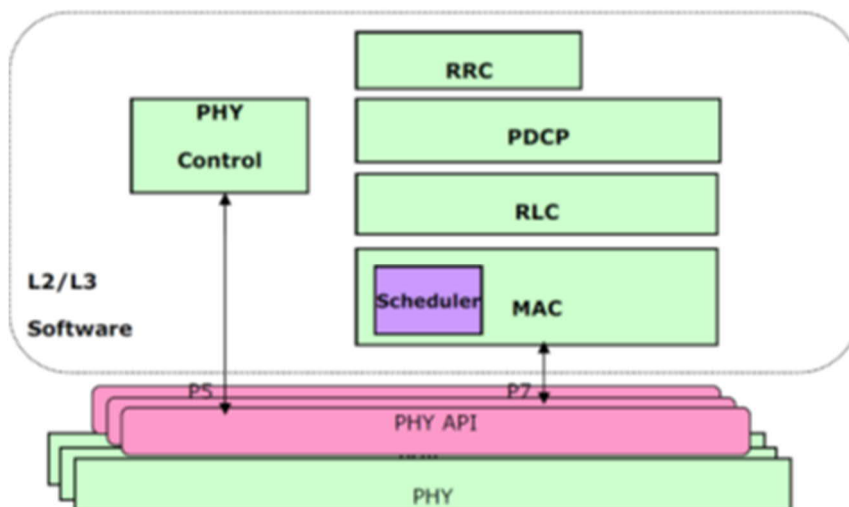


Figure 4.7. Example interactions via PHY API

4.4.3 Other open APIs for RAN control

Apart from the industry driven frameworks (from O-RAN and SCF), there have been few other notable frameworks driven more from the research community. Two of those are described briefly in this subsection.

4.4.3.1 FlexRAN

FlexRAN is an open-source implementation by the Mosaic5G consortium [MOS5G20] of a flexible implementation of a flexible and programmable platform for software-defined RANs. The FlexRAN framework includes two main parts: the FlexRAN Service and Control Plane and FlexRAN Application plane (see Figure 4.8). The latter plane has a hierarchical design that includes RAN controller (Realtime Controller or RTC) connected to number of underlying RAN runtime modules (monolithic or

disaggregated eNB/gNBs), with communications between those modules and the controller facilitated by a FlexRAN protocol. Third-party RAN control applications can be developed both on the top of the RAN runtime and controller SDK facilitating the monitoring, control and coordination of the state of underlying RAN. Similar API exposure is provided for produced RAN data to be consumed by third parties to enable more dynamic and automated RAN control.

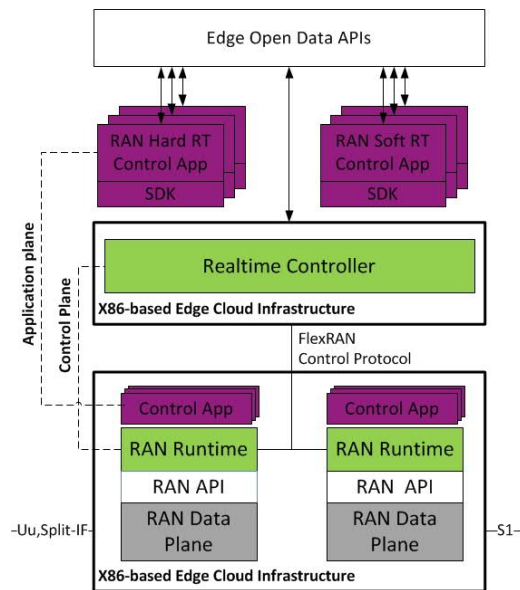


Figure 4.8 FlexRAN framework from [MOS5G20]

4.4.3.2 5G-EmPOWER

5G-EmPOWER is a centralized software-defined RAN controller developed by the Fondazione Bruno Kessler (FBK) [5GEMP20]. The software-defined RAN controller is implemented as an extension of the 5G-EmPOWER Operating System (OS) (see Figure 4.9). The RAN Controller provides a RESTful API for the provisioning of slices and the so-called 5G-EmPOWER Northbound API for executing RAN slice control and management applications (e.g. slice-aware scheduling, admission control etc.) on top of the controller. The RAN controller interacts with the underlying RAN functions via distributed 5G-EmPOWER agents embedded within the RAN. At the time of writing, the 5G-EmPOWER framework had been tested with 4G experimental software-radio stacks, Wi-Fi and LoRA.

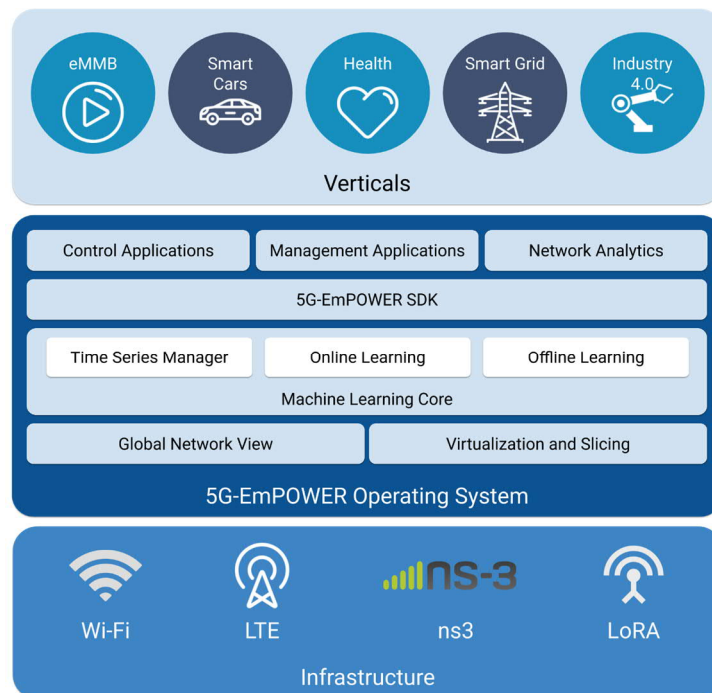


Figure 4.9 5G-EmPOWER framework from [5GEMP20]

4.5 Dynamic RAN slicing

It is recalled that network slicing is motivated by the idea of providing dedicated resources and services for particular service groups. The state-of-the-art network slicing applications rely on very well-defined service level agreements that details the extent and the coverage of the resource ownership as well as providing the technical and economical responsibilities of each parties involved. However, such static applications require the network operators to be perfectly aware of the upcoming demand on their network resources as well as the fluctuations in the daily or hourly demand and its short- and long-term effects on their business. Despite the development of the high efficiency prediction algorithms, predicting the relatively longer-term demand is still a major challenge. Moreover, in the context of public safety communications, which requires not only extremely low latencies but also high throughput, predicting the upcoming emergencies is an unrealistically challenging task (as exemplified by PriMO-5G firefighting use cases). Therefore, the coexistence of public safety communications with commercial 5G services requires an extremely high level of flexibility and efficiency.

