5G key technology enableRs for Emerging media COntent pRoDuction services

Deliverable D3.1

First Description of 5G components

Version 1.0

Date: 2021/07/29

Document properties:

Grant Number:	957102
Document Number:	D3.1
Document Title:	First Description of 5G Components
Editor(s):	Giulio Stante (RAI); Cristina Avellán (UPV), Adrián Rodrigo (UPV), Salvador García (UPV)
Authors:	Pablo Perez (NOK); Pavlo Kondraten (EBU); Paola Sunna (EBU); Giulio Stante (RAI); Davide Desirello (RAI); Davide Milanesio (RAI); Alberto Messina (RAI); Federico Pandolfi (RAI); Peter Brightwell (BBC); Turker Yilmaz (EUR); Norbert Werner (SEN), Jan Duerre (SEN); Christophe Le Thierry d'Ennequin (RED); Baruch Altman (LU); Antonio Garcia (ACC); Manuel Fuentes (5CMM); David Martìn-Sacristàn (5CMM); Hèctor J. Donat Añó (5CMM); Mohamed Nabil Ibrahim (EDD); Thorsten Lohmar (EDD); Henk den Bok (RBEE); Morten Brandstrup (TV2)
Reviewers	Paola Sunna (EBU); Thorsten Lohmar (EDD); Cristina Avellán (UPV); Adrián Rodrigo (UPV); Salvador García (UPV); Davide Desirello (RAI), Davide Milanesio (RAI), Ian Wagdin (BBC), De Mi (University of Surrey)
Contractual Date of Delivery:	2021/07/31
Dissemination level:	PU[1]
Status:	Final
Version:	1.0
File Name:	Deliverable D3.1 – First Description of 5G components

Revision History

Revision	Date	Issued by	Description
1.0	2021/07/29	RAI	EC version

[1] CO = Confidential, only members of the consortium (including the Commission Services)

PU = Public

Abstract

The European H2020 5G-RECORDS Project aims at investigating whether and to what extent the technological improvements introduced by 5G network can be exploited for the production of professional multimedia content.

Within the project, three main use case are considered that could benefit most from the improvements related to the 5G network: live audio production, multiple camera wireless studio and live immersive media production.

For each of them, new 5G components are being designed, developed and used by project partners to investigate the potential of using 5G. Moreover, to this purpose, three different 5G network deployments are provided to the project, one for each of the considered use case.

This deliverable describes a first general description of all the components involved and list all major design and development activities carried out in the first phase of the project. More detailed information will be provided in deliverable D3.2.

Keywords

5G, Audio-visual sector, Component description, Component development and integration, Component elements, end-to-end infrastructures, Non-public networks, Professional media content production.

Disclaimer

Executive Summary

5G-RECORDS is a European project aiming at investigating some of the relevant business opportunities that 5G brings in the context of the professional multimedia content production.

In this project, three main use cases are considered: live audio production, multiple camera wireless studio and live immersive media production. Each use case may have a variety of scenarios on how the technology may be deployed. For each use case, the project considers specialized end to end 5G infrastructures.

The scope of this deliverable is to report a first description of all components and infrastructures involved and to report the progress made in the first phase of activity of WP3 regarding design and development of new 5G component elements.

This deliverable lists, for each use case, all the component elements involved, distinguishing between component elements available at the beginning of the project and component elements designed and developed in the context of the project. Components' descriptions are subdivided in different chapters according to the use case they are used in. Descriptions of 5G components shared among different use cases are collected in a separate chapter.

In this deliverable, the 5G network architecture is described in general terms, 5G-RECORDS will deploy three separate end-to-end 5G infrastructures, most of the 5G functions used are common, therefore all reported together. Remaining specific network elements are reported in the specific use case chapters.

One of the shared components is the Media Operational and Control Gateway (MOCG). This component controls the setup of media resources between a 5G enabled device such as a 5G camera microphone or other devices. It sits on top of the 5G network and is used to automate and simplify the operational control of broadcast equipment deployed in different locations. In production, this is done through AMWA NMOS APIs, specifically IS-05. In the 5G network, the use of these APIs needs to be adapted through the introduction of several middleware components decoupling the remote operations from the studio and replicating/emulating NMOS control functions within a 5G environment.

In the live audio production use case, the goal is to demonstrate the feasibility of using 5G professional audio equipment within a high-quality, ultra-low latency local wireless audio production network. For this purpose, besides end user equipment such as microphones and in-ear monitoring receivers, most of the work is being done on components that constitute the 5G infrastructure, on network configuration and on media orchestration, including network slicing management and shared access servers.

The multi-camera wireless studio looks at the possibilities of having some studio equipment, such as wireless 5G-enabled cameras in a 5G S-NPN (Standalone Non-public network). These could be co-located or separate to the broadcaster's production centre. In this use case, the activities carried out are mainly around the design and development of the Media gateway, the codec board and modem, a possible new workflow for camera control, and the already mentioned Media Operational Control Layer (MOCL).

The last considered use case is the live immersive media production. It consists in an end to end chain for a content production based on Free Viewpoint Video (FVV). Involved components can be classified in three groups: end-devices, Virtual Network Functions (VNF) and 5G network elements.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

The acronyms list has a special style defined as "acronyms". Each acronym is separated by a tabulation with each definition. As is shown below:

3GPP	3rd Generation Partnership Project
5G	5th Generation of mobile communications systems
AF	Application Function
AMWA	Advanced Media Workflow Association
API	Application Programming Interface
AV	Audio-Visual
CPE	Customer Premise Equipment
CH	Content Handler
CU	Centralized Unit
CU-CP	CU-Control Plane
CU-UP	CU-User Plane
EC	European Commission
ETSI	European Telecommunications Standards Institute
FVV	Free Viewpoint Video
FWA	Fixed Wireless Access
HEVC	High Efficiency Video Coding
gNB	gNodeB
ICT	Information Communication Technology
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IEM	In-Ear monitoring
IETF	Internet Engineering Task Force
IGMP	Internet Group Management Protocol
IS (AMWA)	Interface Specification
ISO	International Standards Organization
ITU	International Telecommunications Union
JWT	JSON Web Toke
KPI	Key Performance Indicator
LSA	License Shared Access
MAC	Media Access Control
MCR	Master Control Room
MEC	Multi-access Edge Computing
MG	Media Gateway
MNO	Mobile Network Operator
MOC	Media Orchestration and Control
MOCG	Media Orchestration and Control Gateway
MPEG	Moving Picture Experts Group (formally, ISO/IEC JTC 1/SC 29/WG 11)
MQTT	Message Queuing Telemetry Transport
Near-RT RIC	Near Real-Time Intelligent Controller
NEF	Network Exposure Function
NMOS	Network Media Open Specifications
NSA	Non Stand Alone
NPN	Non-Public Network
NVF	Network Function Virtualisation
NSA	Non-standalone
OPNFV	Open Platform for NFV
O-RAN	Open Radio Access Network

PCF	Policy Control Function
PLMN	Public Land Mobile Network
PMSE	Programme Making and Special Events
PNI-NPN	Public Network Integrated NPN
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
REST	REpresentational State Transfer, an architectural style used in APIs
SA	Standalone
SDP	Session Description Protocol
SIM	Subscriber Identification Module
S-NPN	Standalone NPN
SME	Small and medium-sized enterprise
SMPTE	Society of Motion Picture and TV Engineers
SDR	Software Defined Radio
ST	(SMPTE) Standard
TAI	International Atomic Time
UC	Use Case
UE	User Equipment (the 5G "device”)
UPF	User Plane Function
vRAN	Virtualized RAN
VSF	Video Services Forum
WAN	Wide Area Network

1 Introduction

1.1 Overview on 5G-RECORDS

In recent years, the new 5G technology and the related innovations are enabling new business opportunities in various industrial fields. One field that could benefit from this evolving scenario is multimedia content production. In this respect, the Horizon 2020 project named “5G-RECORDS” aims to investigate the possibility of exploiting 5G technology in professional audio and video applications.

In particular, in 5G-RECORDS new 5G components specific to the field of professional production of media content are designed, developed, and integrated. These components will be validated within a 5G infrastructure provided by project partners.

In this project, three main use cases are considered: live audio production, multiple camera wireless studio, live immersive media production.

1.2 Scope of this deliverable and structure

Deliverable D3.1 aims to provide a first description of the 5G components involved in the project, documenting overall design, as well as the actual implementation of the first phase. Whenever an element is designed but its implementation is delayed for subsequent project phases (or there is any incomplete or pending functionality), it is also recorded in the document. The final release of the infrastructure and 5G components will be reported in deliverable D3.2.

The document is structured in the following way:

- *Chapter 1* introduces the objectives of the 5G-RECORDS project, related activities and describes the purpose of this document.
- *Chapter 2* lists all the component deployed, specifying what component elements were developed during the project.
- *Chapters 3 , 4 & 5* provide a first description of 5G components involved and give a brief overview of the considered use cases, respectively named “Multiple Camera Wireless Studio”, “Live Audio Production” and “Live Immersive Content Production”.
- *Chapter 6* provides a description of the media orchestration services.
- *Annex A: 5G Network* presents the main features that define the 5G network, for both core network and RAN.
- *Annex B: Additional information Edge Cloud*. contains some additional, but relevant information of the Edge Cloud used within UC3.
- *Annex C: FVV capture system calibration* provides information related to FVV capture system calibration for cameras used in UC3.
- *Annex D: Considerations about deployment scenarios* includes some considerations about four different deployment scenarios, with respect to UC2.
- *Annex E: Overview of the AMWA Networked Media Open Specifications (NMOS)* provides an Overview of the AMWA Networked Media Open Specifications (NMOS), that needs to be taken into account for the design of MOCL.

2 Components in 5G-RECORDS

2.1 List of all the components

The lists of components are shown in tables below according to the use case in which they are used.

Table 1: List of components for UC1

Component	Category	Deployment location	In charge of
Local Audio Processing	End device	On-site	SEN
Audio User Terminal	End device	On-site	SEN, EUR
Remote Server	Media orchestrator & management Server	Internet	SEN
5G RAN (inc. Shared Access Client)	Network	On-site	ACC, EUR
5G Core	Network	On-site	CMC
Shared Access Server	Shared Access Server	RED Technologies premises	RED
Time Service	Time Service	On-site	SEN
Network Slice Management	Network Component	Cloud	CMC

Table 2: List of components for UC2

Component	Category	Deployment Location	In charge of
Media Gateway	Video receiver + decoder	Aachen & Turin	EBU
5G Network	Network	Aachen	EDD
5CMM 5G Modem	Network equipment	Aachen	5COM
IM Encoder	A/V encoder	Aachen	IM
RBEE MCR	MCR	Cloud	RBEE
LU800	Video tx + encoder	Aachen	LU
LU2000-SMPTE	Video rx + decoder	Turin	LU

Table 3: List of components for UC3

Component	Category	Deployment Location	in charge of
Capture Server	End device	Smart Venue	UPM
Production Console	End device	Smart Venue	UPM
User Terminal	End device	Smart Venue Remote Loc.	TID (+NOK)
View Renderer	VNF	Near-Edge	UPM

Stream selector	VNF	Near-Edge	UPM
Storage	VNF	Near-Edge	UPM
Media Proxy	VNF	Near-Edge	NOK
Media Delivery Server	VNF	Edge	NOK
5G CPE	Network component	Smart Venue	NOK
5G gNB	Network component	Smart Venue	NOK
5G Core	Network component	Near-Edge	NOK
Edge MEC	Network component	Near-Edge	NOK
Delivery cloud	Network component	Edge	TID
End-to-End SDN	Network component	Edge	TID

Table 4: List of components shared across multiple UCs

Component	Category	Deployment location	In charge of
Media Orchestration and Control Gateway	Media orchestrator	Cloud	BBC

2.2 Components' elements: what was available & what was developed so far

This section presents the list of the components' elements development during 5G-RECORDS within each use case.

Table 5: List of component elements developed during the project for UC1

Component	Existing elements (available before the beginning of the project)	Elements designed / developed / integrated during the project
Local Audio Processing	Rapid Prototyping Audio Hardware Platform	FPGA and software functions: • Network Audio Source/Sink • Audio Device Control Client • PTP Client • Audio Mixer
Audio User Terminal	Rapid Prototyping Audio Hardware Platform	FPGA and software functions: • Audio Device Control Client • Network Audio Source/Sink • PTP Client
Remote Server	-	Software functions: • Audio Device Control Server • Media Operational Control Gateway • Core Configuration Service

5G RAN (inc. Shared Access Client)	Software functions: • eMBB 5G CU • near-RT-RIC • 4G Shared Access Client	Software functions: • URLLC 5G CU • 5G Shared Access Client Open5Glab capabilities: • Several. Please refer to Table 1 of D4.1 [1]
5G Core	Basic 5G Core	• Slicing management capability including external API • Slicing functionality in 5G core • PCF
Shared Access Server	Path loss computation	• Lease creation • Protection of leases • Synchronization with leases database • Leases visualization
Time Service	-	Software functions: • PTP Server

Table 6: List of component elements developed during the project for UC2

Component	Existing elements (available before the beginning of the project)	Elements designed / developed / integrated during the project
Media Gateway	• Gstreamer-based framework • Typescript RESTful API framework	• RESTful API to manage media content handlers • Project-specific graphical user interface • Specific Gstreamer pipelines • Integration of NVIDIA Rivermax in the Gstreamer pipelines • Integration of RIST libraries in the pipelines • Configuration of the NVIDIA Jetson based image for the requirements of the project
Network Slice Management	• N5 • NSSF user interface	• MOG to N5
5G Network	• Trial network already available	• Test network is developed during the project
5CMM 5G Modem	The 5G modem has been designed and developed from scratch based on the needs and requirements of 5G-RECORDS and considering connectivity towards the encoder.	• Designed/developed: 5G Ethernet/USB board for broadband communications. • Integrated: 5G Rel-15 modem, 4x4 MIMO antennas, SIM card, microprocessor.

IM Encoder	Origami Square Bamboo FPGA Module	• Origami Square Carrier board for Broadcast Camera IO ZCU-106 IO board for Broadcast Camera
MCR	On-premise, S2110-based MCR, incl. signal acquisition (e.g. satellite downlink), processing, monitoring, switching and operations.	Public cloud end-to-end "MCR" chain, i.e. signal contribution (JPEG-XS), processing (AWS-CDI) and signal egress.
LU800	Device announced with project start, continued development with the period.	New capabilities and features: • adding Sierra 5G module for 5G SA and its integration/configuration • enhancing the IP pipe for the control traffic.
LU2000-SMPTE	Pre-commercial version already available.	New capabilities and features: • multi-networks support (master clock, input, output), • configurations • robustness • ongoing work on lower latency.

Table 7: List of component elements developed during the project for UC3

Component	Existing elements (available before the beginning of the project)	Elements designed / developed / integrated during the project
Capture Server	• Capture servers • 3 cameras per server • 2 GPUs per server (depth computation + encoding	• Capture simulation module for transmitting pre-recorded FVV sequences. • RTP traffic smoother. • Configurable MTU size.
Production Console	Smartphone to control virtual camera position show synthetized view	To be determined (subsequent phases)
User Terminal		A HTML5 web player based on videojs to serve the video and collect metrics.
View Renderer	Online view renderer over desktop hardware platform	Developed Offline version that works reading pre-recorded video file. • RTP packet reordering capability. • Single-GPU working mode. • Logging statistics generation during execution • Integration in MEC hardware platform.

Stream selector		To be determined (subsequent phases)
Storage		To be determined (subsequent phases)
Media Proxy / Media Delivery Server	Adaptive Streaming Server,	Integration of adaptive streaming server, modules for RTP video processing, transcoding configuration, monitoring tools
5G CPE	3rd party ASKEY modem	Modem configuration and position to optimize performance
5G gNB	5G gNB in 3.5 GHz using Nokia radio	Integrate and configure mmWave equipment (antenna, baseband, controllers) in n257. Fine tune radio options for performance.
5G Core	Distributed core to support 5G NSA in 3.5 GHz	Update core software release and configuration to support n257.
Edge MEC		• New hardware platform acquired and integrated. • OpenStack configuration. • Availability of GPU within OpenStack VM.
Delivery cloud	Telefonica Edge infra-structure available on Peñuelas Central Office based on OCP servers, switches and storage.	A tenant has been configured in Telefonica Edge solution, to ensure compute, storage and network capacity for the required deployments in order to guarantee UC3 needs.
End-to-End SDN	Existing SDN inside Telefonica Edge and Telefonica transport network.	New components: • Conditional DNS • Slice selector • Monitoring elements to manage the required service level and slices. • New edge cloud tenant connected to Telefonica Transport Network using two slices multimedia and best effort. Both slices connected to Segovia central office (local users) and to remote users in the Spain geography

Table 8: List of component elements developed during the project for components shared across multiple UCs

Component	Existing elements (available before the beginning of the project)	Elements designed / developed / integrated during the project

Media Operational Control Gateway	• Sony open source NMOS C++ Node + Registry implementation (nmos-cpp)
	• Sony open source NMOS JavaScript Client implementation (nmos-js)
		• Designed general architecture for Media Orchestration and Control Gateway
		• Developed enhancements to nmos-cpp and nmos-js to additional functionality:
		  ○ Registration and connection of device "through the gateway"
		  ○ Compressed (non-2110) stream types
		  ○ Carriage of camera control information in IS-07

3 UC1 Components - Live Audio Production

In a typical live audio production, such as a concert, musical, theatre or studio performance, one or several artists are performing (i.e., acting, dancing, singing, or playing music instruments) live either to create content that can be used later or to entertain an interested audience that can be live on site or follow the content live via stream.

This use case will focus exemplarily on the AV production scenario to demonstrate the feasibility of 5G for professional audio equipment targeting a local wireless high-quality and ultra-low latency audio production network. For further details see D2.1 [2].

3.1 General Architecture

Figure 1 shows the general architecture which is used in this project to realize the UC1 scenario. As the use case requires, multiple 5G-enabled microphones and in-ear monitoring receivers are connected to a local 5G network. An in-ear mix for each IEM can be generated in a local audio processing device that is connected via Ethernet cable to the 5G system (5GS). Ultimately all audio devices should be controllable by a user over the internet. Therefore, a media orchestration controller (MOC) is used that translates the user inputs into NMOS control commands. A Media Orchestration Control Gateway (MOCG) translates this NMOS commands to device specific protocols. Through a core configuration service (CCS) this MOCG also configures the QoS flows in the 5G network to allow best performance for the media devices. For spectrum management a shared access service allows user interaction.