Another key challenge that is usually underestimated by the academical research is the economic effects of this technical models. Since the transition from a voice dominated business model to data and service dominated business model, the profitability of network provisioning is under a rapidly increasing economic pressure. Despite the non-standalone application opportunity in 5G, still an excessive deployment of network resources is needed to ensure the successful transition to 5G era. Without a clear return on investment as well as the traffic projection in beyond 5G networks and services, the required excessive deployment cost is challenging the network operators and risking a monopolization of the broadband market by a few rich network operators. In order to prevent this monopolization, the regulatory authorities have been considering some possible solutions including the city owned network resources that can be used leased to virtual network operators (i.e. tenants) to serve their customers. Despite these regulative measures, without a clear economic model that can outline the short and long term tecno-economic aspects as well as the extend of ownership of resources, the virtual network operations are unclear business models. The network slicing, similar to handle the

coexistence of different services, can be used to provide virtual networks for different tenants which can determine the different aspects of the network resources based on their long-term goals and strategies. Such a coexistence model for different tenants can provide a relatively simpler business platform where both newcomers as well as well-established operators can compete and provide their services. Similar to the coexistence of different 5G services, providing a flexible and efficient sharing platform is also critical.

The dynamic network slicing can provide the required flexibility and efficiency to the network operators. Unlike the static network slicing where the resources are (almost) permanently allocated to the different slices, the dynamic slicing suggests the allocation of physical resources for short periods of time. The extend of the time period strictly depends on the needed level of flexibility in the region. Another key aspect is the emergency traffic. Public safety communications requires immediate availability of the spectral and computational resources. Consequently, the increased time periods between reallocation of the resources can massively impact the response time. Moreover, in public safety communications, the different tasks become active at the different time. For example, in the PriMO-5G context, the network traffic is divided into preparatory actions and the supportive actions. Although these communication requests require fundamentally different functions and resources, the duration of each process is not fix. Therefore, the network management solution needs to be prepared to adapt and reallocate the slices on demand. Consequently, the renegotiation interval, i.e. the time duration in-between two reallocation decision, must be minimized.

Similar to the general 5G use-cases, a key attribute in emergency type communications is the coexistence of different applications and service types with varying urgency, priorities, spatiotemporal requirements and so on. In the context of PriMO-5G, we are considering the fire trucks and the nearby first-responder vehicles to provide a considerable computational power (see Section 1.3). However, the techno economic constraints challenge the availability of such a solution. The envisioned systems with lots of data sources, e.g. drones, robots or bodycams etc., requires immense computational capacity to be implemented in the firetrucks. However, such a deployment indeed places immense economic pressure on the city and can be quite hard to be implemented. Therefore, we are also considering the scenario where the surrounding spectral and computational resources, e.g. eNBs/gNBs or edge, to be utilized in order to execute the tasks. However, this coexistence brings out the challenge of how to handle the inter-service priorities. For example, between an public safety communication and industrial URLLC service (e.g. crane control), how should the resource allocation be handled given that only one of these users would be served. Also, since more than one emergency type communication can exist in the network, how should the resources be distributed among them. The conventional static prioritization mechanisms can easily result in a single service to dominate the complete resources. Therefore, the proposed resource allocation strategies have to dynamically determine the status of each task and then adjust the priority of different traffic accordingly. This way the multiplexing gain can be maximized without endangering the differing tasks. Finally, a key challenge is completing the inter-tenant and inter-service decisions within the given time limits. Although the delay constraints in a conventional network is relatively lax, the firefighting use case requires excessively low delay constraints.

4.5.1 Feasibility Analysis

In order to integrate the flexibility in RAN slicing as well as maintaining the pre-described industrial aspects, we have implemented the dynamic RAN slicing and sharing model described in [AMC19] as the reference model. The simulations are performed in Matlab in a commercially available computer (equipped with 16 GB RAM and Intel i7 CPU) while the optimization problem is solved in Gurobi. Following the approach in resource allocation problems, the time is discretized and divided into time slots. We assumed that a set of tenants (M) are serving in the given region. We assume that the governmental institution acts as a tenant in the environment and in case of emergency response organization (e.g. firefighters), joins to the negotiation processes and dynamically owns resources. Similar to [AMC19], a set of users (K) are distributed homogenously around a base station. We assumed that the users have a walking speed, i.e. 5.5 km/hr.

4.5.1.1 Scenario 1

The first scenario is designed to explore the performance of the mathematical programming-based solution and the time complexity posed by it. In this analysis the model presented in [AMC19] (cf. Model 1), due to its compliance with the ORAN principles, i.e. openness and intelligence. Due to the complexity, the model has been implemented as a two-step algorithm where the first step, i.e. P1, handles the real time resource allocations and run in RT. After a pre-negotiated *Renegotiation Interval* (RI), the algorithm enters the second step, i.e. P2, where the observations during the RI are used to determine the optimum sharing parameters as well as the optimum resource allocations. The calculated parameters are used to determine the sharing parameters for the upcoming RI.