All audio devices and the 5G network itself are synchronized to a common time base to avoid increasing jitter through drifting.

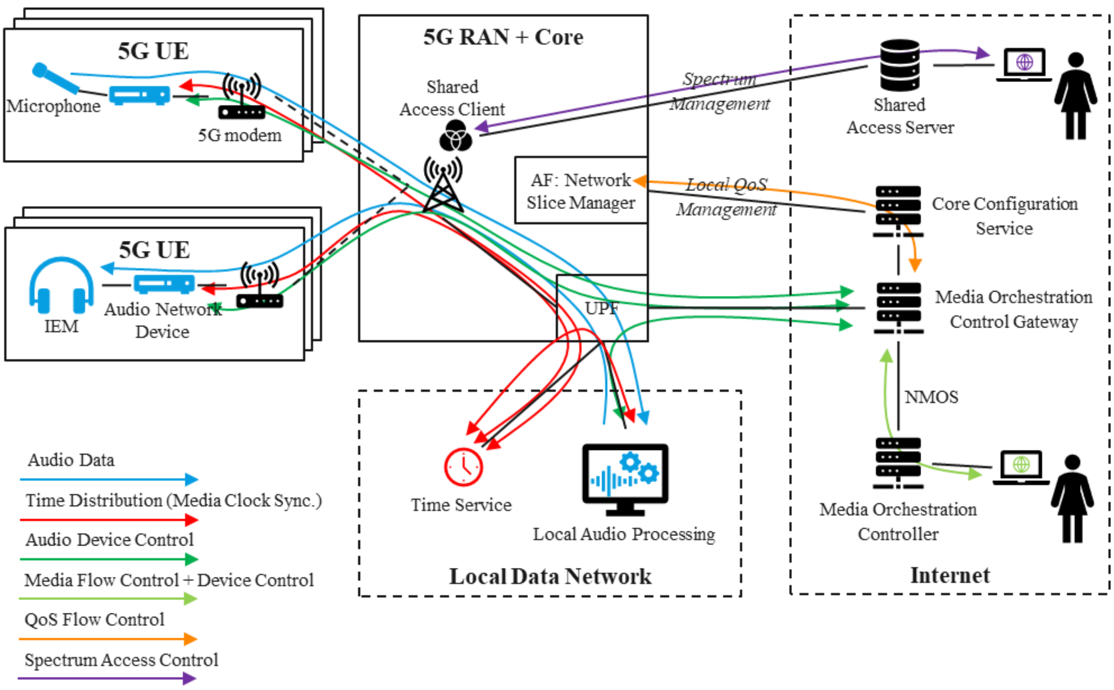


Figure 1: High level architecture of use case 1 (live audio production)

3.2 Control Data Flows

The control data flows of this use case are described in this section.

3.2.1 Setup 5G Network and connect UEs

To connect a UE to the local 5G network first the 5GS itself has to be configured properly. Figure 2 shows the necessary steps. A spectrum access server (SAS) gets a long-term frequency lease from the national regulator. When the 5GS powers up a list of channels is negotiated with this SAS server. When the RAN is operational the UE can connect and establish the default PDU session for basic connectivity.

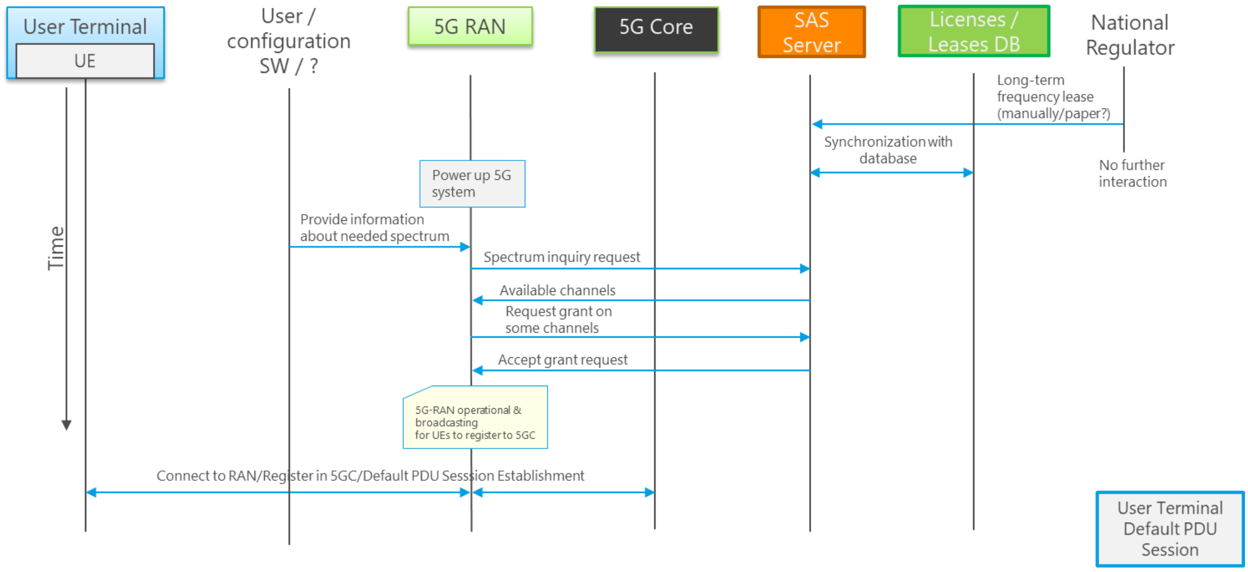


Figure 2: Data flow for setup 5G network and connect UE

3.2.2 Discovery of Audio Devices and establishing Audio Data Stream

When audio devices connect to the network, they need to be known to the media orchestration and control layer. Figure 3 shows this process. The audio devices announce their capabilities to the MOCG which allows the MOC to setup streams. When the MOC requests streams the core configuration service (CCS) request the required QoS flows from the 5GS. If the 5GS has capacities for this flows a new QoS-PDU session per UE is created that should carry the audio payload. The MOCG configures the audio devices and the MOC shows the user that everything is up and running.

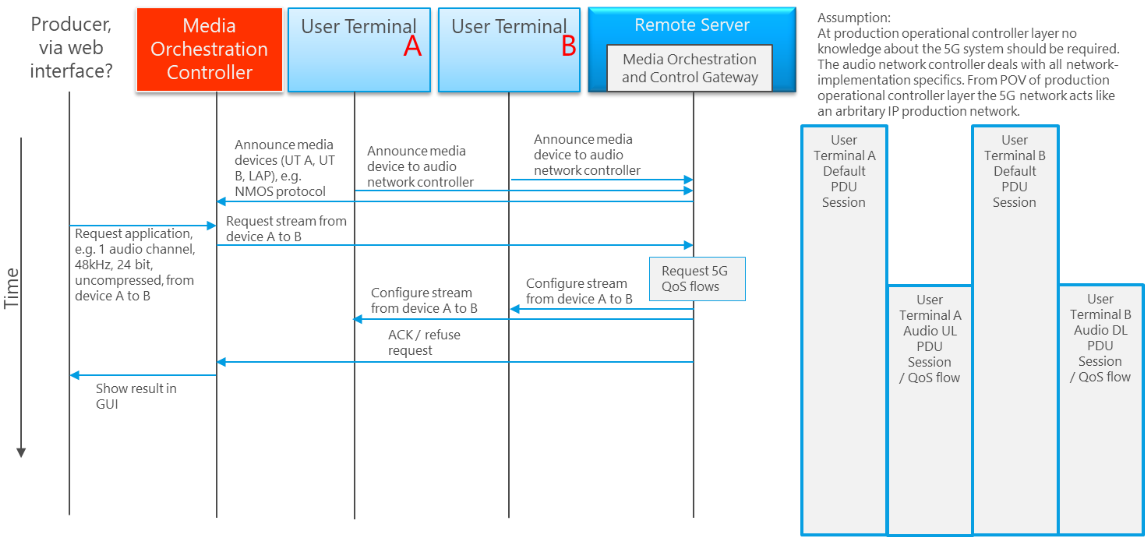


Figure 3: Data flow for device discovery and audio stream establishment

The process of requesting QoS flows from the 5GS is shown in Figure 4. Therefore, the CCS requests a QoS flow form the network slice manager (NSM) which checks the availability of radio resources with the shared access client (SAC) in the RAN.

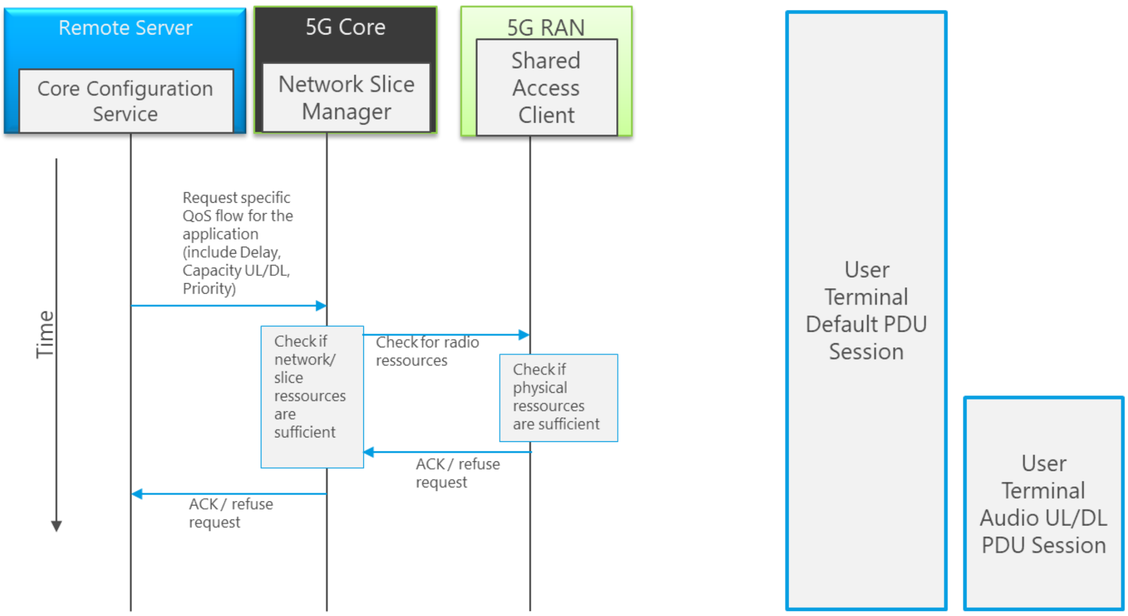


Figure 4: Data flow for requesting a QoS flow from the 5G network

3.3 Delay Budget

One key requirement for this use case is the streaming latency. For a deeper understanding of the influence of different components and to identify potential for optimization a detailed delay budget will be created. The impact on latency of the following functions will be identified either by calculations or measurements.

The delay budget for uplink includes the functions/parameters listed in Table 9.

Table 9: Delay budget for downlink and uplink audio transmission

Downlink	Uplink
Periodicity of transfers (related to 5GS)	Periodicity of transfers (related to 5GS)
UE layers processing (assuming RLC UM)	DN to UPF networking
Slot boundary	UPF processing
Scheduling request delay (SR periodicity)	UPF to CU-UP networking
Radio propagation	CU-UP processing (inc. GTP-U)
RU/DU processing (inc. GTP-U)	CU-UP to DU networking
DU to CU-UP networking	DU/RU processing (inc. GTP-U)
CU-UP processing (inc. GTP-U)	Slot boundary
CU UP to UPF networking	Radio propagation
UPF processing	UE layers processing
UPF to DN networking	

3.4 5G End-to-end user plane protocol stacks

The end-to-end 5G user plane stacks involved in the communication between the audio source/output application on the UE side and the audio source/output application (local audio processing) on the Data Network side are shown in Figure 5. The figure shows the path of the user data packets across the different protocols involved. It is worth to note that although it is not explicitly shown with the red/blue lines, the protocols of the O-RAN Fronthaul interface would also be involved in a fully disaggregated DU and RU according to O-RAN Alliance specifications.

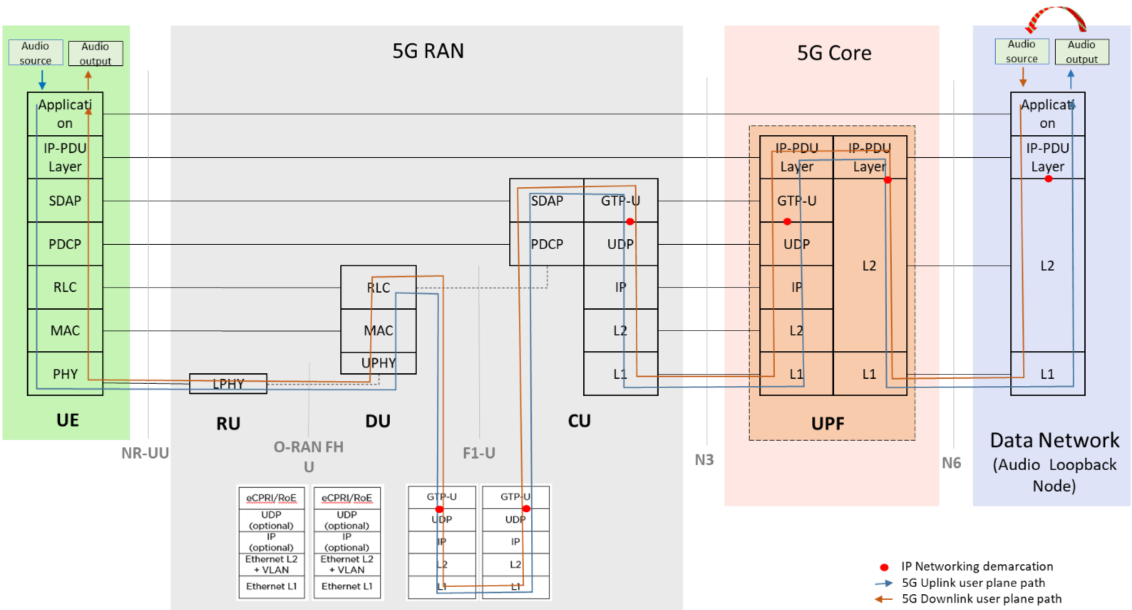


Figure 5: End-to-end user plane protocol stacks

3.5 List of Components

3.5.1 Component: Local Audio Processing

The local audio processing device is a stationary device connected wired to the 5G system. It is located on-site to allow low latency interaction with the wireless audio equipment. Figure 6 shows the architecture of the local audio processing device. Network audio streams from microphones are received, mixed and send to IEM receivers in the 5G network. Additionally, local analogue audio inputs could be used in the IEM-mix. Analog outputs allow to connect speakers or headphones for monitoring. Table 10 describes the functionalities of the local audio processing device.

The device is based on a custom developed hardware including an FPGA for signal processing and an ARM CPU that runs the device control software.

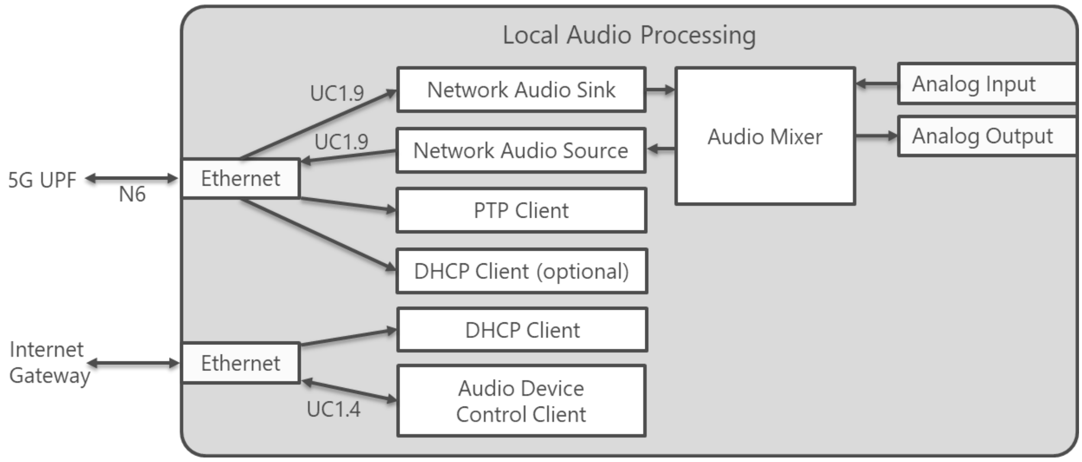


Figure 6: Block diagram of local audio processing component

Table 10: Functions of the UC1 local audio processing component.

Function	Description
Network Audio Source/Sink	The device can receive and transmit a network audio stream using interface/protocol UC1.9.
Audio Mixer	The device can mix local and network audio streams and forward them to local audio out or the network.
PTP Client	PTP client for synchronization
DHCP Client	The two ethernet interfaces must be supplied with an IP via DHCP. One has the option for a static IP setting.
Audio Device Control Client	Allows device control from the remote server via interface UC1.4.
Analog Input/Output	To add local audio sources to the mixes or to monitor via headphones/speakers

3.5.2 Component: Remote Server

The remote server is a piece of software that is installed locally or on the internet and allows to interface between the Media Orchestration Controller (MOC) and the audio devices in the local network. Figure 7 shows the architecture of the remote server component.

One main functionality is the Media Orchestration and Control Gateway (MOCG). It advertises the audio network devices capabilities to the MOC. The MOCG receives the desired configuration from the MOC and configures the audio network devices accordingly. The MOCG will set up and terminate audio data flows based on requests by the MOC. This configuration is done via the audio device control server that translates the MOC commands to the device specific protocol.

The MOCG configures the QoS flows in the 5GS via the Core Configuration Service. This translates the desired configuration from the MOC into network resource requirements and communicates these to the core (Network Slicing Manager).

Table 11 summarizes the functionalities of the remote server component.

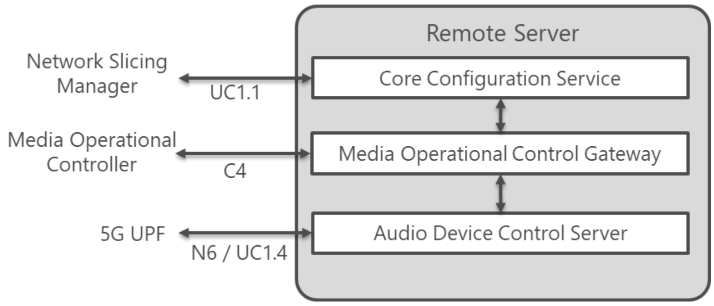


Figure 7: Block diagram of Remote Server

Table 11: Functions of the UC1 remote server component.