$$\begin{aligned}
 & \min \sum_{m \in M} \xi_m[n] \\
 & \epsilon_m[n] = \left(\frac{1}{(a+1)} \sum_{i=n-a}^n \sum_{k \in K_m} x_k[i] \right) - S_m, \forall m \in M \\
 & |\epsilon_m[n]| \leq \Delta_m, \forall m \in M, \forall n \in N \\
 & \xi_m[n] \geq \max \left(0, U_{th,m} - \frac{1}{(a_m+1)|K_m|} \sum_{i=n-a_m}^n \sum_{k \in K_m} U_k(x_k[i], r_k[i]) \right), \\
 & \quad \quad \quad \forall m \in M, a_m \equiv (n-1 \bmod W_m) \\
 & \sum_{k \in K} x_k[n] \leq 1 \\
 & \sum_{i=n-a_m}^n (S_m(C_{ca} + C_{op}) + \epsilon_m[i]C_{op} + \xi_m[n]C_{pre}) \leq B_m(a_m + 1) \quad \forall m \in M \\
 & 0 \leq \Delta \leq \frac{1}{a_m+1} \sum_{i=n-a_m}^n \left(\sum_{k \in K_{m,elastic}} x_k[i] \right) \quad \forall m \in M \\
 & \sum_{m \in M} S_m \leq 1, S_m \geq 0, \forall m \in M
 \end{aligned}$$

Model 1 The considered optimization model taken from [AMC19]

Through the update mechanism and the feature scaling coefficient, the optimization model can provide an adaptive mechanism that can provide the needed flexibility and the efficiency. However, as reported in [AMC19] this mathematical model is a derivation of Knapsack Problem which is known to be NP-Hard, and as such it would require excessively high time to be completed in particular scenarios.

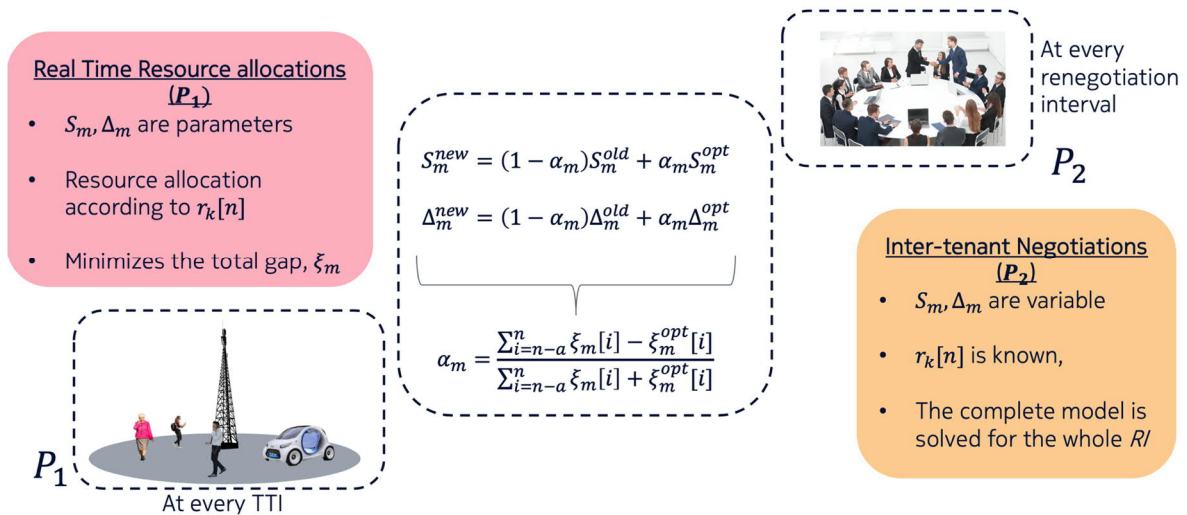


Figure 4.10 Considered 2-step algorithm as proposed in [AMC19]

4.5.1.2 Scenario 2

The time complexity posed by exact optimum solution in Scenario 1 challenges the applicability of the optimum model in the fire-fighting use case. As previously described, the firefighting use case requires the transmission time intervals (TTIs) to be between 1ms to 1s. Therefore, the models that require longer time than the RI could be considered to be non-applicable for our consideration. Therefore, an AI-guided model is considered to increase the speed of the exact optimum model in Scenario 1.

Despite the promising performance of AI based solutions in demand prediction and mitigation decisions, direct implementation of AI in resource allocation decisions in a multi-tenant and multi-service environment imperil the SLAs. Therefore, a hybrid model, where the real time allocations are performed by an optimization model while the non-real time decisions are performed by an AI block is considered. More specifically, the P2 block presented in Scenario 1 is replaced with a neural network block. In this scenario we have considered 3 popular AI models in computer networks, i.e. feed forward neural networks, convolutional neural networks and long-short term memory recurrent network. The AI block is inputting the user mix, the tenants' budgets, average achievable rate per tenant, the number of users per tenant and the window size per tenant. Based on these parameters, the sharing parameters are determined and are forwarded to the P1, the real time resource allocations block. In order to preserve the flexibility and the efficiency of the model presented in Scenario 1, the P1 is maintained in the optimization form. Note that, as outlined in [AMC19], the time complexity of P1 is in the feasible range and can be used in the practical systems.

The AI block is trained using a set of 50 independent simulations (each with 5000-time slots). In the run time, the sharing parameters (S_m and Δ_m) are predicted using the observed achievable rates of the users ($r_k[n]$) from the completed renegotiation interval, the number of users, the window length, and the tenant budgets. Input is passed through a 64-unit convolution layer with processes data in M channels, one for each tenant. After convolution layer 2x2 max pooling is used to flatten the convoluted data and passed onto a fully connected neural network layer with 64 units with the ReLu activation function. The output of the fully connected layer is passed to the output layer with M units, providing M outputs. The second type of neural network used in this work is Long Short-Term Memory (LSTM). The input is passed through two concurrent LSTM layers with 64 units, before passing on to two fully connected dense neural network layers, one with 64 units and M units respectively. Finally, the last neural network type used in this work is a Feed-Forward neural network with 32 nodes in the hidden layer and M outputs. We have used similar AI structures for both S_m and Δ_m predictions.

4.5.1.3 Time complexity comparison

The Table 4.3 presents the time comparison between the different AI blocks and the exact optimum model. Although the duration of a TTI can be adjusted according to the considered scenario, available technology and the available computational power, in the Primo-5G use case, it is assumed to be 1ms, i.e. the most demanding scenario. Note that the demonstrated results for the neural networks in Table 4.3 are not considering the training times for the respective models as it is assumed that the training is done offline and then the respective models are pushed to the edge of the network for the resource management decisions.

Table 4.3 Comparison between measured time complexities (ms) of feed-forward neural network (FFNN), convolutional neural networks (CNN), the long-short term memory (LSTM) and the optimization-based model (P2).