Function	Description
Media Orchestration Control Gateway	Interface between media devices, 5G network configuration and MOC
Core Configuration Service	Configures QoS flows in 5GS
Audio Device Control Server	Remote control of the audio devices

3.5.3 Component: Audio part of User Terminal

In the scenario 5G-enabled microphones and IEMs are used. Both of them are realized by a similar hardware/software implementation. Each of these Audio UEs consists of an audio part and a 5G part (the modem). The audio part of the UE is based on a custom developed hardware including an FPGA for signal processing and an ARM CPU that runs the device control software. The audio- and the 5G-part are connected by a 1 Gbit/s Ethernet connection.

The device can be configured to act as a microphone transmitter or an IEM receiver. Therefore, the device has analogue interfaces to connect a microphone and headphones. IP audio streams can be received or transmitted, and the audio network device converts between analogue and IP domain.

To control the functionality of the device it connects to a remote audio device control server. A PTP client for time synchronization allows media clock synchronization.

Figure 8 shows the architecture of the audio network device. An overview of its functionalities is given in Table 12.

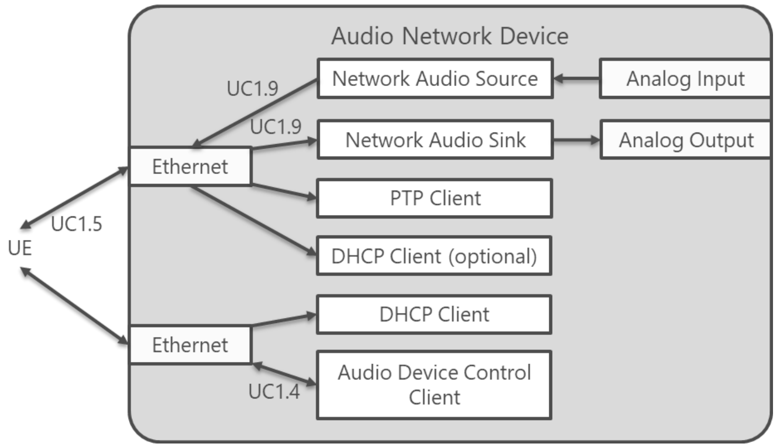


Figure 8: Audio Network device and UE block diagram

Table 12: Functions of the UC1 Audio UE component.

Function	Description
Network Audio Source/Sink	The device can receive and transmit a network audio stream using interface/protocol UC1.9.
PTP Client	PTP client for synchronization
DHCP Client	The two ethernet interfaces must be supplied with an IP via DHCP. One has the option for a static IP setting.
Audio Device Control Client	Allows device control from the remote server via interface UC1.4.
Analog Input/Output	To connect a microphone or a headphone.

3.5.4 Component: 5G part of User Terminal

EURECOM supplies 1 Ettus Research Universal Software Radio Peripheral (USRP) B210 software defined radio (SDR), and 2 commercial off-the-shelf (COTS) user terminals (UTs) (i.e., SIMCom Wireless Solutions Ltd. SIM8202G-M2 and Quectel Wireless Solutions Co., Ltd. RM500Q-GL) for the first phase. The features of each device are as follows:

Ettus Research USRP B210 SDR

A SDR platform is a radio communication system that allows the implementation of radio components handling signal processing via software running on general-purpose processor (e.g., personal computer or embedded system) rather than special-purpose hardware. The SDR cards to be used within the first phase of the 5G-RECORDS project are Ettus B210 USRPs from National Instruments.

The B210 is an entry-level multiple-input multiple-output (MIMO) SDR, which has been selected to enable common reproduction of the OpenAirInterface (OAI) code, an open-source software project founded by Eurecom that provides a fully 3GPP compliant reference implementation.

The B210 USRPs are designed for low-cost experimentation covering from 70 MHz – 6 GHz and combine the Analog Devices AD9361 RFIC direct-conversion transceiver providing up to 56 MHz of real-time RF bandwidth and an open and reprogrammable Spartan6 FPGA. Further, full support for the USRP Hardware Driver (UHD) software is available to immediately begin developing with GNU Radio. During the first phase of 5G-RECORDS, the USRPs B210 will be configured to operate in 3.4 GHz Band n78 with SCS 30 kHz and 10 MHz BWP aimed at achieving the lowest latency possible.

Figure 9: Ettus Research USRP B210 SDR Board

The B210 is connected through USB 3.0 to a GIGABYTE BRi5-8250 model mini PC with Intel core i5-8250U (4 cores @ 3.4 GHz max) which is capable of further incorporating a Sennheiser audio UT and a Sennheiser local audio processing engine into the system via USB 3.0, USB 3.1 or 1Gb Ethernet standards. Thus, there are three main interconnections on the PC: (i) one towards the radio, (ii) another for control of the Ethernet over USB3, and (iii) finally the application interconnection with the Sennheiser audio device. The corresponding devices will be connected over the monitoring interface(s) to a Eurecom's lab switch so that UC1 5G-RECORDS partners doing experimentations can go in and perform their tests.

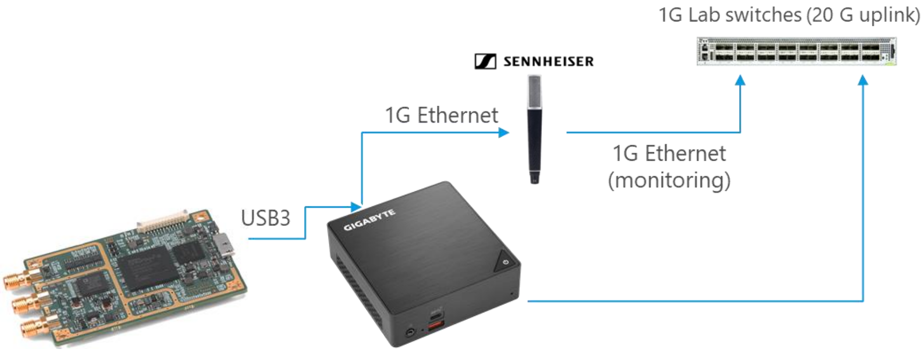


Figure 10: 5G-based UE prototype.

SIMCom Wireless Solutions Ltd. SIM8202G-M2

SIM8202G-M2 is a sub 6 GHz multi-band mobile communications module that supports the 3GPP Release 15 standalone (SA) 5G. EURECOM operates it within an industrial computer on a Pico-ITX form factor motherboard over the M.2 connector and using all 4 of its antenna ports.

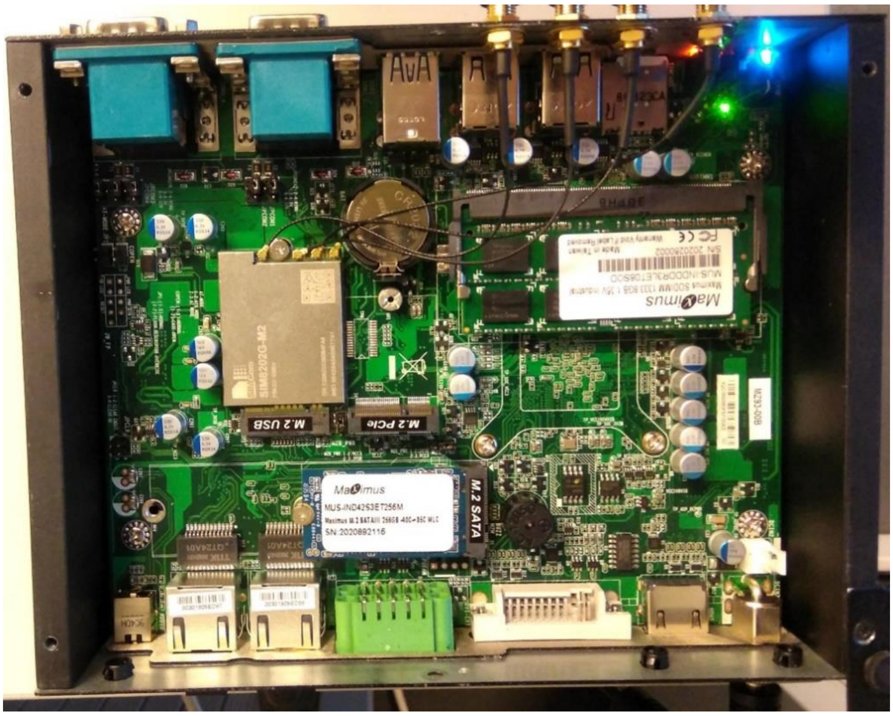


Figure 11: SIMCom SIM8202G-M2 module embedded on a Pico-ITX motherboard

Quectel Wireless Solutions Co., Ltd. RM500Q-GL

RM500Q-GL is another mobile communications module supporting the 3GPP Release 15 SA 5G New Radio (NR) specifications. It is employed over a credit-card sized "UP Board", and again, all 4 of its antenna interfaces are used.

Figure 12: Quectel RM500Q-GL module connected over USB to an "UP Board"

3.5.5 Component: 5G RAN

The overview of the NG-RAN architecture in 3GPP is depicted in Figure 13.

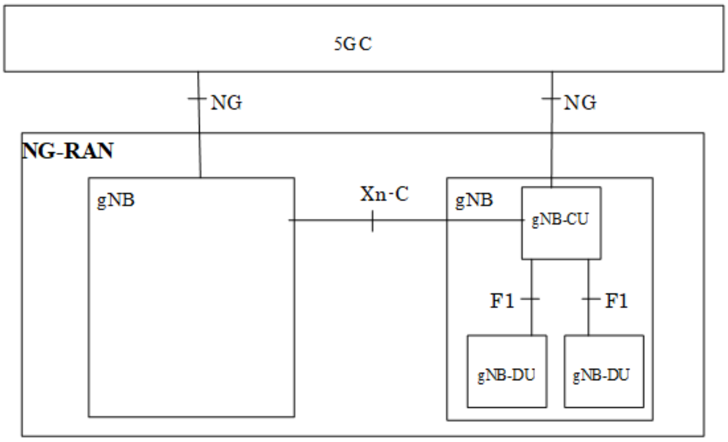


Figure 13: 3GPP NG-RAN architecture

3GPP 5GNR allows to split the gNB functionality into three logical modules: Radio Unit (RU), provisioned with RF circuitry; the Distributed Unit (gNB-DU), hosting gNB real-time functions; and the Centralized Unit (gNB-CU), hosting gNB non-real-time functions. 3GPP has defined a normative interface between the CU and DU components according to a HLS (High Layer Split) defined as F1. Additionally, CUPS (Control User Plane Separation) at CU level is enabled via normative E1 interface between CU CP and CU UP components as in Figure 14.

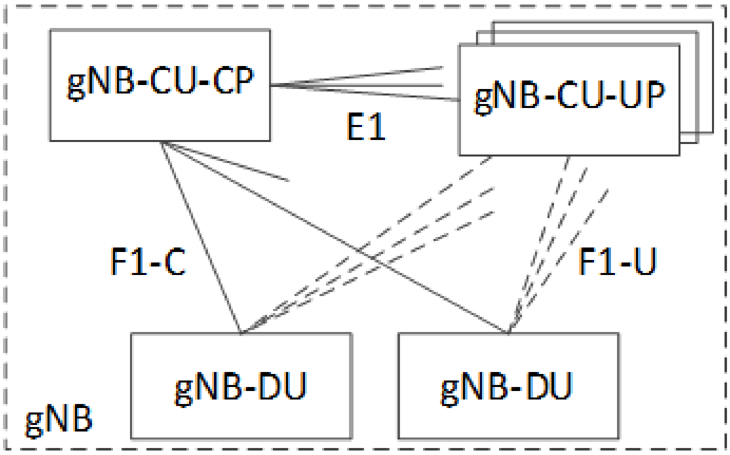


Figure 14: 3GPP gNB components

3GPP has not defined a normative interface for the Low Layer Split (LLS) between the DU and the RU but has recommended either one based on Split 7.2 (defined in the context of O-RAN Alliance) or Split 6 (defined in the context of Small Cell Forum). Different deployment options and topologies are therefore possible, some of which are shown in Figure 15.

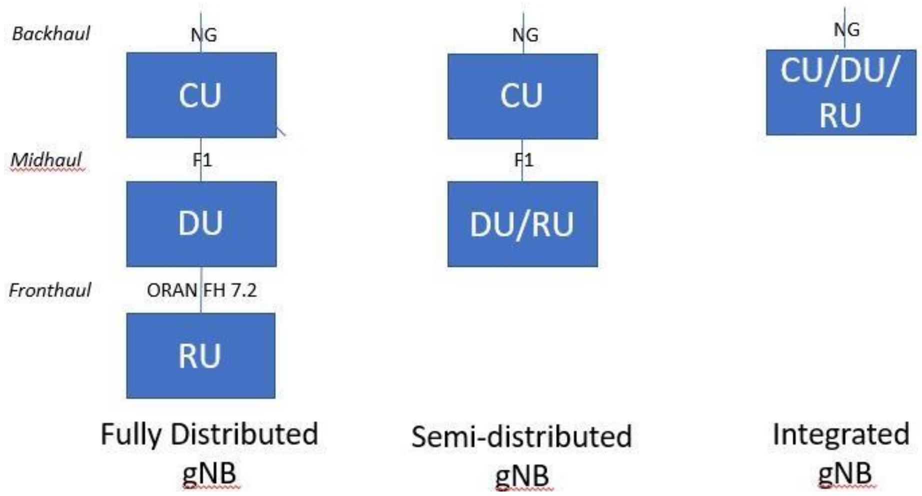


Figure 15: gNB deployment topologies

As part of O-RAN Alliance additional interfaces and components are being defined and standardized in addition to the standard 3GPP ones, with the general reference architecture shown in Figure 16.

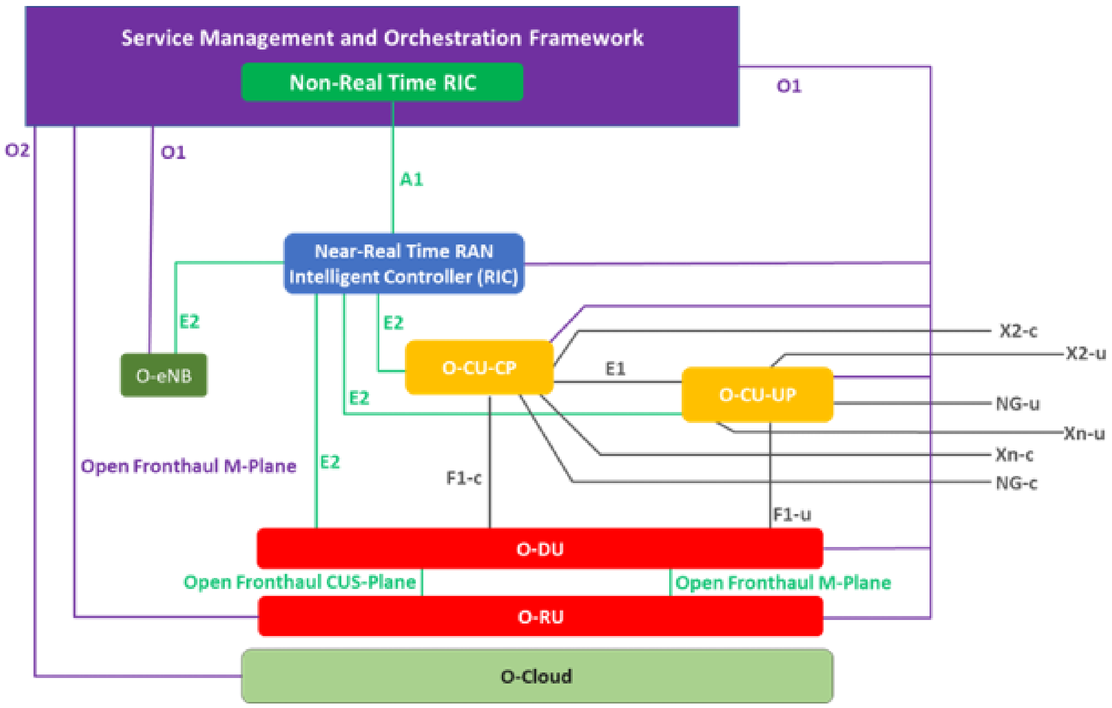


Figure 16: O-RAN Alliance reference architecture

3.5.5.1 *nRT-RIC, CU and Shared Access Spectrum client*

Accelleran brings dRAX™ is the marketing name for Accelleran´s virtual RAN product line, a cloud native and O-RAN aligned 5G SA vRAN solution consisting of a near-RT (nRT) RIC, CU-CP, CU-UP and xApp framework components. It is engineered to provide an open and extensible software framework for the control plane functions of 4G and 5G RAN and aligns with Open RAN architecture principles defined by both 3GPP and the O-RAN Alliance. dRAX™ is aligned to O-RAN architecture, but not all interfaces are fully interoperable, as most of them are still being defined in O-RAN Alliance.

dRAX™ is a genuinely cloud-native architecture based on containerised microservices communicating with each other via an asynchronous messaging framework. Each of the major components of the RAN (CU-CP, CU-UP, near-RT RIC) are themselves disaggregated into a fine-grained set of service entities. For example, the CU-CP is composed of a set of collaborating services handling: (i) F1AP connections to DU, (ii) E1AP connections to CU-UP, (iii) NGAP connections to AMF, and (iv) RRC connections to UEs.
The lifecycle (deployment, upgrade, scaling requirement) of these components are managed independently and they are unaware of location since the dRAX™ messaging framework handles routing of all messages between services.

The near-RT RIC is a key component of dRAX™. It supports the deployment of xApps (again as microservices) and provides them with a number of services in the context of the dRAX™ environment. dRAX™ leverages Accelleran field-proven RAN software and is compatible with Accelleran 4G CE certified carrier-grade small cells and supports 5G gNB using standards-based DU/RUs from the developing ecosystem of 5G Open RAN.

The key features of dRAX™ are summarized in Table 13.

Table 13: Key Features of dRAX

Key feautures of dRAX	Description
O-RAN aligned vRAN	The Accelleran dRAX™ vRAN platform delivers a true multi-vendor, disaggregated and open virtualized RAN Intelligence aligned with the O-RAN Alliance. Implementing 3GPP Control User Plane Separation (CUPS), the user and control planes are fully decoupled. Support for 3rd party DUs, RUs & e/gNB encourage an open disaggregated eco-system to bring innovative 4G/5G products to the commercial market at very competitive price points.
Open Orchestration & Data APIs	dRAX™ is open. Orchestration supports the industry preferred NFV/SDN framework APIs above and the NIB (Network Information Base) data APIs support industry standards and best practices (NFV/SDN, O-RAN, OAM, 3GPP, Netconf/Yang, …).
4G and 5G SA support	dRAX™ is field proven today for LTE and is being integrated with 5G SA support, leveraging standards-compliant DU/RUs.
Extensible RIC xApps	dRAX™ provides an open platform for the development of customised RAN intelligence, either by the customer, Accelleran or 3rd parties
Scalability	dRAX™ can be implemented on a single microserver for the smallest edge cloud. At the other end of the scale, dRAX™ is designed to scale to clusters of hundreds of cells.
Mission Critical Reliability	All dRAX™ code is written to Accelleran's unique set of SW development standards based on established practices from safety critical industries.

Accelleran dRAX™ in Figure 17 will be enhanced to support the requirements of the live audio production use case and will be integrated with the other needed 5G components on the associated use case testbed, namely the 5GC from Cumucore, DU/RU from Eurecom and Shared Access Spectrum server from RED Technologies.

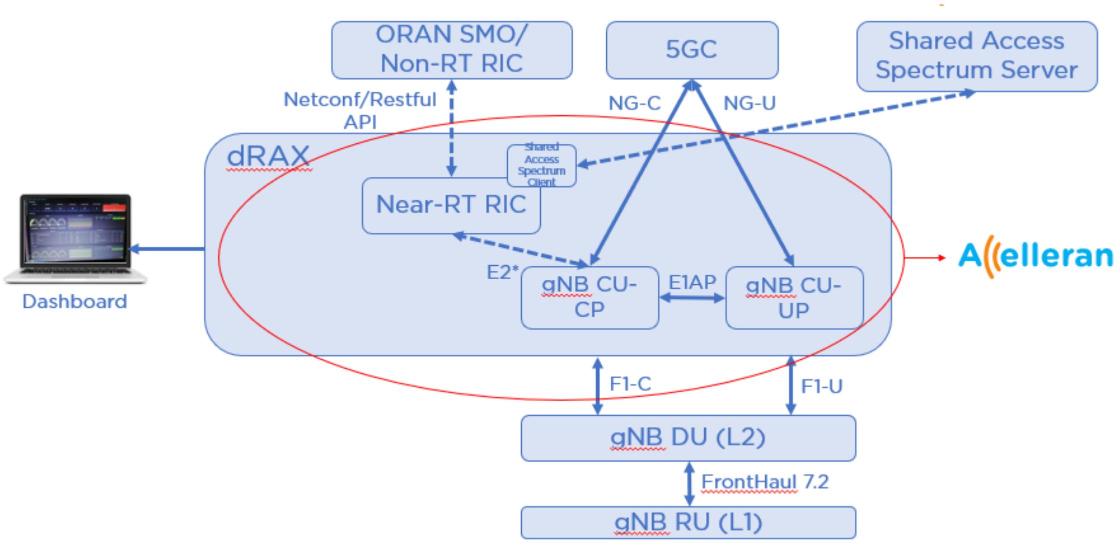


Figure 17: Accelleran dRAX™ in UC1 Live Audio Production

Near-RT RIC:

The near-RT RIC (Near Real-Time RAN Intelligent controller) is a key component of dRAX™. It has been developed as a true open development platform where 3rd parties can leverage dRAX™ open data and control knobs.

It supports the deployment of containerized xApps and provides them with a number of services in the context of the dRAX environment:

- xApp on-boarding and lifecycle management,
- Access to real-time RAN measurements and events,
- Configuration of RAN components,
- Real-time commands to direct RAN behaviour (e.g., force a handover, sub band masking),
- Real-Time state database,
- Inter xAppS communication,
- API-driven xApp configuration and policy management.

CU-CP and CU-UP:

The gNB-CU-CP is a logical node hosting the RRC and the control plane part of the PDCP protocol of the gNB-CU for a gNB. The gNB-CU-CP terminates the E1 interface connected with the gNB-CU-UP, the F1-C interface connected with the gNB-DU and the NG-C (N2) interface connected to the AMF in the 5GC.

The gNB-CU-UP is a logical node hosting the user plane part of the Packed Data Convergence Protocol (PDCP) protocol and the Service Data Adaptation Protocol (SDAP) protocol for a gNB. The gNB-CU-UP terminates the E1 interface connected with the gNB-CU-CP, the F1-U interface connected with the gNB-DU and the NG-U (N3) interface connected to the UPF in the 5GC.

Figure 18 shows the internal CU-CP and CU-UP components.

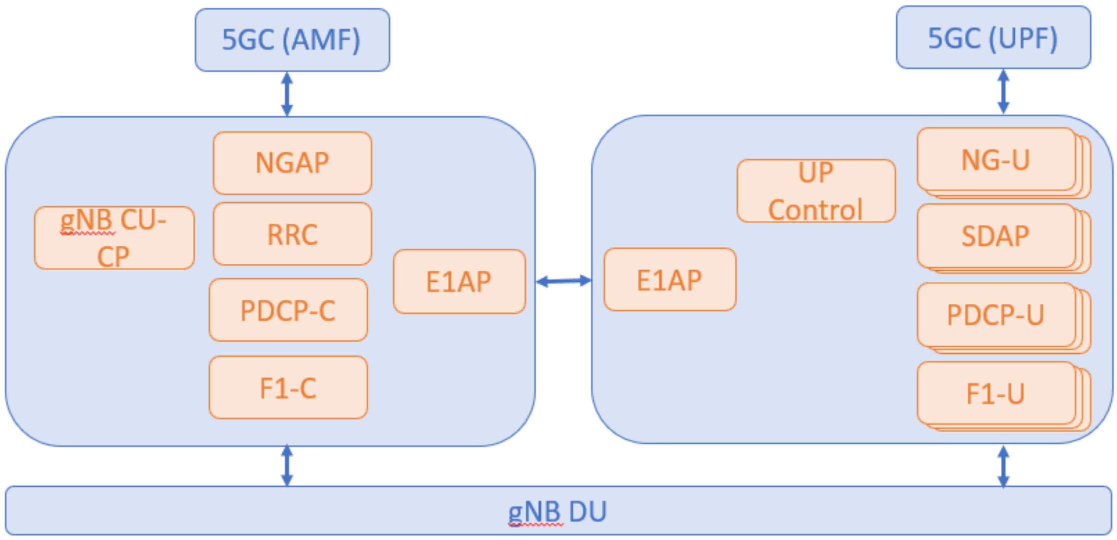


Figure 18: Accelleran CU components

Shared Access Spectrum client:

The Accelleran xApp framework will be used to enable and enhance Accelleran´s CBRS Shared Access Spectrum client based on LTE to support the 5G-NR functionality needed to communicate with RED Technology Shared Access Spectrum server for the acquisition of 5G shared spectrum dynamically.

3.5.5.2 *DU and CU*

OpenAirInterface (OAI) is the only open-source project today delivering implementations of both 5G NSA and SA RAN. OAI gNB is able to support an end-to-end SA setup based on a 5G CN (OAI or other) and SA capable COTS UE devices. As there are many developments taking place in parallel and capturing the whole RAN stack, intermediate validation steps have taken place using first the OAI UE (which is also developed in parallel to support SA deployments) in simulation and RF mode, and then the COTS UE devices.

The 5G SA access mode does not depend on legacy 4G LTE. However, it requires a new 5G core network (5GC). This new 5GC uses a cloud-aligned Service-Based Architecture (SBA) that supports control-plane function interaction, re-usability, flexible connections and service discovery that spans all functions. The main 5GC functions are AMF, SMF, NRF and UPF (SPGW-U-tiny), all of which have been implemented in OAI and can easily be deployed using docker-compose.

As per the 3GPP Specification Series 38, compared to NSA, in SA the gNB needs to also implement the complete RRC layer and handling of all the associated messages as well as the NGAP to interface with AMF (N2 interface) and UPF (N3 interface). Moreover, the gNB needs to support multiple bandwidth parts as the initial access happens only on the initial bandwidth part, which has a smaller bandwidth than the full cell bandwidth. Further support for contention based random access is needed, as well as support for common and dedicated control channels. Only after the initial connection and authentication with the AMF the full bandwidth part is configured and used for user-plane traffic.

From a deployment perspective, two options are provided for OAI gNB: The monolithic and the CU/DU functional split mode. As it has been already introduced before, the former option corresponds to a single gNB program on a single host running the whole 5G NR RAN stack. In the latter option, the OAI gNB portion is divided into two blocks: the Central Unit (CU) that contains the implementation of RRC and PDCP layers and the Distributed Unit (DU) that contains the implementation of RLC, MAC and PHY layers. The two units communicate with each other over the F1-C interface for the control plane and configuration exchanges based on F1AP protocol. The Downlink and Uplink user plane data transfer is made through the F1-U interface over GTP-U protocol. The CU and DU portions can thus run as separate programs in different hosts, offering significant flexibility for the deployment of the OAI 5G SA setup and the interoperability of OAI blocks with other commercial CUs or DUs.

Figure 19 depicts the 5G RAN protocol architecture of the OAI gNB according to the CU/DU functional split deployment. The layers that had to be extended in order to support end-to-end SA functionality are highlighted in yellow.

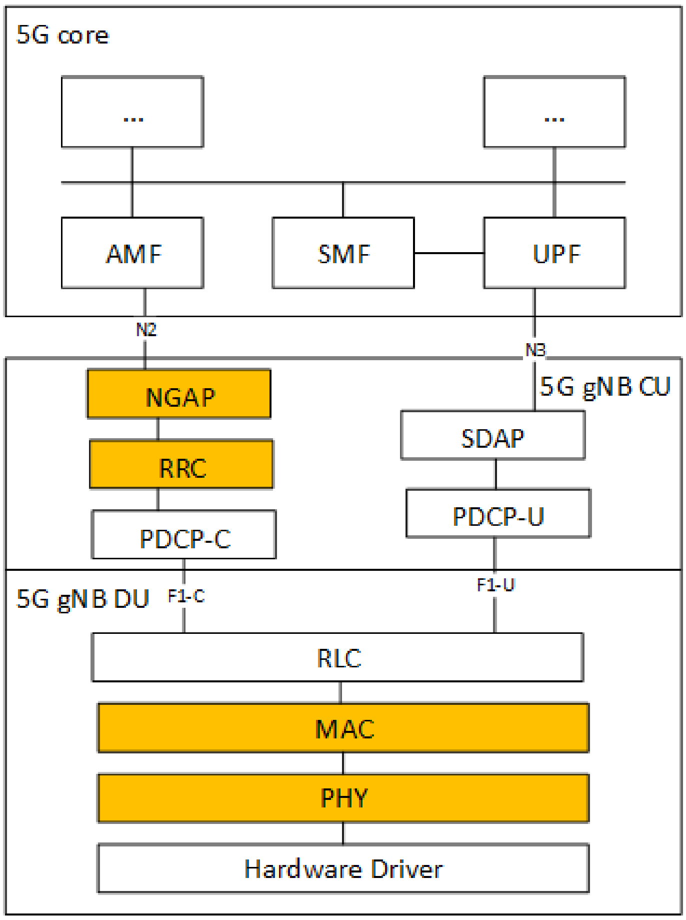


Figure 19: OAI gNB protocol architecture supporting 5G SA with CU/DU split

Interoperability tests of the OAI gNB with different 5G CNs and UE components from different vendors are ongoing. Specifically, interoperability with the OAI CN and the Nokia SA Box has been fully validated. With respect to the UE components, interoperability has been fully validated with the Quectel RM500Q-GL module and OAI UE.

In the following, some checkpoints for the validation of the end-to-end SA setup are provided to highlight the establishment of the 5G radio connection, the UE registration to the 5G CN, the PDU session establishment for the exchange of user plane traffic and some basic traffic test. The underlined tests were performed using the Quectel RM500Q-GL module.

After the UE synchronizes to the 5G cell and receives the System Information messages from the gNB, it initiates the contention based random access procedure (CBRA) in order to connect to the 5G cell. The procedure is finalized through the reception of Msg4 (*RRCSetup*) acknowledgment from the gNB (Figure 20). Then the UE replies with the *RRCSetupComplete* message which encapsulates the *NAS registration request* message towards the AMF. Upon reception of RRC Setup Complete, the UE state at the gNB becomes RRC Connected.

```
1170 [0m[0m[NR_MAC]   (ue 0, rnti 0x70bd) Received Ack of RA-Msg4. CBRA procedure succeeded!
1171 [0m[0m[34m[RLC]   RB found! (channel ID 1)
1172 [0m[0m[RLC]   /home/timarque/panos/openairinterface5g/openair2/LAYER2/nr_rlc/nr_rlc_oai_api.c:452:deliver_sdu: delivering SDU (rnti 28861 is_
     1 rb_id 1) size 92
1173 [0m[0m[NR_RRC]   Received message NR_RRC_DCCH_DATA_IND
1174 [0m[0m[NR_RRC]   Decoding DCCH : ue 28861, inst 0, ctxt 0x7f2ba7ffeeb0, size 86
1175 [0m12 c0 00 20 00 04 f7 e0 17 b6 ce d3 60 47 e0 04 14 90 00 bf 20 5f 51 00 10 00 00 00 00 02 [0m[NR_MAC]   884. 4 RNTI 70bd: 3 bytes from DC
     (ndata 3, remaining size 14)
1176 [0m[0m[NR_MAC]   884. 4 RNTI 70bd: 0 bytes from DCCH 1 (ndata 3, remaining size 8)
1177 [0mb2 e0 2f 07 07 10 03 07 e0 04 14 90 00 bf 20 5f 51 00 10 00 00 00 00 02 b1 00 10 02 e0 2f 07 02 f0 50 40 1d 14 3a 55 20 5f 51 00 00 00 11 80
     17 40 00 09 05 30 10 10
1178 [0m[NR_RRC]   [FRAME 00000][gNB][MOD 00][RNTI 70bd] [RAPROC] Logical Channel UL-DCCH, processing NR_RRCSetupComplete from UE (SRB1 Active)
1179 [0m[0m[NGAP]   [gNB 0] Build NGAP_NAS_FIRST_REQ adding in s_TMSI: GUAMI amf_set_id 0 amf_region_id 1 ue 70bd
1180 [0m[0m[NR_RRC]   [FRAME 00000][gNB][MOD 00][RNTI 70bd] UE State = NR_RRC_CONNECTED
```

Figure 20: Successful CBRA procedure and reception of RRC Setup Complete message at the gNB

This message is conveyed transparently from the gNB to the AMF through the NGAP *InitialUEMessage* (Figure 21). A sequence of NGAP/NAS messages are exchanged afterwards between the gNB, the UE and the AMF to perform the authentication and security procedures leading to the successful Registration of the UE to the AMF (*Registration Accept* and *Registration Complete* NAS messages).

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.18.203	192.168.69.131	NGAP	134	NGSetupRequest
2	0.003745	192.168.69.131	192.168.18.203	NGAP	614	NGSetupResponse
3	8.077278	192.168.18.203	192.168.69.131	NGAP/NAS-5GS	146	InitialUEMessage, Registration request
4	8.094364	192.168.69.131	192.168.18.203	NGAP/NAS-5GS	630	DownlinkNASTransport, Authentication request
5	8.123432	192.168.18.203	192.168.69.131	NGAP/NAS-5GS	146	UplinkNASTransport, Authentication response
6	8.126800	192.168.69.131	192.168.18.203	NGAP/NAS-5GS	462	DownlinkNASTransport, Security mode command
7	8.135349	192.168.18.203	192.168.69.131	NGAP/NAS-5GS/NAS-…	174	UplinkNASTransport, Security mode complete, Registration request
8	8.140345	192.168.69.131	192.168.18.203	NGAP/NAS-5GS	1302	InitialContextSetupRequest, Registration accept
9	8.257311	192.168.18.203	192.168.69.131	NGAP	122	UERadioCapabilityInfoIndication
10	8.459484	192.168.18.203	192.168.69.131	NGAP	86	InitialContextSetupResponse
11	9.338930	192.168.18.203	192.168.69.131	NGAP/NAS-5GS	118	UplinkNASTransport, Registration complete
12	9.341520	192.168.69.131	192.168.18.203	NGAP/NAS-5GS	710	DownlinkNASTransport, Configuration update command
13	9.341561	192.168.18.203	192.168.69.131	NGAP/NAS-5GS	162	UplinkNASTransport, UL NAS transport, PDU session establishment request
14	9.355758	192.168.69.131	192.168.18.203	NGAP/NAS-5GS	266	PDUSessionResourceSetupRequest, DL NAS transport, PDU session establishment accept
15	9.356056	192.168.18.203	192.168.69.131	NGAP	214	PDUSessionResourceSetupResponse

Figure 21: NGAP/NAS exchanges with the Core Network for UE Registration and PDU Session establishment