RI	P2	FFNN	CNN	LSTM
10 TTIs	43.1	14.689	1.162	4.132
25 TTIs	192.3	20.458	3.728	10.402
50 TTIs	506.9	33.977	9.820	25.007
100 TTIs	2441.2	36.643	25.471	52.103

The results in Table 4.3 demonstrates the infeasibility of the exact optimum solution in the firefighting use case. For the considered extreme case (i.e. 1 TTI =1ms), LSTM and CNN are the candidate solutions. FFNN is measured to be infeasible for the extreme conditions, however, in more relaxed scenarios (e.g. 1TTI = 2ms) it can also be feasible.

4.5.1.4 Performance comparison

As the application of AI block in the non-RT decision making process satisfies the time complexity requirements, its implications on the algorithm's performance is measured. As a performance metric, the utility gap term in [AMC19] is used. Figure 4.11 demonstrates the performance comparisons between different AI blocks with respect to the exact optimum, whereas the average utility gaps are presented in Table 4.4.

Table 4.4 Performance comparison between different AI blocks (Lower gap values indicates higher performance).

RI	10 TTI	25 TTI	50 TTI	100 TTI
P2	0.4395	0.4180	0.3708	0.3070
FFNN	0.4704	0.4394	0.4192	0.3717
CNN	1.4327	1.5507	1.4600	1.6496
LSTM	0.4482	0.4221	0.3861	0.3107

On one hand, the performance results prove the inefficiency of using convolutional neural networks in the considered management problem. Considering the time varying nature of the input data, this inefficiency can be explained. On the other hand, the results demonstrate the efficiency of the LSTM model. The results report an 2% performance loss by replacing the mathematical programming based P2 with an LSTM model. In conjunction with the decreased time complexity, the measured performance loss is considered to be in acceptable region. In addition to the performance comparison among different models, the results show the increased RI would further decrease the measured gap due to the increased flexibility of the algorithm. Figure 4.11 reports the variation of gap over the simulation horizon. The envisioned hybrid model in Scenario 2 can provide a stable performance over the complete simulation.

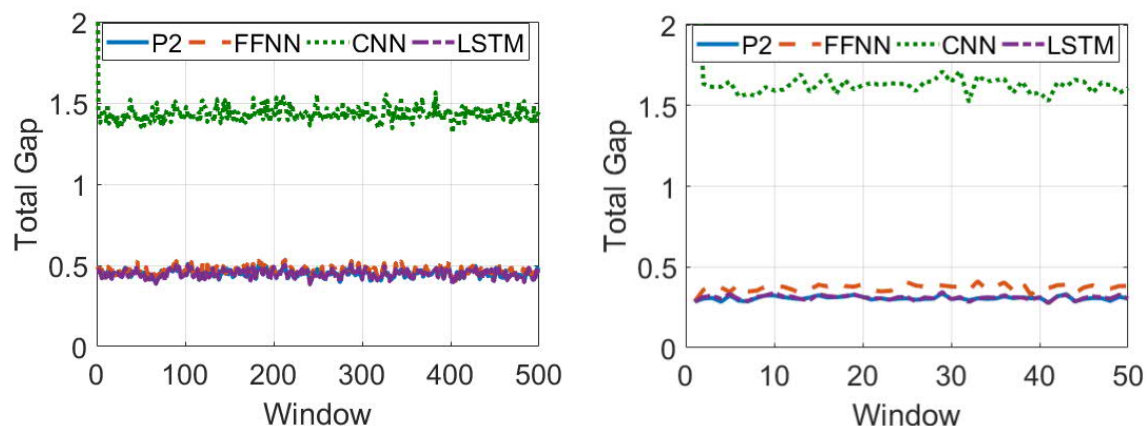


Figure 4.11 Performance comparisons between different algorithms for two different RI, RI=10 TTI (on the left) and RI=100 TTI (on the right)

In addition to the total gap analysis between different algorithms, the variation of the sharing parameters is observed and reported in Figure 4.12 and Figure 4.13.

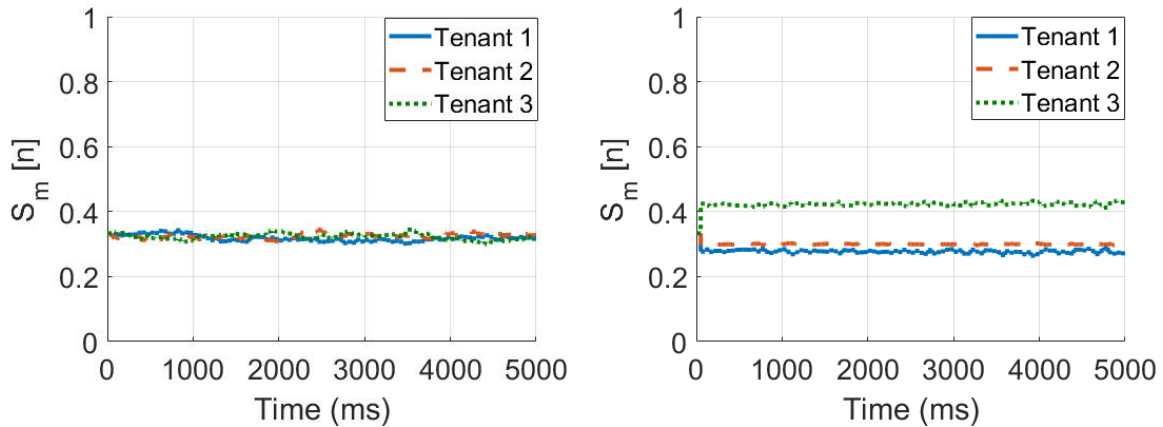


Figure 4.12 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)