The UE then initiates the PDU Session Establishment which is validated through the *PDU Session Establishment Accept* NAS message coming from the CN. This message contains the IP address of the UE provided from the SMF. In Figure 22 the configured IP address is shown through the Quectel connection manager software.

```
[06-01 11:01:51:500] Quectel_QConnectManager_Linux_V1.6.0.16
[06-01 11:01:51:500] Find /sys/bus/usb/devices/1-1.3 idVendor=0x2c7c idProduct=0x800, bus=0x001, dev=0x014
[06-01 11:01:51:501] Auto find qmichannel = /dev/cdc-wdm0
[06-01 11:01:51:501] Auto find usbnet_adapter = wwan0
[06-01 11:01:51:501] netcard driver = qmi_wwan, driver version = 22-Aug-2005
[06-01 11:01:51:501] ioctl(0x89f3, qmap_settings) failed: Operation not supported, rc=-1
[06-01 11:01:51:501] Modem works in QMI mode
[06-01 11:01:51:508] cdc_wdm_fd = 7
[06-01 11:01:51:594] Get clientWDS = 15
[06-01 11:01:51:625] Get clientDMS = 1
[06-01 11:01:51:658] Get clientNAS = 4
[06-01 11:01:51:690] Get clientUIM = 1
[06-01 11:01:51:722] Get clientWDA = 1
[06-01 11:01:51:754] requestBaseBandVersion RM500QGLABR11A02M4G
[06-01 11:01:51:882] requestGetSIMStatus SIMStatus: SIM_ABSENT
[06-01 11:01:51:914] requestGetProfile[1] oai.ipv4///0
[06-01 11:01:51:946] requestRegistrationState2 MCC: 0, MNC: 0, PS: Detached, DataCap: UNKNOW
[06-01 11:01:51:977] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[06-01 11:01:51:977] ifconfig wwan0 0.0.0.0
[06-01 11:01:51:979] ifconfig wwan0 down
[06-01 11:03:22:967] requestRegistrationState2 MCC: 505, MNC: 1, PS: Detached, DataCap: UNKNOW
[06-01 11:03:22:999] requestRegistrationState2 MCC: 505, MNC: 1, PS: Detached, DataCap: UNKNOW
[06-01 11:03:23:031] requestRegistrationState2 MCC: 505, MNC: 1, PS: Detached, DataCap: UNKNOW
[06-01 11:03:41:088] requestRegistrationState2 MCC: 505, MNC: 1, PS: Detached, DataCap: UNKNOW
[06-01 11:03:41:729] requestRegistrationState2 MCC: 505, MNC: 1, PS: Attached, DataCap: 5G_SA
[06-01 11:03:41:985] requestSetupDataCall WdsConnectionIPv4Handle: 0x3b50c6b0
[06-01 11:03:42:113] ifconfig wwan0 up
[06-01 11:03:42:114] busybox udhcpc -f -n -q -t 5 -i wwan0
udhcpc: started, v1.30.1
udhcpc: sending discover
udhcpc: sending select for 192.198.0.2
udhcpc: lease of 192.198.0.2 obtained, lease time 7200
[06-01 11:03:42:190] ./netup wwan0 192.198.0.2
```

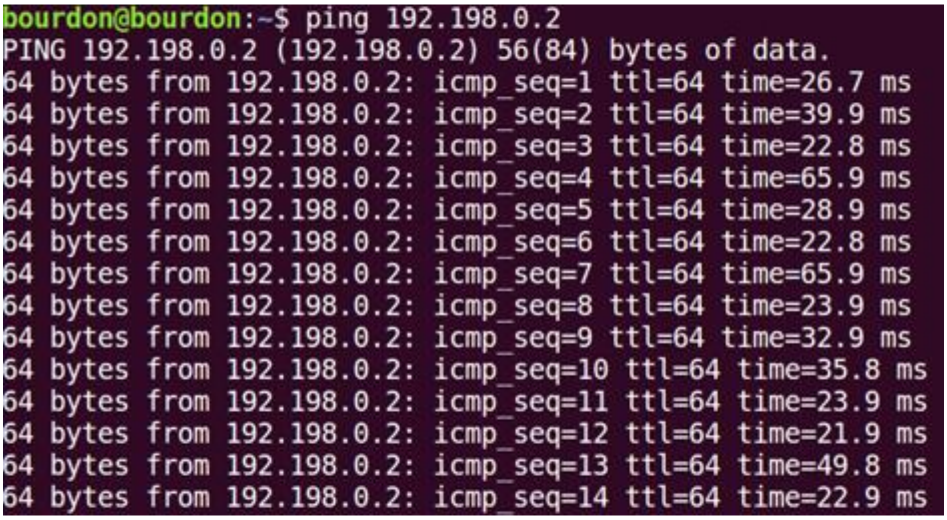
Figure 22: Quectel module connection manager

At the same time, the gNB also sends a dedicated *RRCReconfiguration* message to the UE containing the configuration of the DRB that should be established at the UE to enable the user plane traffic flow at the RAN stack (PDCP, RLC, MAC layers). The UE replies with a *RRCReconfigurationComplete* message to signal the successful reconfiguration. Upon reception of the Reconfiguration Complete message the gNB performs its own DRB configuration for the lower layers, as can be seen from the figure above. In parallel, the gNB establishes a GTP-U tunnel with the UPF to enable the user-plane traffic flow over the N3 interface.

```
[0m[NR_RRC]   Receive RRC Reconfiguration Complete message UE 70bd
[0m[0m[NR_RRC]   Configuring PDCP DRBs/SRBs for UE 70bd
[0m[0m[PDCP]   /home/timarque/panos/openairinterface5g/openair2/LAYER2/nr_pdcp/nr_pdcp_oai_api.c:add_drb:878: added DRB for UE RNTI 70bd
[0m[0m[NR_MAC]   Modified UE_id 0/70bd with CellGroup
[0m[0m[NR_MAC]   Adding SchedulingRequestconfig
[0m[0m[NR_MAC]   Adding BSR config
[0m[0m[NR_MAC]   Adding TAG config
[0m[0m[NR_MAC]   Adding PHR config
[0m[0m[NR_MAC]   Adding LCID 1 (SRB 1)
[0m[0m[NR_MAC]   Adding LCID 2 (SRB 2)
[0m[0m[NR_MAC]   Adding LCID 4 (DRB 4)
[0m[0m[NR_RRC]   Configuring RLC DRBs/SRBs for UE 70bd
[0m[0m[RLC]   Trying to add SRB 2
[0m[0m[RLC]   /home/timarque/panos/openairinterface5g/openair2/LAYER2/nr_rlc/nr_rlc_oai_api.c:710:add_rlc_srb: added srb 2 to UE with RNTI
[0m[0m[34m[RLC]   /home/timarque/panos/openairinterface5g/openair2/LAYER2/nr_rlc/nr_rlc_oai_api.c:792:add_drb_am: added drb 1 to UE with
0x70bd
[0m[0m[RLC]   /home/timarque/panos/openairinterface5g/openair2/LAYER2/nr_rlc/nr_rlc_oai_api.c:add_drb:879: added DRB to UE with RNTI 0x70bc
[0m[0m[NR_RRC]   [gNB 0] Frame      0 CC 0 : SRB2 is now active
[0m[0m[NR_RRC]   [gNB 0] Frame  0 : Logical Channel UL-DCCH, Received NR_RRCReconfigurationComplete from UE rnti 70bd, reconfiguring DRB 1
```

Figure 23: DRB establishment at the gNB upon reception of RRC Reconfiguration Complete message

After these steps the UE can exchange IP traffic through the CN. Finally, a ping test initiated from the CN towards the UE is shown in the figure below.



```
bourdon@bourdon:~$ ping 192.198.0.2
PING 192.198.0.2 (192.198.0.2) 56(84) bytes of data.
64 bytes from 192.198.0.2: icmp_seq=1 ttl=64 time=26.7 ms
64 bytes from 192.198.0.2: icmp_seq=2 ttl=64 time=39.9 ms
64 bytes from 192.198.0.2: icmp_seq=3 ttl=64 time=22.8 ms
64 bytes from 192.198.0.2: icmp_seq=4 ttl=64 time=65.9 ms
64 bytes from 192.198.0.2: icmp_seq=5 ttl=64 time=28.9 ms
64 bytes from 192.198.0.2: icmp_seq=6 ttl=64 time=22.8 ms
64 bytes from 192.198.0.2: icmp_seq=7 ttl=64 time=65.9 ms
64 bytes from 192.198.0.2: icmp_seq=8 ttl=64 time=23.9 ms
64 bytes from 192.198.0.2: icmp_seq=9 ttl=64 time=32.9 ms
64 bytes from 192.198.0.2: icmp_seq=10 ttl=64 time=35.8 ms
64 bytes from 192.198.0.2: icmp_seq=11 ttl=64 time=23.9 ms
64 bytes from 192.198.0.2: icmp_seq=12 ttl=64 time=21.9 ms
64 bytes from 192.198.0.2: icmp_seq=13 ttl=64 time=49.8 ms
64 bytes from 192.198.0.2: icmp_seq=14 ttl=64 time=22.9 ms
```

Figure 24: Ping test for user plane traffic with the OAI SA setup

3.5.6 Component: 5G Core

Cumucore develops dynamic network features to meet the 5G-RECORD project requirements. For this purpose, Cumucore develops the capability to connect local data network directly to UPF, which enables local audio mixing within UC1. Besides, the 5G network virtual functionalities are further developed to handle Network Slice lifecycle functionality. Cumucore also develops PCF functionality to be able to dynamically create data flows in 5G network and provides Cumucore Network Wizard to add and remove data flows using external API.

Cumucore 5G SA network

3GPP Release 15 already includes the specifications on the 5G Core (5GC) Network, in TS 23.501 [3]. The 5GC follows a number of principles that are mainly targeted for reaching higher flexibility, supporting many different use cases. This includes the introduction of service-based principles, where network functions provide services to each other. A clean control plane/user plane split allows independent scaling of control plane and user plane functions, and also supports flexible deployments in terms of where the user plane can run (this principle was, in fact, already introduced in EPC in Release 14). The architecture allows for different network configurations in different network slices.

The 5GC control plane is based on the Service Based Architecture (SBA). In SBA, the network functions communicate with each other via a logical communication bus and network functions can provide services to each other. A network function instance is registered to a Network Repository Function (NRF). Using the NRF, a network function instance can find other network function instances providing a certain service. The goal of such architecture is to get a higher flexibility in the overall system, and to make it easier to introduce new services.

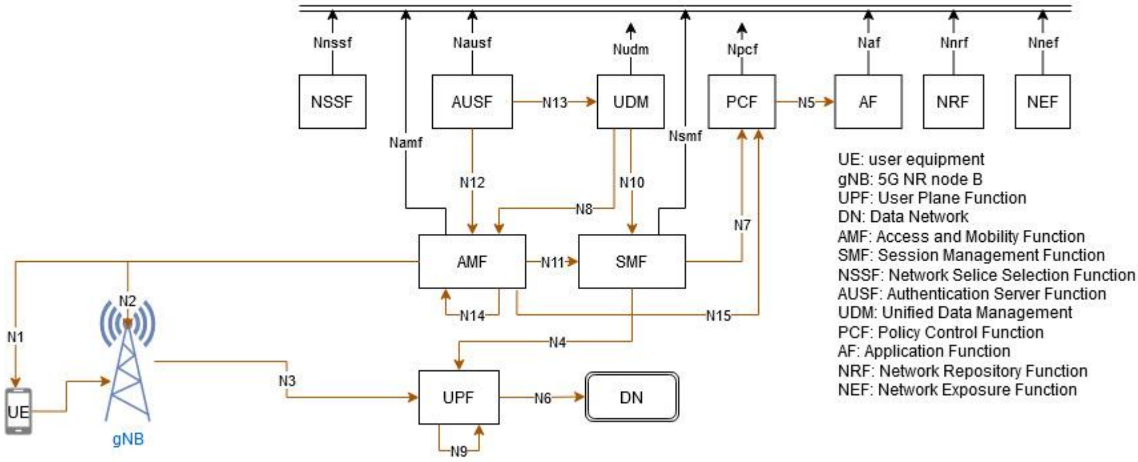


Figure 25: 5G Core architecture overview

A detailed description of 5G core network features within 5G-RECORDS (i.e., service-based architecture, standalone and non-standalone architectures, QoS, network slicing, non-public networks, edge cloud, and network exposure) is presented in *Annex A: 5G Network,* including the definition of each network function.

In particular, the AF in 5G-RECORDS UC1 would consist of specific media related applications that have to interact with 5GC to request network slice with specific requirements for traffic. Moreover, additional AF from other media services can request MEC capabilities to run specific audio/video processing.

Cumucore Network Wizard

Cumucore also provides Cumucore Network Wizard that has a capability to add and remove data flows using external API. This functionality is also used in the UC2, so further details can be found in 3.5.9.

3.5.7 Component: Shared Access Server

RED Technologies brings the Shared Access Server for the UC1, whose main features are presented in this section.

SAS server architecture

The SAS server is responsible for determining the maximum allowed transmission power of each device. Figure 26 shows the architecture of the SAS server.

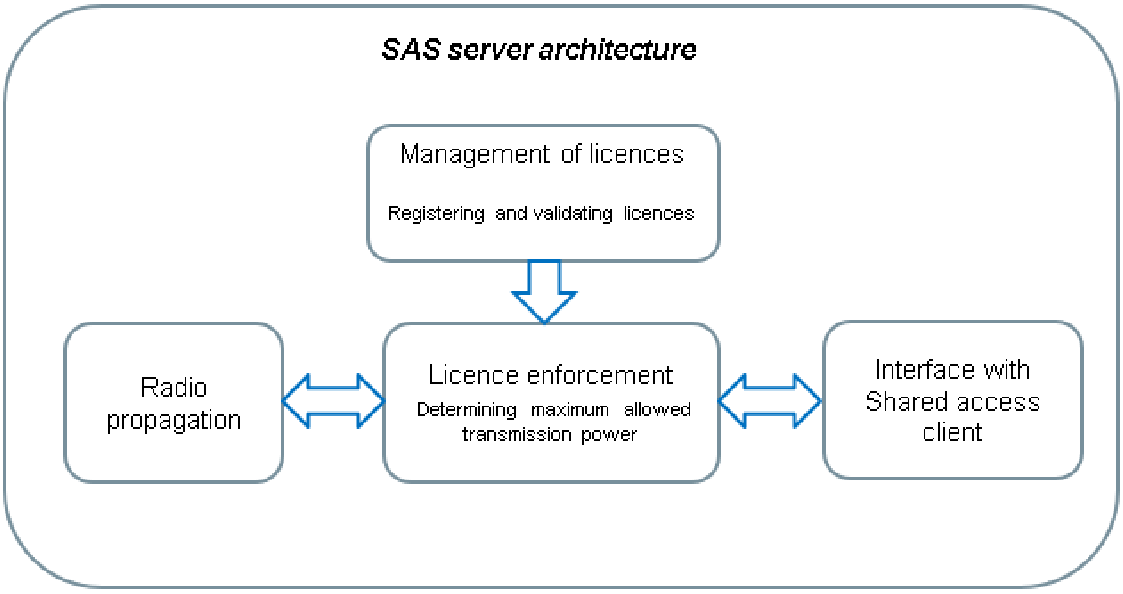


Figure 26: SAS Server Architecture

Management of licenses

A **licence** is associated with: (i) a licence identifier, (ii) an administrative area identifier, (iii) a frequency range, (iv) an expiration date, (v) a licensee identity, and (vi) an interference protection threshold (in dBm/MHz).

A **lease** is associated with: (i) a lease identifier, (ii) a lease area, (iii) a frequency range, (iv) an expiration date, (v) a lessee identity, (vi) the identifier of the licence associated with this lease, and (vii) an interference protection threshold (in dBm/MHz).

The owner of a lease or a licence is allowed to create protection zones within the area associated with this lease or licence.

Based on the planned deployment, desired transmission power, and radio environment the SAS determines the protection zone contour and identifies the most suitable frequencies.

License and lease enforcement / protection from interference

For each protection zone (associated with a license or a lease), the SAS server ensures that the aggregation of emissions from all devices (e.g., all 5G small cells) located outside this zone is below the interference threshold associated with this licence.

The level of interference at given point within the protection zone is determined as follows:

1. Computing interference of each device located outside of this protection zone.
    a. Computing the path loss between each device and this given point (considering height, azimuth, beamwidth).
    b. Adding the transmission power currently allowed for this device.
2. Aggregating all interferences computed in 1).

PMSE setup operation

The overall flow to allow a PMSE to operate is shown below.

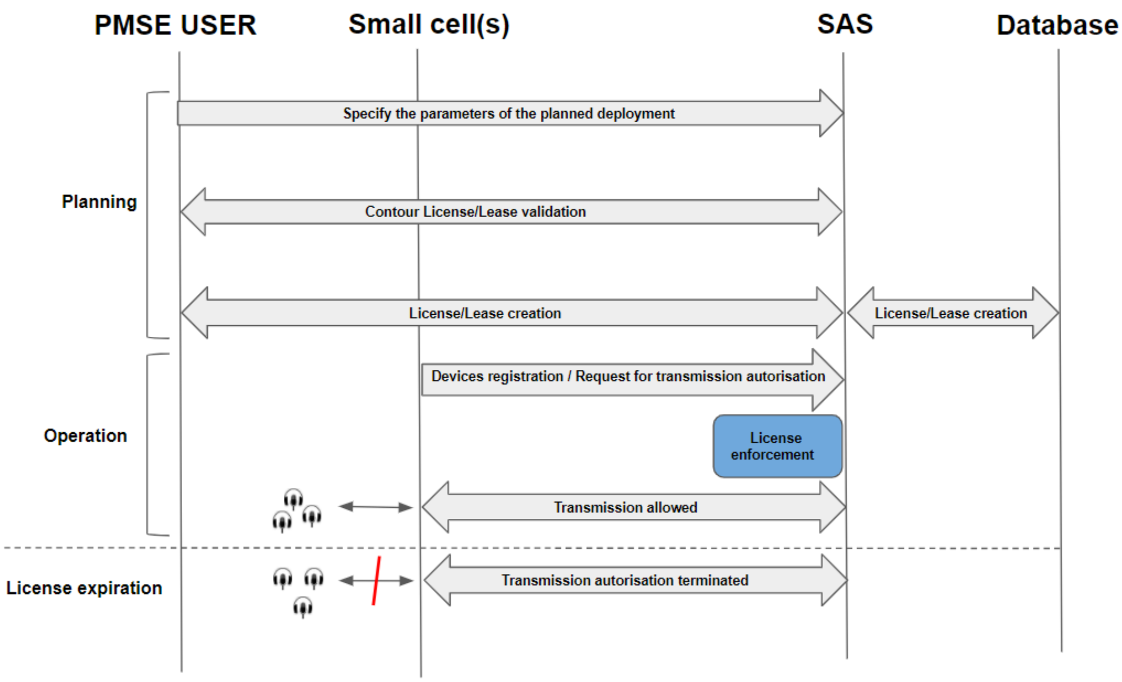


Figure 27: PMSE setup operation

3.5.8 Component: Time Service

Media networks use time synchronization to synchronize media clocks in order to avoid buffer over- or underruns and to maintain best audio quality. In IP-based media networks time synchronization is achieved by deploying PTP clients and server [4]. In a wired scenario this is typically achieved by providing a wired wordclock signal either via BNC or via PTP over Ethernet/IP.

During the first integration phase the PTP time service (the PTP server) will be a distinct service that bypasses the 5GS. In further steps the time information should be provided through the 5GS, and the time service could also be a function in the 5GS.

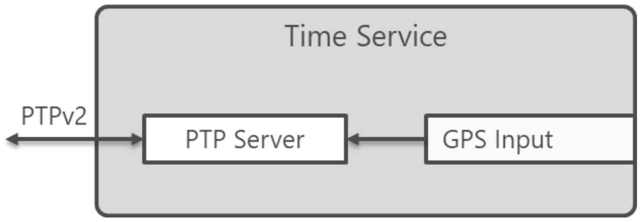


Figure 28. Block diagram of time service component

3.5.9 Component: Network Slice Management

Cumucore Network Wizard manages Network Slices and usage of network slice on data flow level. There is a user interface that can be used to create, operate, and terminate Network Slices. In the 5G-RECORDS project, APIs have been developed which are used to generate dataflows with specific QoS settings and related SLAs.

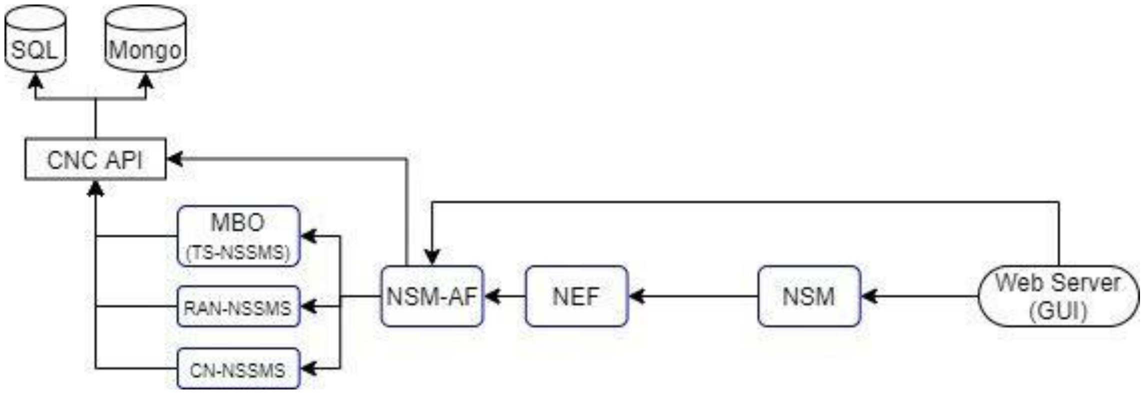


Figure 29: Block Diagram of Network Slice Manager

The Network Slice Management is done through an Application Function which can be called NSM-AF and in 3GPP specifications is called as Network Slice Management Service (NSMS) or Network Slice Subnet Management Service (NSSMS) that support different use cases defined in TS 28.531. A slice subnet is considered a different segment of the end-to-end system e.g., RAN subnet, Transport subnet, Core subnet.

The NSMS/NSM-AF can be and internal function only accessible from other network functions or from internal management console (i.e., GUI). The NSMS/NSM-AF can be made accessible from external applications through Network Exposure function (NEF).

NSMS/NSM-AF provides an interface to create, activate, terminate slices in the network and perform feasibility checks before creating new slices. The NSMS/NSM-AF will interact with different modules to create, delete, modify the network slices.

NSMS/NSM-AF will interact with the so-called Transport Network (TN) manager, which is equivalent to the Mobile Backhaul Orchestrator (MBO) to reconfigure the Transport Network to support new network slices. The NSMS/NSM-AF sends the transport network related requirements (e.g., external connection point, latency and bandwidth) to the TN Manager which reconfigures the TN accordingly (TS 28.531: section 5.1.1)

NSMS will interact with the other network or radio orchestrators to allocate the resources on different subnets for creating the new slice. NSMS/NSM-AF will interact with Radio Resource Manager (RRM) to configure radio cells for the slice. NSMS/NSM-AF will interact with Network Function Management Service or Network Function Virtualization Orchestrator (NFVO) to allocate network functions required for the slice.

The NSMS/NSM-AF will also perform feasibility checks to confirm resources available for network slices through the NWDAF and other subnet management modules like MBO or RRM.

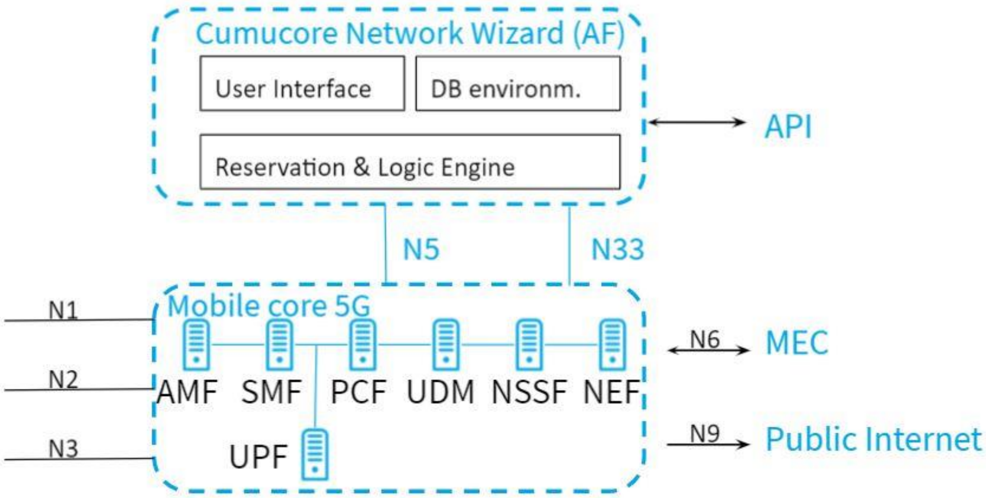


Figure 30: Interfaces and Interactions among modules

3.6 Interfaces

Figure 31 depicts the UC1 interfaces within the overall E2E infrastructure.

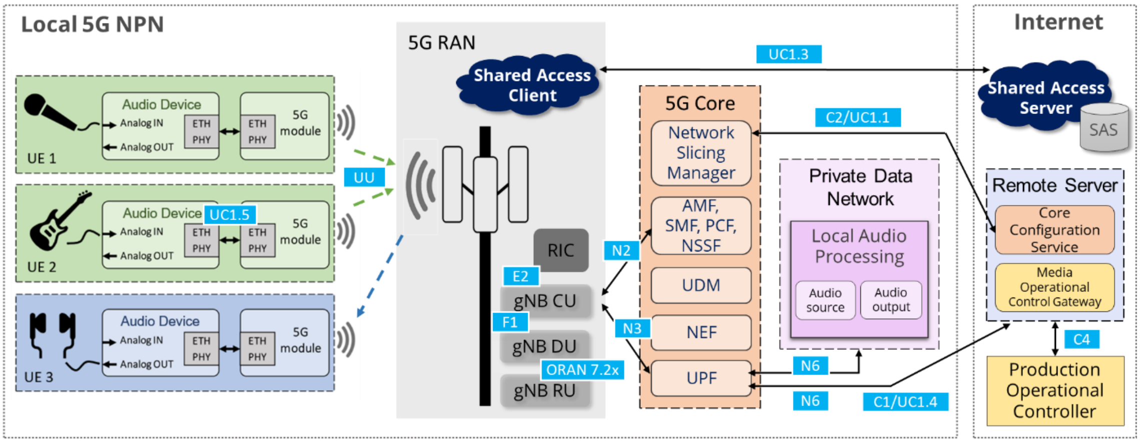


Figure 31: UC1 Interfaces

3.6.1 Analogue Input

Analog audio input to connect a microphone or line signal.

3.6.2 Analogue Output

Analog audio output to connect a headphone or a PA system (Public Address system).

3.6.3 GPS Input

The GPS (Global Positioning System) input is used as a reference for providing the PTP service.

3.6.4 [UC1.1] Core Configuration Protocol

REST API to request application specific data flows for media devices.

3.6.5 [UC1.4] Audio Device Control

A proprietary TCP-based protocol to control audio devices. It is an outbound connection initiated by the audio devices towards a remote server. The protocol allows to control audio settings as well as installation of new software.

3.6.6 [UC1.5] Audio Network Device to UE Interface

Physical: 1 Gigabit Ethernet.

3.6.7 [UC1.9] Network Audio Protocol

For network audio a custom IP/UDP-based audio transmission protocol is used. It is similar to RTP (Real Time Protocol) and contains beside the actual audio data timestamps and sample indexes. In case of time synchronization between source and sink this allows for calculation of an absolute one-way transmission latency.

3.6.8 Near-RT RIC

In the context of the 5G-RECORDS UC1, the Near-RT RIC will be managed and configured via the dRAX dashboard which embeds autonomous SMO functionality, i.e., no external SMO will be required. The near-RT RIC controls the CU CP and CU UP components via Netconf/RESTful APIs and an Accelleran dRAX databus based on NATS and Kafka technologies.

3.6.9 [N2, N3, F1] CU

The CU CP interfaces northbound with the 5G Core AMF via the 3GPP-standardized NG-AP (N2) interface based on SCTP/IP protocols using Ethernet Datalink/Physical layers. The CU CP interfaces southbound with the DU via the 3GPP-standardized F1-C (F1AP) interface based on SCTP/IP protocols using an Ethernet based Datalink/Physical layer.

The CU UP interfaces northbound with the 5G Core UPF via the 3GPP-standardized NG-U (N3) interface based on GTP-U/UDP/IP protocols using an Ethernet based Datalink/Physical layer. The CU UP interfaces southbound with the DU via the 3GPP-standardized F1-U interface based on GTP-U/UDP/IP protocols using an Ethernet based Datalink/Physical layer.

The CU CP and CU UP interface each other via the 3GPP-standardized E1 (E1AP) interface based on SCTP/IP protocols using Ethernet Datalink/Physical layers.

3.6.10 Spectrum Sharing Interfaces

Figure 32 presents the specific interfaces within the spectrum sharing component.

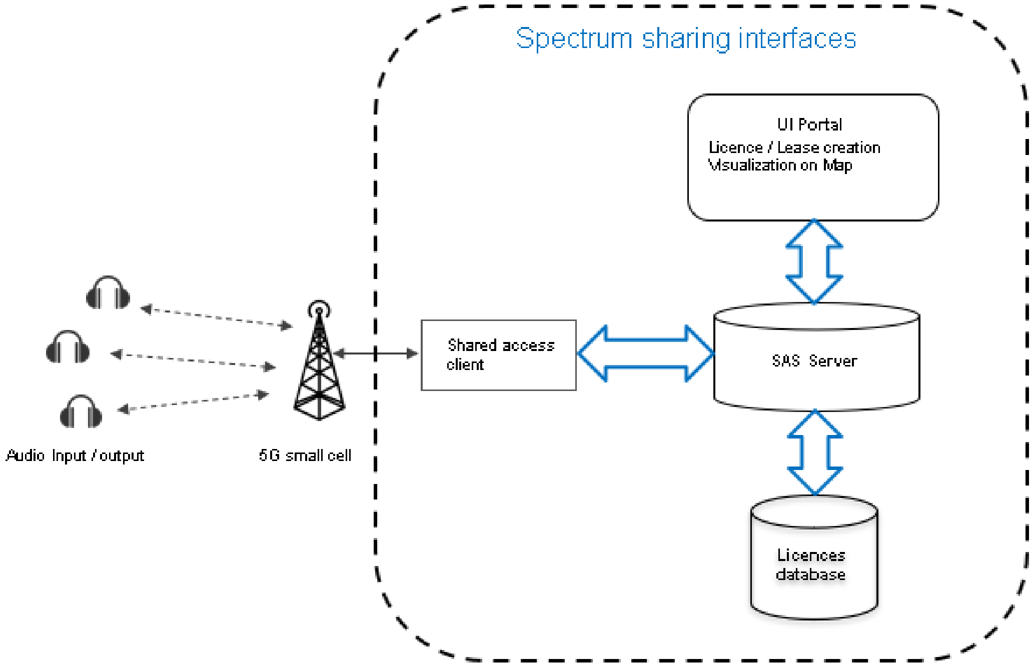


Figure 32: Spectrum Sharing Interfaces

3.6.10.1 *[UC1.3] SAS Server – Shared access client*

The SAS client interfaces with the SAS server using a JSON-based Winnforum protocol over a secure connection using TLS v1.2.

1. Device registration (e.g., 5G Small Cell): Reception of parameters from deployed devices required for interference protection including localization and antenna parameters.
2. Handling of requests from a device to access spectrum: Reception of the frequency range and power over which to operate.
3. Real-time control of spectrum grants: The SAS can receive heartbeats from the devices at regular intervals.

This enables to move the station to another frequency when a more suitable frequency is identified and to update the transmission power as needed (e.g., to provide a higher maximum allowed power)

3.6.10.2 *Database of licenses*

The SAS server is connected to a database containing the list of active licenses and leases.

3.6.10.3 *UI portal*

A UI portal enables users to connect to the SAS server from a web browser in order to: (i) monitor spectrum sharing activity on a map, and visualize the following elements: licenses, leases, protection zones, devices deployment, and active spectrum grants; (ii) create protection zones within licenses and leases

3.6.11 [N6] 5G Core to Data Network Interface

Physical: 1 Gigabit Ethernet.

Routing traffic between this device and the internet gateway must be possible.

4 UC2 Components - Multiple Camera Wireless Studio

4.1 General Architecture

The Multiple Camera Wireless Studio use case is based around a multi-camera audio and video production in a professional environment.

This use case will aim to the integration of a media production system, with up to 5 wireless cameras that will replicate existing technologies such as COFDM radio cameras in terms of performance and capabilities using 5G. Multiple location scenarios with production facilities local to an event as well as remote and distributed production models will be explored.

In additional scenarios we expect to integrate 5G based contribution solutions using different types of network configuration to provide contribution links into production centres.

On Figure 33 , a basic overview of the use case architecture is presented.

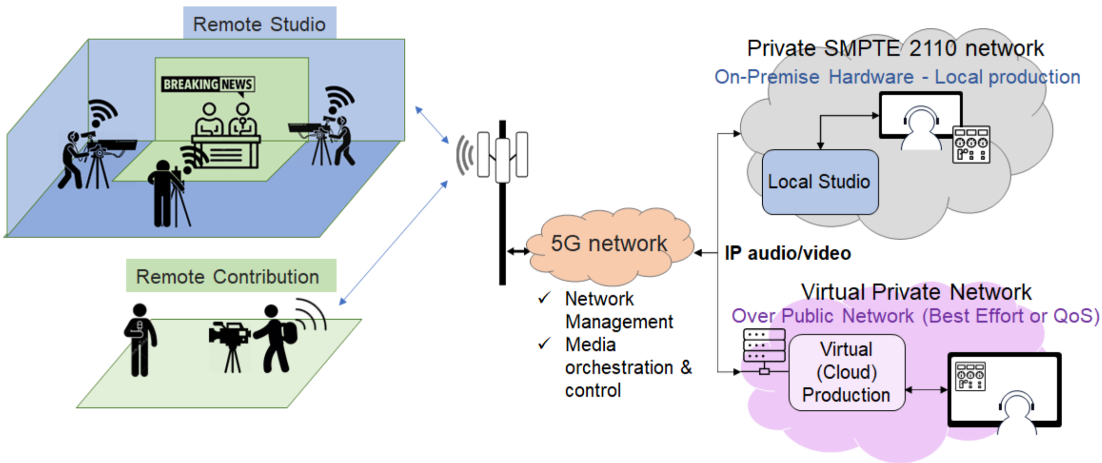


Figure 33: UC2 Basic architecture.

The Video production side of these use cases will explore several scenarios that are complimentary to the whole production process:

- Single or multiple wireless cameras in a studio or outside broadcast event (such as sport, music events, etc.) with control and management.
- Multiple wireless cameras in a studio or outside broadcast event with distributed control and management (cloud)
- Additional contribution feeds supplied over different network scenarios.

The use cases defined for UC2 refer to different production scenarios where single or multiple cameras are connected via a 5G radio access network, where the following functions are common to all scenarios:

1. Discovery, integration and management of 5G-enabled wireless cameras within an IP studio production network.
2. Delivery of streams with minimal latency, produced by 5G-enabled wireless cameras to a remote studio whose transport infrastructure is based on SMPTE ST-2110.

3. Support for seamless integration with wired services (e.g., mixing with local sources)
4. Time synchronisation of devices in the field and in the studio.
5. Feedback (return video, tally, etc..) and controls signals from the studio to the production location.

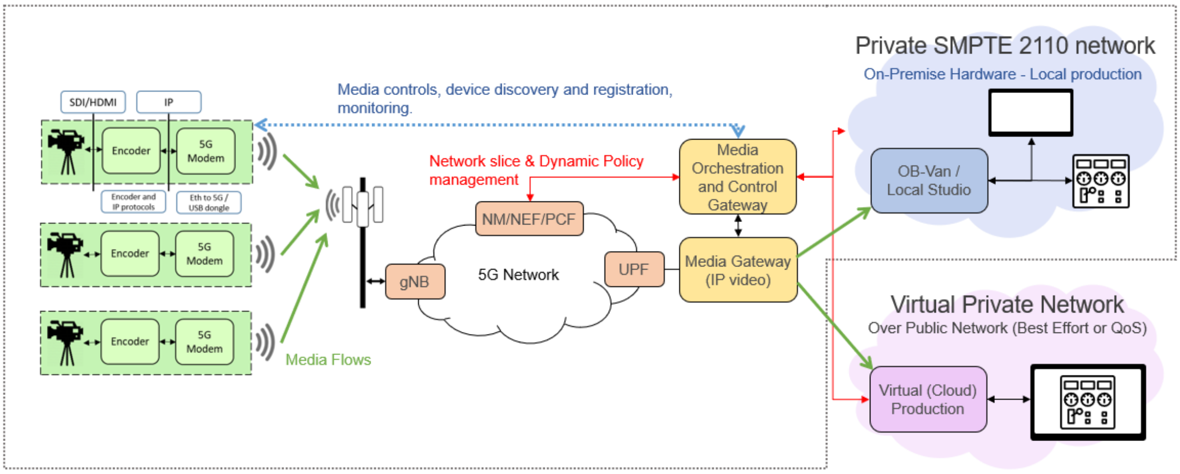


Figure 34: Multiple camera production studio

4.2 Use Cases under analysis

Live media content production usually requires deploying large amounts of equipment and crew on the event location or studio, all connected to the production facilities. The focus of UC2 is to utilise 5G and develop other new technologies to create a wireless, holistic, interconnected content production system that reduces logistical effort and provides all the functions needed for media production. This use case will deploy a studio with wireless 5G-enabled cameras taking into consideration operational requirements for working alongside wired studio equipment and cloud-based workflows. Three main scenarios will be considered: “wireless cameras within a production”, “integration of cloud-based production”, and “remote contribution”.