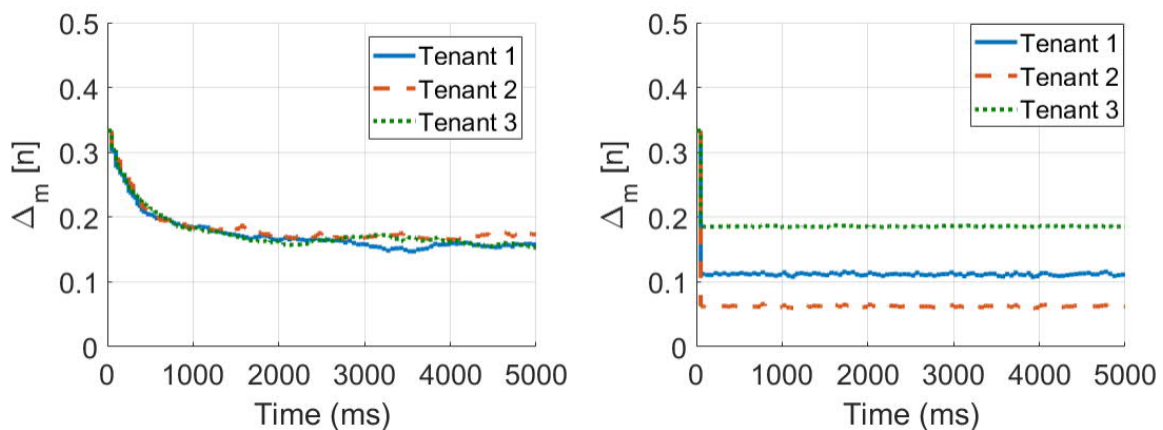


Figure 4.13 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)

Based on the definitions on the individual sharing parameters, it is possible to summarize the parameter in Figure 4.12 as the tenants complete ownership behaviour while the parameter in Figure 4.13 as the tenant's incentive to share. The results indicates that in terms of the ownership behaviour, the AI block behaves nearly the same as the exact optimum behaviour. On the other hand, the sharing flexibility of the tenants shows a steady state characteristic which is slightly different from the optimum solution. This steady state is a reflection of the losing the multiplexing gain. More specifically, since the AI block only considers the average achievable rate rather than the variations, the tenants has less incentive to share their resources. However, considering the utility gap plots, the P1, i.e. the optimization based resource allocation strategy, can compensate the non-optimum sharing parameters.

4.6 Service continuity

As experimentally demonstrated for the steady-state conditions, the AI-based solutions can provide comparable performances as optimization-based models with lower time complexities. However, for the envisioned PriMO5G use cases, the steady-state behaviors can be hard to maintain. The changing dynamics of fire as well as the different tasks becoming active/passive depending on various conditions require rapid adaptation to the most recent network state. The optimization-based solutions guarantee to be always operating at the optimum working point, i.e. the sharing parameters that are most suitable

for the given traffic mix and the channel conditions. However, the application of AI-based solutions brings its advantages and challenges in maintaining the service continuity under the transient-state.

On one hand, unlike the optimization-based models, the AI-based solutions require the pre-trained models, that can demonstrate the different network states and the optimum response to these states. The performance of this AI-based model is strongly tied to the training process. For the AI-model to be efficient, the training sample has to contain the majority (if not all) of the network states that can be observed. However, considering the continuous nature of the network parameter values as well as the infinite number of combinations of network states, training an AI network can be quite challenging. Moreover, the envisioned service heterogeneity in 5G and the flexible working environment further challenges the model training and provisioning. To capture the inter-service dynamics, the trained models have to be fully updated every time a tenant would like to introduce a new service to its customers. Therefore, unlike the optimization-based network management strategies, maintaining the optimum working point in AI-guided solutions is very challenging.

On the other hand, utilizing the data aggregated from different sources can arguably be simpler in an AI-based model. Through correlating the different sensor data aggregated from various regions, the evolution of specific events can be predicted and efficiently be reacted. This anticipatory network management is especially efficient for the rapidly changing network states, e.g. the emergency scenarios. For the considered firefighting use-case in Primo5G, the wind speed data around the emergency area can be collected and utilized in the AI block to determine the evolution of fire and the shifting dynamics in the fire area, which can be used in the network management decisions. This anticipatory network management strategy can provide a critical advantage in emergency scenarios. In the future smart cities, the different AI blocks from different parts of the city management can be connected to provide a broader vision of the fire and provide a city-scale response to emergencies.

A fully deployed AI-based network management solution proposes self-awareness in the network management, anticipatory strategies, and slower response time to the shifting dynamics in the network. However, to maintain an optimum (or close to optimum) working point, the deployed models need to be trained adequately, which is quite challenging. In a relatively long term (i.e. in the order of days), the non-optimum working condition can be tolerated with strict control of the network conditions and performing the necessary updates on the AI block to respond to the shifting dynamics. However, in the short term, this non-optimality can lead to economically undesired and even dangerous situations. As such, the value of AI-based network management is strongly tied to the network performance during the transient state, i.e. the time between the observation of the new network state and the receipt of the accurate AI-model. Therefore, in this section, the adaptation skills of previously described two models (i.e. in Scenario 1 & 2) during the transient state are investigated.

4.6.1 Adaptation to the changing channel conditions and tenant strategies

In order to test the implications of changing achievable rates and the tenant policies on the resource sharing dynamics, we have artificially decreased the achievable rates of the second tenant. In this considered case, the achievable rates of the users of tenant 2 are decreased 60% at $n=1000$ TTI. Following this decrease, the tenant 2 changes his policy by making a special agreement and increases his utility expectation (i.e. the tenant's priority) 2 times.

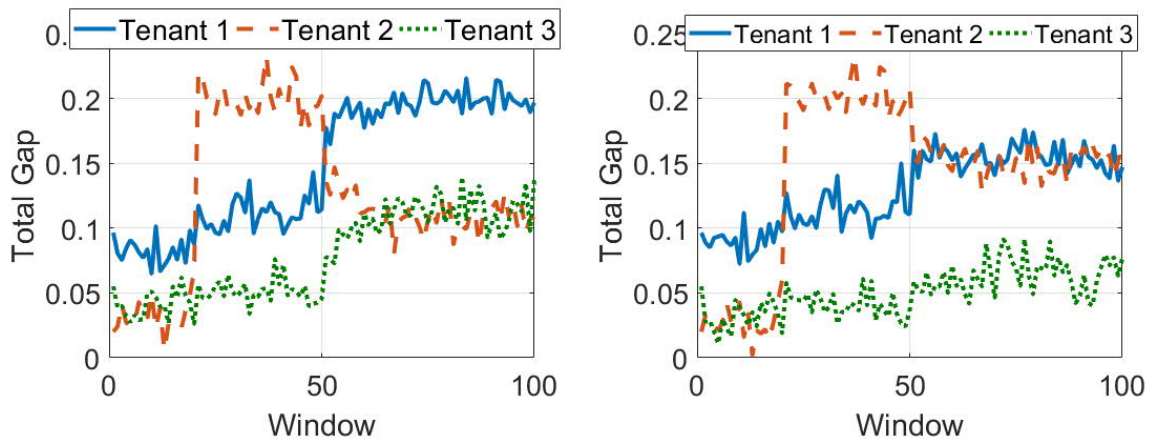


Figure 4.14 Performance comparisons between different algorithms (P2 (on the left) and LSTM (on the right)) for RI=50 TTI