4.2.1 Wireless cameras within a production

The core aim for this scenario is to explore the substitution of current wireless RF cameras with 5G. Currently, these digital cameras are based on the same technology used in the distribution of terrestrial television (DTT) broadcasting. Remote production will be enabled via wider area connectivity or the cloud. These scenarios require high level of movement and picture detail as well as low camera to vision mixer (switcher) latency. The cameras will be equipped with an external codec and 5G a modem. A media gateway will be developed to translate the protocols from the 5G network to the wired (ST 2110) production islands and the cloud.

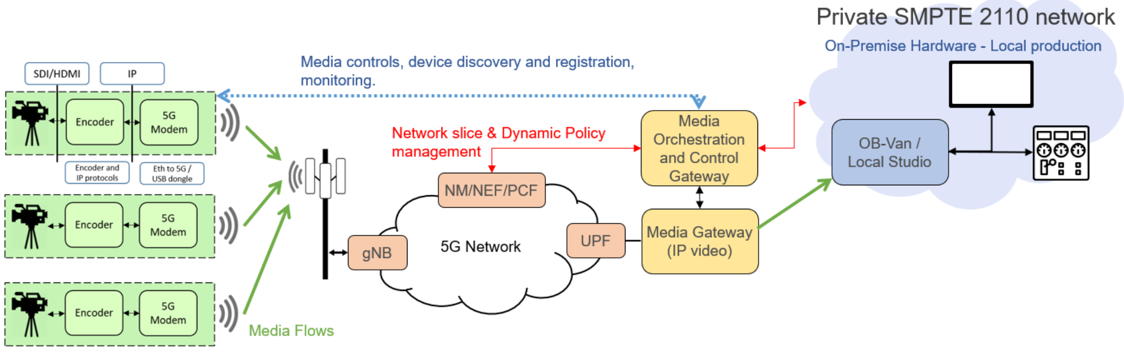


Figure 35: Wireless cameras within a production

4.2.2 Integration of cloud-based distributed production

A (Master Control Room) MCR will be deployed to monitor and manage the incoming and outgoing feeds. The MCR will pull together all these signals and organize them for presentation to operational galleries. The MCR will also provide the connection to cloud-based production tools offering services like audio and video mixing, storage and multi-viewers.

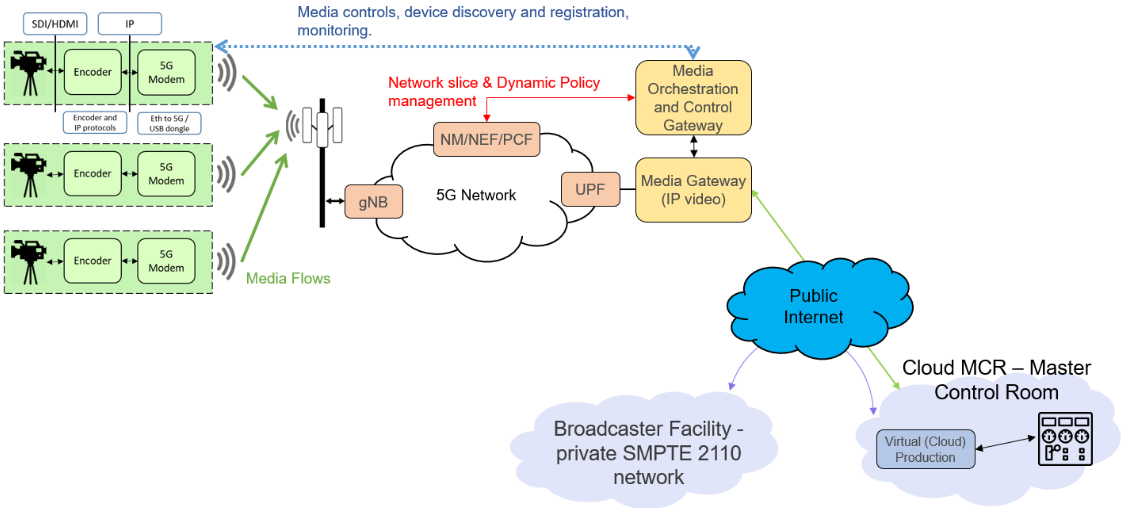


Figure 36: Integration of cloud-distributed production

4.2.3 Remote contribution

Remote contribution consists of receiving the contribution feeds from the remote locations: a backpack or camera-mounted device is used to encode and broadcast video without the need for mobile units (vans) and/or satellite or microwave links. However, the use of 4G networks can bring several disadvantages. For example, due to the bandwidth required, mobile solutions require multiple connections and therefore multiple SIM cards to provide adequate service; this method of connection aggregation is known as bonding. Additionally, when these devices are outside the mobile network provider coverage area, other SIM cards are required to use an alternate network. The video must also be highly compressed due to network bandwidth restrictions. To achieve this, it is common to deploy proprietary solutions that require a paired transmitter and receiver from the same manufacturer. These technologies tend provide a single video link and so if more than one camera is required it needs multiple units that are often timed differently. There is also no differentiation between the networks to which these devices connect and public networks, so in large events 4G connections become unreliable as they struggle for connectivity and bandwidth with other users. It can be expected that 5G solutions will evolve to meet these workflows with little or no intervention.

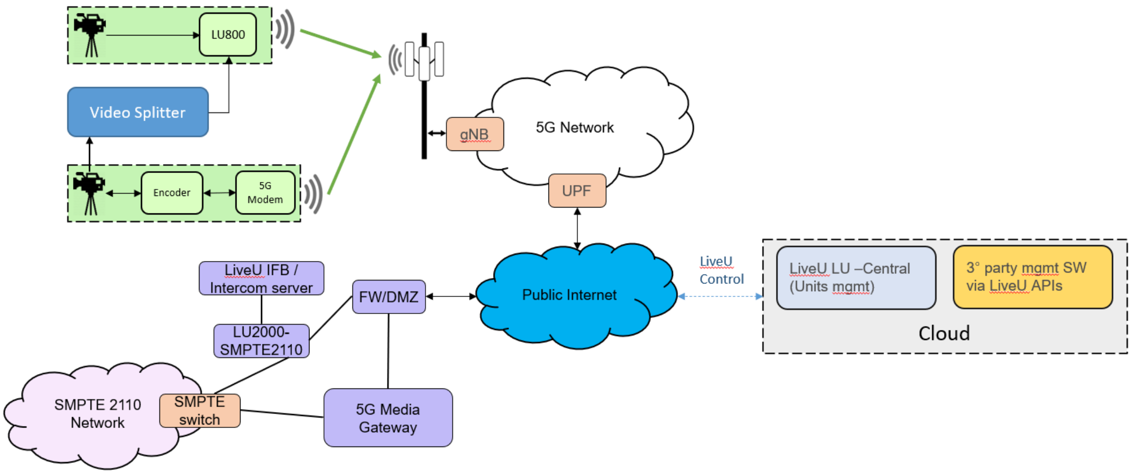


Figure 37: Remote contribution

4.3 Data Flows

This section presents the data flows present in UC2 systems.

4.3.1 Control Flows

For UC2 systems control flows, see Section 6.3 (Media Operational Control Gateway)

4.3.2 5G end-to-end user plane protocol stack

Below, Figure 38 depicts the 5G data flows in the UC2 systems.

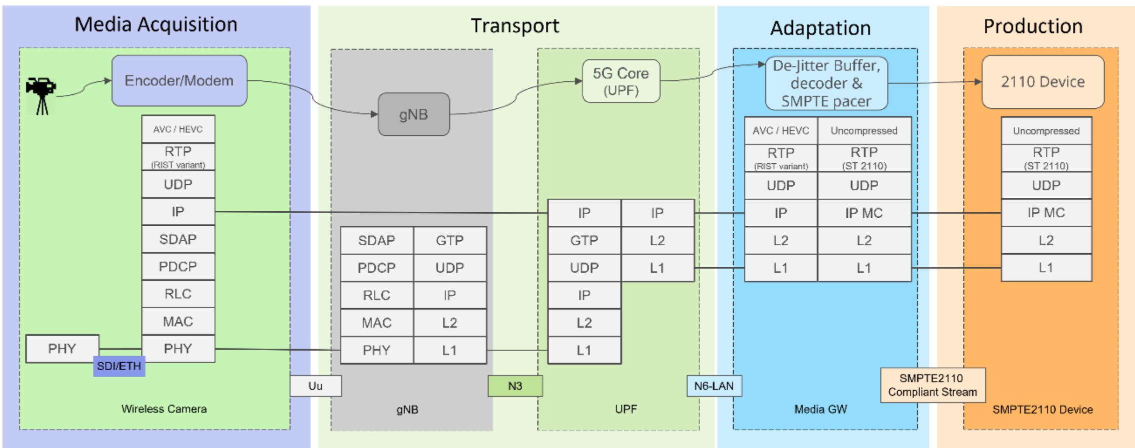


Figure 38. UC2 end-to-end user plane protocol stack.

4.4 Delay budget

The use case defines different latency requirements per user story. The wireless studio scenario requires lower latency than the production or contribution scenario. For more information about latency requirements please refer to D2.1 [2].

The use case latency breakdown considers the latency from the media acquisition to the production studio as glass-to-glass latency. The table below depicts the breakdown of components that contribute to the delay budget. UC2 will investigate the delay budgets mentioned in Table 14.

Table 14: delay budget per category

Category	Property	Description
Encoding and Packaging	Video encoding, RTP/IP packaging	The time required by the encoder to compress the raw images, to package into a media transport protocol and to make them available for transmission at data link layer.
Transport	5G transport	The time in which packets are transferred from the camera to the 5G network edge (N6 interface).
Adaptation	HEVC to SMPTE	The latency induced by the media gateway including de-jitter buffer, decoder.
Production	SMPTE ST-2110 network	The latency induced by transferring the SMPTE ST-2110 stream from the media gateway (i.e., SMPTE ST-2110 Sender) to the SMPTE ST-2110 Receiver.

4.5 List of Components

4.5.1 Component: Fivecomm 5G Modem (F5GM)

The Fivecomm 5G Modem (F5GM) will be integrated into the complete end-to-end system at as part of the user equipment (UE). The objective is to develop, integrate and validate a compact and flexible module solution that provides 5G wireless connectivity that can be customized. It will be optimised for the connection link between the cameras and the 5G network, depending on the specific needs of the use case.

It will connect via Ethernet to the video encoder component developed by Image Matters and connect to the 5G network via its integrated or external antennas. It is a powerful, versatile, and compact device designed to bring all the advantages of the new 5G technology to the media industry.

The F5GM has simplified its electronics to make the most of the 5G modules currently on the market while minimizing the power consumption and cost. It is versatile enough to fit the different requirements specified by content production professionals in the context of this use case. The F5GM is shown in Figure 39.

Fivecomm is working in the development of a prototype to be used in UC2 tests in the summer of 2021. This prototype will provide a single ethernet port with an external IP (Bridge Mode) or multiple ethernet ports with private IPs (not reachable by external IPs unless the connection is initiated from the private IPs) (NAT Mode).

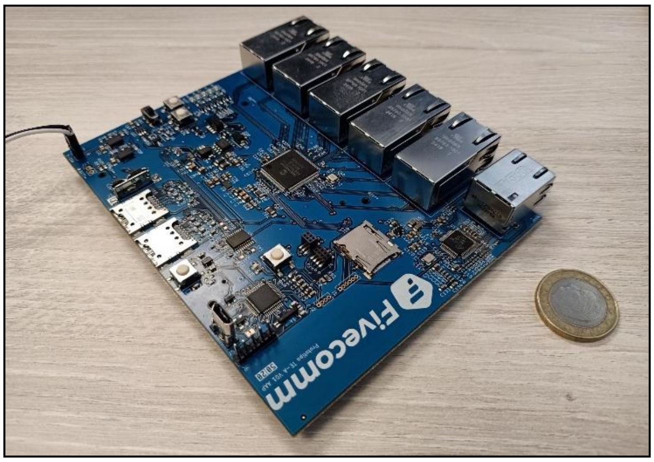


Figure 39: Fivecomm 5G Modem: first prototype.

4.5.1.1 *Functionality*

The 5G modem implements the functionality collected in Table 15 and provides the technical specifications reported in Table 16. More details will be provided in deliverable D3.2.

Table 15: Fivecomm 5G Modem functionality

Functionality	Description
Easy deployment	‘Plug and play’ fashion. It only needs to connect the cables from the video encoder to the digital connectors, fix the device to the infrastructure and press the ‘ON’ button.
Customization	Different IP protection modes are available, with up to 4 internal antennas or external ports to provide the best experience even in low coverage scenarios.
Remote management	It will include a management platform that allows to configure, monitor, and perform software updates remotely.

Table 16: Fivecomm 5G Modem technical specifications

Technical Specifications	Description
5G native mode	Both 5G Non-Standalone (NSA) and 5G Standalone (SA) modes are supported. Option 3x, 3a and 2 network architectures. 3G/4G connectivity is additionally supported.
5G New Radio (NR) Release	Rel-15.
Sub-6 GHz frequency bands	n41/n77/n78/n79/n1/n3/n5/n7/n8/n20/n28/n38/n40
Antennas	Up to 6 antennas (external or integrated) to provide the best experience even in low coverage scenarios.
SIMs	Dual SIM.
Ports	Up to 5 Ethernet (on-demand) and 1 USB connections.
DL&UL	Up to 2.5 Gbps in the DL and 900 Mbps in the UL.
Configuration and monitoring	It will include a management platform that allows to configure, monitor, and perform software updates remotely.

4.5.2 **Component: Image Matters A/V Encoder**

The A/V Encoder from Image Matters will offer the most convenient encoding scheme to serve the 5G-RECORDS project UC2.

The A/V encoder is based on a FPGA chip including a quad-core ARM processor and one 4K MPEG Encoder and Decoder hard core. It presents all the requested Input-outputs (IOs) to transform a basic video camera into a broadcast studio camera. It will connect to the Fivecomm modem to stream to and from the 5G network.

From one 3G-SDI input (maximum capability is 12G-SDI), the A/V encoder supports a high quality HEVC video compression at 1080p50 with ultra-low latency toward the up-link. It uses GStreamer encapsulation to manage video TS streaming over RTP Special attention is brought to the interoperability of the provided stream to enable the Media gateway to decode it on its internal GPU.

From the 5G-modem, the A/V encoder also supports HEVC decoding from the return-video stream coming from the network and outputted on an SDI link toward the operator monitor.

Both video encoding and video decoding operations are done in a full duplex mode.

To synchronize the video signals, the A/V encoder recovers the master clock from the 5G network using PTP protocol. From this, the A/V encoder build an internal video clock and generate a genlock signal to sync the video camera.

Multiple IOs are provided with the A/V encoder to enable uses of additional Studio camera features such as: Tally signals, intercom signals, Additional Audio (analogic / digital) inputs, camera parameter control, lens control, Neutral filter control, …

By its intrinsic flexibility (processor + FPGA) and its high performances (4K HEVC codec, 12G capable IOs, …), the A/V encoder is also expandable to handle 4K video streams and to adapt to other compression schemes (JPEG-XS, VP2, TICO, etc.) and to other transport protocols (SMPTE ST-2110, NDI, etc.).

The A/V encoder is implementing NMOS protocol to be managed by the Operation Control Layer (OCL).

First, as a proof of concept and an enabler for the 5G-RECORDS UC2 demonstrations, the A/V encoder, will be implemented on a Xilinx ZCU-106 evaluation board with specific IO daughter boards. Multiple A/V encoders will be provided to the project for the purpose of use case 2 demonstration set. In a second time and hopefully before the end of the project the A/V encoder will integrate an Origami Bamboo module from Image Matters.

4.5.2.1 *Functionality*

Table 17: IM Encoder technical specifications

Technical Specifications	Description
Video Encoding	Video input: SDI Codec: HEVC Low-Latency or in Xilinx-Low-Latency mode
Audio Encoding	Audi input: SDI – Embedded audio channels Codec: AAC
Stream Out	Streaming: GStreamer, TS over RTP, Essence over RTP
Stream In	Streaming: GStreamer, TS over RTP, Essence over RTP

Video Decoding	Codec: HEVC in Low-Latency or in Xilinx-Low-Latency mode. Other modes to be confirmed
Intercom Signal	To be defined.
Device Control	Control: local or Remote (Through NMOS, to be confirmed)
Clock Sync	PTP clock recovery from 5G master Clock
GenLock	Tri-level sync to camera

4.5.3 Component: LiveU LU800

The LU800 [5] is a portable multi-camera all-in-one production-level field unit. Inside 5g-records project, it is used to encode and send video flows captured from cameras to 5G network. The LU800 enables complex remote productions (REMI), supporting up to four fully frame-synced feeds in high resolution from a single unit, delivering the highest-quality video performance.

The LU800 is a state-of-the-art field production unit. It is a HW-SW device. It captures video feeds from up to 4 sources (cameras, recorders etc), encodes the video and transmits it reliably using a single or multiple links simultaneously and adaptively using a LiveU patented technology.

4.5.3.1 *Functionality*

Table 18: LiveU LU800 functionality

Functionality	Description
5G SA support	Using Sierra modules (and testing others) to support using 5G SA in EU bands.
Channel Sync	Multi-camera with up to 4 fully synced feeds & flexible unit/station switching
IP Pipe	Support bi-directional IP traffic with the studio LU2000-SMPTE server, including low bit rate small-sized packets for remote control of cameras.

Figure 40: LU800 device

4.5.4 Component: LiveU LU2000-SMPTE

LiveU SMPTE2110 [6] is a Bonded Video Decoder that receives the video from the LiveU field encoders. It has a SMPTE2110 A/V out instead of SDI output. The LU2000-SMPTE expedites the transition to an all-IP live video workflow and ensure interoperability with the latest broadcast IP environments. LiveU's LU2000-SMPTE2110 server offers a fully compliant and powerful SMPTE 2110 solution, enabling to simplify the studio infrastructure, maximize agility and reduce costs within an all-IP workflow. Support of: 2110-10,20,30; 2110-40 is not supported. For the PTP external clock the developed support for IEEE 1588-2008 (PTP), SMPTE ST 2059-1/2. Some SMPTE features are still not supported, for example: DSCP markings according to AES67, 2022-7 redundancy- Clock redundancy.