Figure 4.14 presents the total gap variations of different tenants over the time horizon. Following the sudden decrease in the average achievable rate the second tenant starts observing a gap value approximately 4 times higher than the normal value. However, following the policy change at 50th time window, the second tenant manages to decrease the utility gap 50% while the gaps of the other two tenants increases. Based on the values in Figure 4.14, the policy change affects the two models in different values. However, the overall performance (i.e. the average total gap over all the tenants and simulation horizon) loss of using AI-guided model in scenario 2 is measured to be approximately 2%. This result shows that the AI-based models needs to be updated in order to reflect the tenant policies, the hybrid nature of scenario 2 can limit the implications on the overall performance.

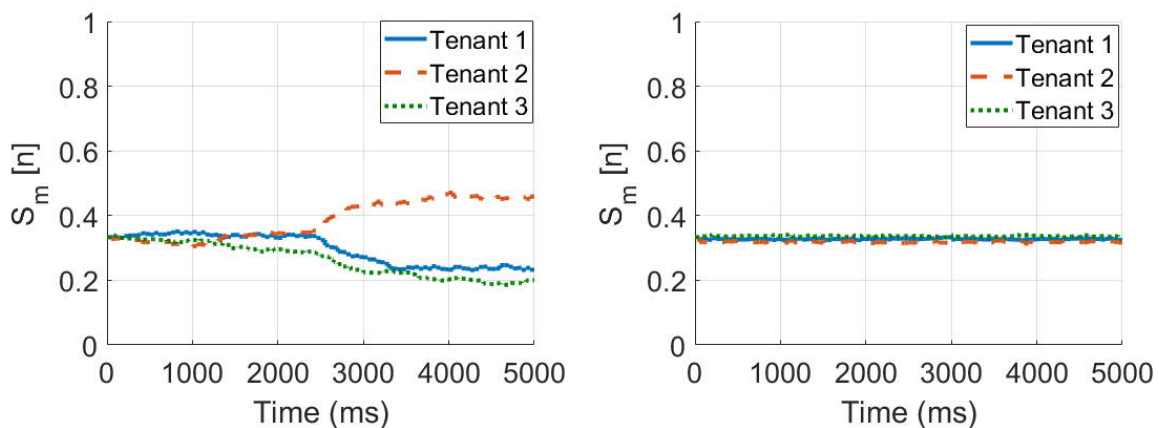


Figure 4.15 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)

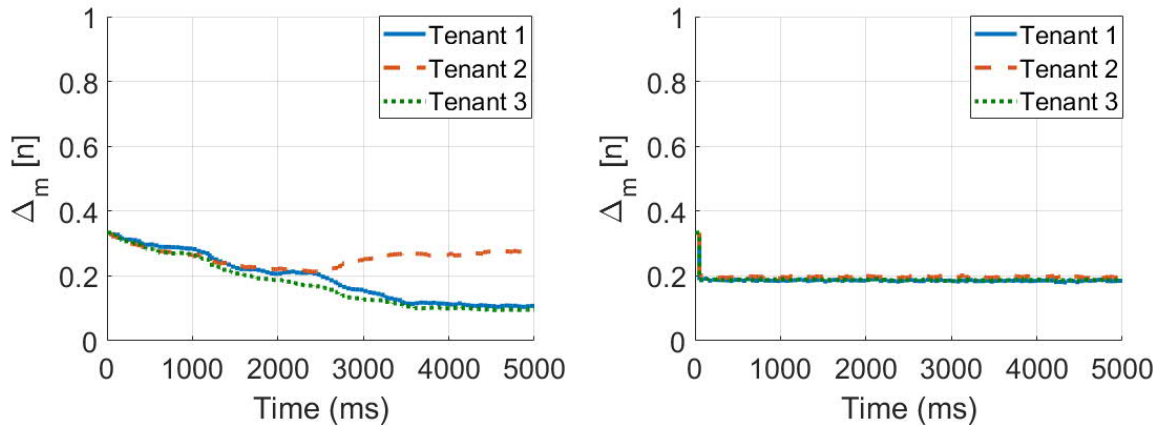


Figure 4.16 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)

Figure 4.15 and Figure 4.16 report the sharing parameter changes over the simulation horizon. The optimization-based model reacts to the changes in the tenant's policy by changing the sharing parameters. As such we are observing that the second tenant starts increasing both his guaranteed resources and the trade incentive. However, the AI-guided solution does not react to the change in the sharing parameters. Therefore, the performance of the algorithm has been obtained through the flexibility of the hybrid approach.

4.6.2 Adaptation to the changing traffic mix

Finally, in this part, the adaptation to the changing traffic mix has been measured. In this scenario, the tenants are assumed to have symmetric traffic mix, namely has similar traffic types and the user demand per slice type. At $n=2500$ TTI, the traffic mix changed, and the tenants are considered to have asymmetric traffic types. From the overall performance, we measured approximately 1% performance loss by using an AI based solution. Considering the sharing parameters presented in Figure 4.17 and Figure 4.18, the low decrease can be explained by the small changes in the sharing parameters. More specifically, the AI-based model's lack of adaptation does not really limit the performance of P1 (i.e. real time resource allocation problem) due to the small difference between optimum and heuristic solution.

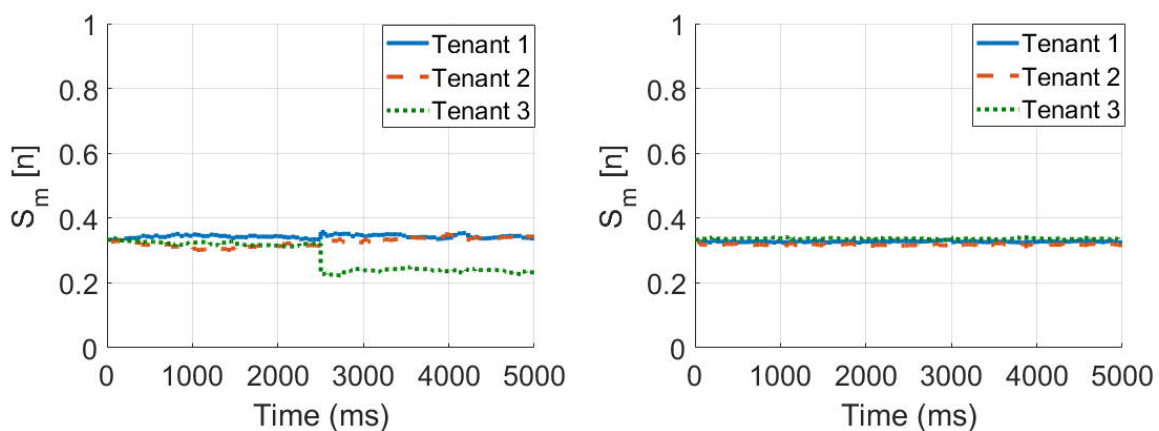


Figure 4.17 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)

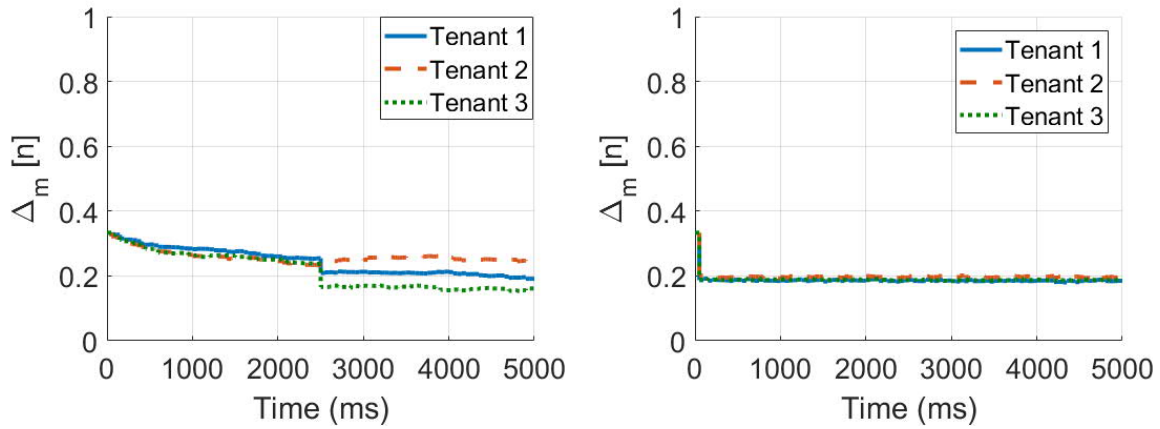


Figure 4.18 Comparison between variations in the sharing parameters over time using P2 (on the left) and LSTM (on the right)

The lack of adaptation of the AI block during the transient phase, i.e. the time duration between detection of a new network state and the reception of an accurate model, can lead to non-optimal solutions. However, our analysis shows that the hybrid application of AI and optimization-based models can provide both the high efficiency and low time complexity.

5 Conclusions and Outlooks

This deliverable aims to present the necessary frameworks and architectural components that expose network services and capabilities to external network users and application developers in order to satisfy KPIs of use cases such as those relevant to Primo-5G. We presented definitions and an analysis on network service and capability exposure, backhaul and MEC as well as RAN slicing using standardised APIs.

The document was divided in three main sections, namely “Network Service and Capability Exposure”, “Backhaul and MEC” and “Radio Access Network (RAN) slicing”. Section 2 presented an overview of technologies, APIs and architecture elements that allow service exposure in LTE and 5G. Afterwards, Section 3 focused on resource management regarding mobile backhaul and the optimal placement of the MEC. This section also presents the usage of network slicing to guarantee the reliability for the firefighting traffic. Moreover, this section describes the network function (NEF) to be used for exposing MEC for external video processing applications used during the firefighting incidents. Finally, Section 4 provided RAN slicing definitions, an overview and analysis of function splits within RAN, a presentation of open APIs for RAN slicing as well as an analysis for dynamic RAN slicing and service continuity using AI-based solutions.

Therefore, it has been shown that by employing such frameworks and architectural components within 5G systems, it is possible to expose network functionalities and service capabilities for external applications to deploy and manage resources regarding MEC platforms as well RAN slices.

At the time of writing, as the telecom industry moves towards 5G, the network capabilities exposed over the north bound (T8) APIs, supported by SCEF and NEF for LTE and 5G, will enable IoT servers to access the capabilities and services of the 3GPP networks. The industry adoption of the APIs has already begun. For instance, OneM2M is a standards development organization that is developing service layer standards for IoT devices, gateways, and servers [ONE19]. OneM2M’s Infrastructure Common Services Entity (IN-CSE) is a type of SCS/AS that can make use of the T8 APIs. OneM2M has already standardized how an IN-CSE can take advantage of the T8 APIs, including defining how oneM2M messages can be exchanged with UE-hosted oneM2M applications via NIDD APIs.

Furthermore, there also appears to be a significant industry momentum behind exposure of cloud-based disaggregated RAN architectures to external service providers and developers. A recent online article noted this trend to be global in nature and covering different kinds of deployment scenarios (rural, urban, private etc.) [RCR20]. The early efforts, notably from O-RAN Alliance (described in Section 4) in creating open interfaces towards the RAN and specifying RAN controllers is prompting other alliances (e.g. Open Network Foundation’s Software-Defined RAN initiative) [OPEN20] and vendors to leverage those specifications to build RAN products that can be flexibly adopted to different use case requirements (e.g. public safety).

The overall API standardization work and the development of capability exposure ecosystem has now reached the stage where the exposed network capabilities can be readily monetized transforming the business models of the network operators while providing opportunities for service providers, application developers, and enterprises that may want to create new products and services with the telecommunications capacities afforded to them via core network APIs. Therefore, new opportunities emerge for growth and innovation beyond simply accelerating connectivity.

6 References

- [3GPP-22127] 3GPP Technical Specification 22.127 Service Requirement for the Open Services Access; Stage 1, Release 9, 2009-12
- [3GPP-23127] 3GPP Technical Specification 23.127 Open Service Access OSA Virtual Home Environment-VHE, Release 4, 2006-06
- [3GPP-23198] 3GPP Technical Specification 23.198 Open Service Access; Stage 2, Release 9 2009-12
- [3GPP-23222a] 3GPP Technical Specification 23.222, Functional architecture and information flows to support Common API Framework for 3GPP Northbound APIs, Release 17, Stage 2 2020-03
- [3GPP-23222b] 3GPP Technical Specification 23.222 Common API Framework for 3GPP Northbound APIs; Release 17, Stage 2, 2020-03
- [3GPP-23288] 3GPP Technical Specification 23.288 Architecture enhancements for 5G System (5GS) to support network data analytics services. Release 16. 2020-07
- [3GPP-23401] 3GPP Technical Specification 23.401, Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access, Release 16, 2020-03
- [3GPP-23501] Technical Specification 23.501 System architecture for the 5G System (5GS), Release 16, stage 2 2020-03
- [3GPP-23502] 3GPP Technical Specification 23.502 Procedures for the 5G System (5GS); Release 16, Stage 2 2020-03
- [3GPP-23682] 3GPP Technical Specification 23.682, Architecture enhancements to facilitate communications with packet data networks and applications, Release 13, 2018-06
- [3GPP-24250] 3GPP Technical Specification 24.250, Protocol for Reliable Data Service, Release 16, Stage 3, 2019-12
- [3GPP-28530] 3GPP Technical Specification 28.530 Aspects; Management and orchestration; Concepts, use cases and requirements; Release 16, 2019-12
- [3GPP-28801] 3GPP Technical Report 28.801 Study on management and orchestration of network slicing for next generation network; Release 15, 2018-01
- [3GPP-29122] 3GPP Technical Specification 29.122, T8 reference point for Northbound APIs, Release 16, 2019-12
- [3GPP-29222] 3GPP Technical Specification 29.222 Common API Framework for 3GPP Northbound APIs; Release 16, 2020-06
- [3GPP-29522] 3GPP Technical Report 29.522 5G System; Network Exposure Function Northbound APIs; Stage 3, Release 16
- [3GPP-38300] 3GPP Technical Specification 38.300 NR and NG-RAN Overall Description; Stage 2, Release 15 <https://www.3gpp.org/DynaReport/38300.htm>
- [3GPP-S2] 3GPP S2-1908583 Feasibility Study on Enhancement of Network Slicing Phase 2, June 2019
- [5GEMP20] <https://5g-empower.io/>
- [AMC19] Ö. U. Akgül, I. Malanchini and A. Capone, "Dynamic Resource Trading in Sliced Mobile Networks," in *IEEE Transactions on Network and Service Management*, vol. 16, no. 1, pp. 220-233, March 2019, DOI: 10.1109/TNSM.2019.2893126.
- [ETSI-OSA] ETSI ES 201 915-1 Open Service Access (OSA); Application Programming Interface (API); Part 1 (Parlay 3) 2003-07

- [GSMA19] GSMA, "Generic Network Slice Template", Version 1.0, 23 May 2019, URL: <https://www.gsma.com/newsroom/wp-content/uploads/NG.116-v1.0.pdf>
- [MCM+17] G. Mountaser, M. Condoluci, T. Mahmoodi, M. Dohler and I. Mings, "Cloud-RAN in Support of URLLC", 2017 IEEE Globecom Workshops (GC Wkshps), Singapore, 2017, pp. 1-6, DOI: 10.1109/GLOCOMW.2017.8269135.
- [MK03] A. -. Moerdijk and L. Klostermann, "Opening the networks with Parlay/OSA: standards and aspects behind the APIs," in IEEE Network, vol. 17, no. 3, pp. 58-64, May-June 2003, DOI: 10.1109/MNET.2003.1201478.
- [MOH19] A. Mohammedadem "Optimizing Mobile Backhaul Using Machine Learning", URL: <https://aaltodoc.aalto.fi/handle/123456789/39839>
- [MOS5G20] <http://mosaic5g.io/flexran/>
- [ONE19] OneM2M Technical Specification: 3GPP Interworking, URL: https://onem2m.org/images/files/deliverables/Release3/TS-0026-3GPP_interworking-V3_0_0.pdf
- [OPEN20] <https://www.opennetworking.org/sd-ran/>
- [ORAN19] O-RAN Alliance, O-RAN WhitePaper - Building the Next Generation RAN, October 2019
- [ORAN20] O-RAN Alliance, O-RAN Use Cases and Deployment Scenarios WhitePaper, February 2020
- [PAR-OSA] PARLAY/OSA information on: <https://web.archive.org/web/20071003121805/http://www.etsi.org/WebSite/Technologies/OSA.aspx>
- [PRIMO-D11] PriMO-5G D1.1, "PriMO-5G Use Case Scenarios," PriMO-5G project deliverable, Available: <https://primo-5g.eu/project-outcomes/deliverables/>, 2019.
- [RCR20] <https://www.rcrwireless.com/20200806/opinion/readerforum/open-ran-101-open-ran-adoption-in-different-regions-why-what-when-how-reader-forum>
- [SAM20] Samsung, "Network Slicing", Samsung Technical White Paper, April 2020
- [SCF20] Small Cell Forum <https://www.smallcellforum.org/>
- [SCF-API] Small Cell Forum, 'Network Monitor Mode API' March 2020 https://scf.io/en/documents/224_5G_FAPI_Network_Monitor_Mode_API.php
- [SCF-FAPI] Small Cell Forum, '5G FAPI: RF and Digital Front End Control API', March 2020 https://scf.io/en/documents/223_5G_FAPI_RF_and_Digital_Frontend_Control_API.php
- [SCF-PHY] Small Cell Forum, '5G FAPI: PHY API', March 2020 https://scf.io/en/documents/222_5G_FAPI_PHY_API_Specification.php
- [SSW18] M. Starsinic, D. Seed, and C. Wang. 2018. "An Overview of 3GPP Exposed Services for IoT Service Platforms". GetMobile: Mobile Comp. and Comm. 22, 2 (June 2018), 16–21. DOI:<https://doi.org/10.1145/3276145.3276153>