The LU2000-SMPTE2110 bonded video decoder is used to receive, decode and playout any HEVC/H.264 single modem and bonded video streams sent by LiveU's field units or Matrix cloud video management and distribution platform. The 1U rackmount decoder can simultaneously output two full HD streams (up to 1080p) over SMPTE ST-2110 Ethernet ports, sending them to the selected IP destinations.

Figure 41: LU2000 device

4.5.4.1 *Functionality*

Below is a short list of the main features and capabilities developed or integrated within 5G-RECORDS.

Table 19: LiveU LU2000 Functionality

Functionality	Description
Multi-network support	Support of 3 networks/sub-nets: input, output (into studio SMPTE network), master clock input.
IP-pipe support	Support bi-directional IP pipe communication on Ethernet port with the field LU800
Intercom support	Support intercom communications between studio and field LU800
Feed sync	The decoder also includes support for feed synchronization through the Precision Time Protocol (PTP) via the SMPTE protocol (no internal master clock or master clock direct connectivity).

In Figure 42 a typical setup for LU2000 is shown:

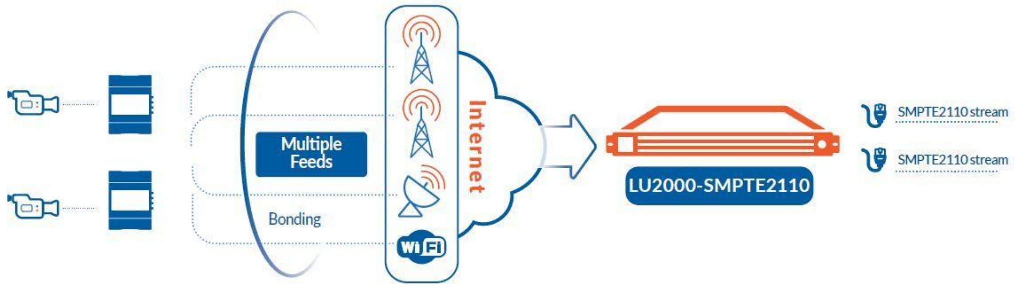


Figure 42: LU2000 chain

Table 20: LiveU LU2000 technical specifications

Technical specs	Description
PTP	supported via external grand master clock
2110 IP outputs	up to two video streams over 2 x 25 GBE SMPTE-2110 Ethernet ports
Additional Video Output	RTMP for streaming to CDBs or social media. MPEG-TS, LiveU Matrix, NDI
SMPTE supported standards	2110-10, 2110-20, 2110-30, 2110-21 Narrow Gap profile, 2022-7 (media redundancy, not clock redundancy). (2110-40 not supported)
Video Decoder	HEVC H.265, H.264
Video Resolutions	1080p60, 1080p59.94, 1080p50, 1080i60, 1080i59.95, 1080i50
Video input Interfaces	2 (redundancy) x 1000/100/`0 RJ-45
Audio channels	2
Video connectors	SFP28
Form Factor	1U rackmount
Power sources	110-240 VAC 50/60 Hz
Video Interfaces	2 (redundancy) x 1000/100/`0 RJ-45
Audio channels	2

4.5.5 Component: Media Gateway

The Media Gateway (MG) is responsible for anchoring the media components within the 5G network and the production studio. It is located at the centre of the whole media production network. The MG converts the 5G protocols and formats into the ST 2110 compliant protocols and formats and vice-versa. It is also able to convert an input stream into a RIST stream. The MG is configured and controlled by the Media Orchestration and Control Gateway (MOCG). In Figure 43, the relation between MG, MOCG and the other orchestration elements is shown:

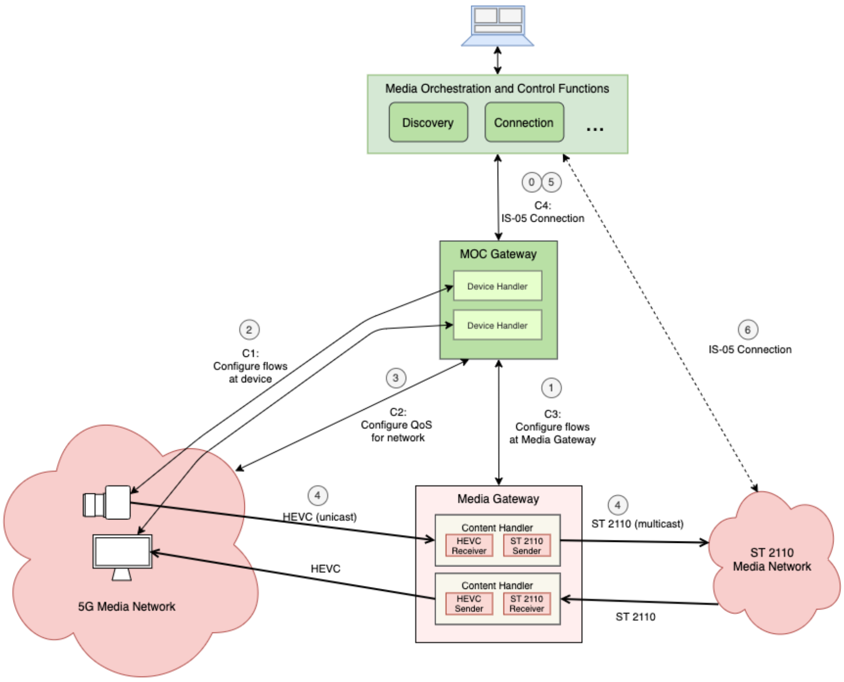


Figure 43: System architecture

The MG exposes two control-related interfaces: one RESTful API and one simple GUI for system monitoring purposes only. The operational control will be performed via the API.

The core component of the MG is the Content Handler (CH). Content handlers are created dynamically, as requested by the MOCG via the API. Each CH receives one input stream and converts it into one or more output streams, possibly in different formats, but with the same content. The supported formats are defined in a separate section below. The following pictures depict several possible configurations of content handlers.

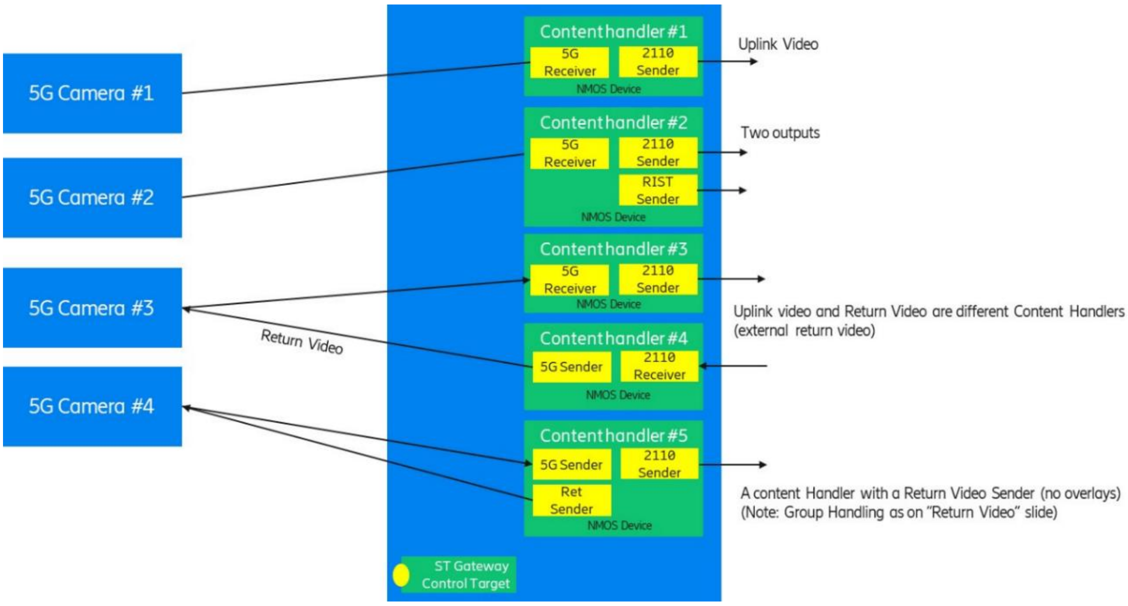


Figure 44: Multiple content handler configurations.

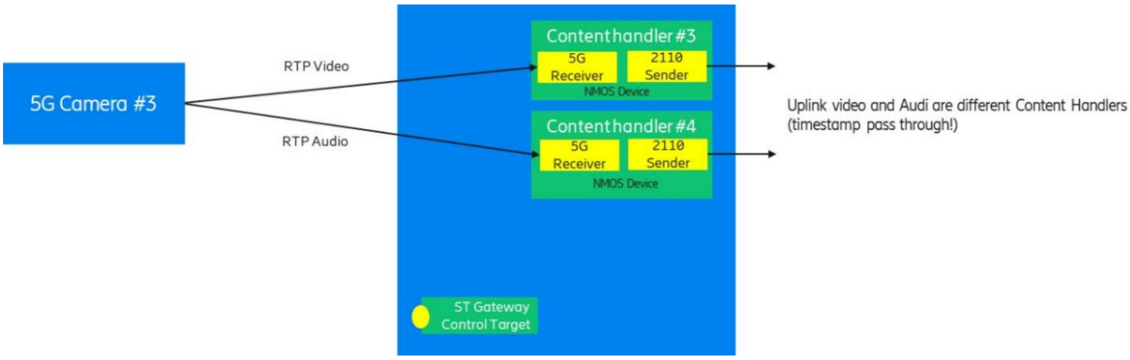


Figure 45: Each content handler processes an elementary stream.

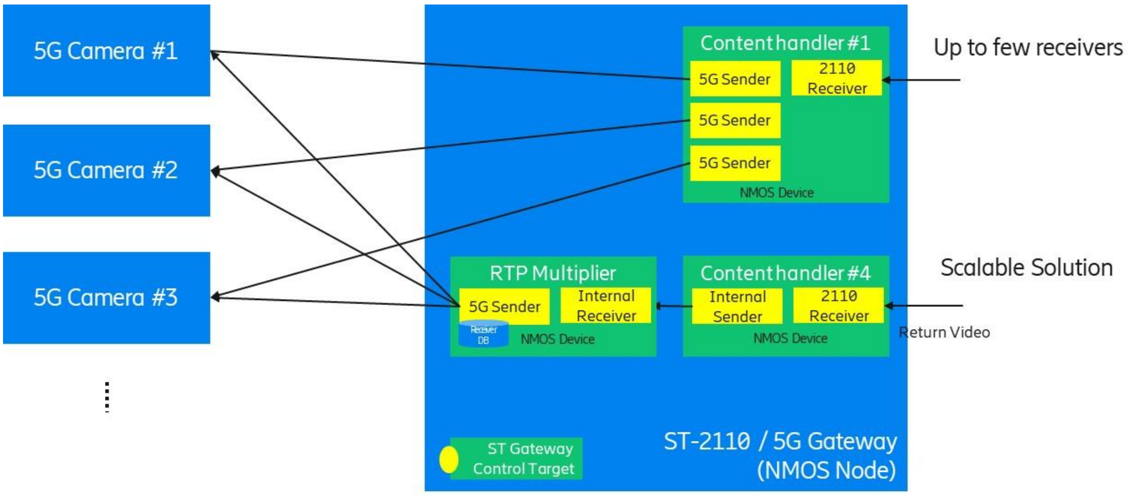


Figure 46: Content handler can output the same stream to multiple unicast destinations.

The MOCG is responsible to send requests to the MG to instantiate and deploy CHs. Each CH is independent of all others. The maximum number of concurrent CHs will depend on the format of the data being handled by the active CHs and will be determined experimentally.

4.5.5.1 *Functionality*

4.5.5.1.1 Application Programming Interface (APIs)

In the following table, the functions of the 2 available APIs are shown:

Table 21. APIs functions for Media Gateway.

Control API functions	Event API functions
Create CHs	Keep alive messages
Destroy CHs	State messages
Get CHs status	State changes
List active CHs	Alarms

4.5.5.1.2 Physical specification

The Gateway is composed of two major components:

Table 22. Media Gateway physical components.

Major Components	Description
NVIDIA Jetson AGX Xavier Development Kit	NVIDIA Jetson Xavier is an AI computer with the performance of a GPU workstation in under 30W of power consumption. It has 8-core ARM v8.2 64-bit CPU, 512-core Volta GPU with Tensor Cores, 32GB of 256-bit wide LPDDR4X memory and 2x 4Kp60 | HEVC/2x 4Kp60 encoder/decoder. A combination of high-performance, low-power computing makes it a good platform for compute-intensive projects, including the Gateway.
NVIDIA Mellanox ConnextX-6 network adapter	ConnectX-6 supports two ports of 200Gb/s Ethernet connectivity, sub-800 nanosecond latency, and 215 million messages per second, plus block-level encryption and NVMe over Fabric offloads. For the Gateway, this NIC will provide packet pacing for ST 2110 video streams (as required by ST 2110-21). In future it might be replaced by NVIDIA BlueField-2X DPU (Data Processing Unit), which will host the media pipelines and may be also potentially used to offload AVC/HEVC encoding and decoding.

Figure 47: Jetson AGX Xavier Development Kit and ConnextX-6 NIC

The gateway contains 3 Ethernet interfaces: (A) 1x Gb/s RJ-45 and (B) 2x 200 Gb/s QSFP. Interface A is meant to be used for the API and to transport the streams to and from both the 5G network and the public Internet. Interface B is meant to be connected to the ST 2110 studio, transporting both the essence streams and the PTP timing data.

4.5.5.1.3 Software platform

The operating system running on the Jetson is based on NVIDIA L4T 32.5, which includes a bootloader, Linux Kernel 4.9, the appropriate drivers, and a file system based on Ubuntu 18.04.

The Gateway software makes use of several libraries, including NVIDIA CUDA [CUDA] and Rivermax [Rivermax].

4.5.5.1.4 Architecture

The block diagram of the Gateway is presented in the following figure:

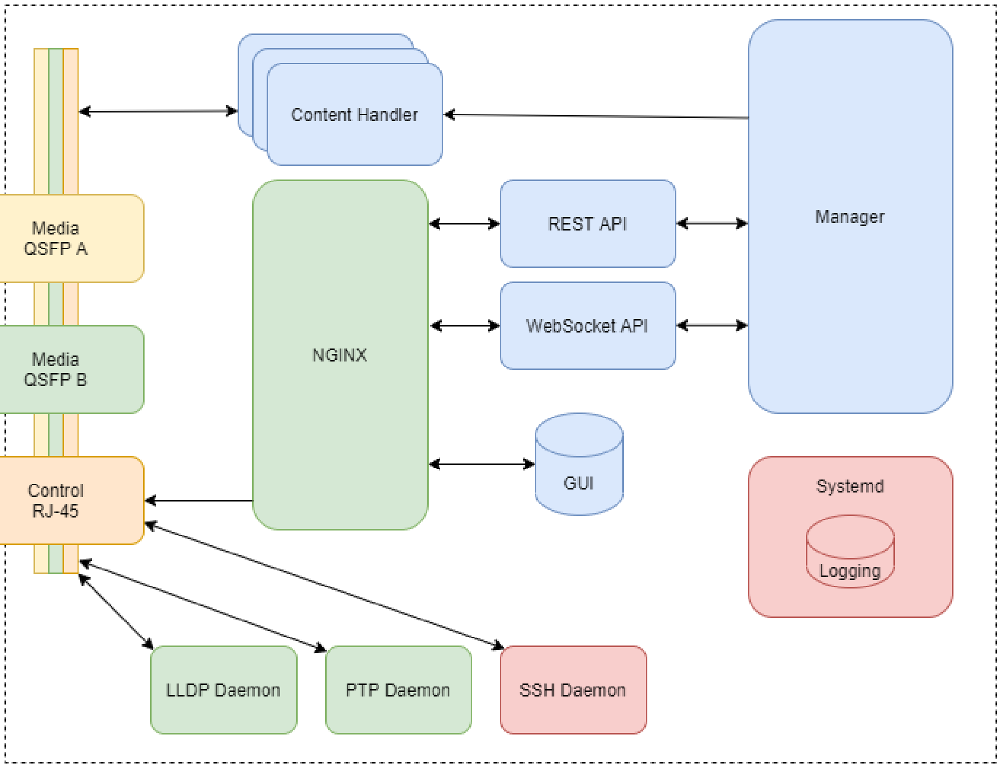


Figure 48: Gateway block diagram

Each of the blocks is an independent software unit, communicating with each other using internal APIs.

Table 23. Media Gateway software units.

Software units	Description
Systemd	Systemd is a system and service manager for Linux. On the Gateway, it is responsible for launching and monitoring the different services that comprise the Gateway software, as well as for storing the logs of all services, which are internally accessed using journals
Manager	The Manager is responsible for coordinating the operation of the Gateway. It initializes the internal resources; it receives requests from the API and executes them; and it reports the current status through the API
REST API	The REST API is exposed by a service that implements an HTTP server that receives and responds to API requests. The requests are forwarded to the Manager and its responses are formatted and relayed back to the client by the REST API server
WebSocket API	The WebSocket API is handled by a service that manages the connections requested by API clients. It is notified by the Manager of relevant events (including the “Keep alive” timer) and forwards them to registered clients.
LLDP Daemon	The Link Layer Discovery Protocol (LLDP) daemon responds to LLDP requests publishing information such as the host name of the Gateway and the IP address of the control interface, in order to allow other devices to be easily discover the gateway on the network.

PTP Daemon	The Precision Time Protocol (PTP) daemon handles the communication with the PTP master, in cooperation with the OS and the network stack, providing accurate time information to the whole system, including the real time clock.
SSH Daemon	The SSH daemon allows a user to connect to the Gateway shell using the SSH protocol. This can be used for all sorts of operations including updating the software and troubleshooting
NGINX	It performs several tasks on the system: • it serves the GUI using HTTPS; • it performs HTTPS termination for the REST and WebSocket APIs; • it acts as a reverse proxy and maps all services to the same external base URL.
GUI	the GUI is a Single Page Web Application, served statically by NGINX. It runs on the client's Web browser and interacts with the gateway exclusively using HTTPS through the REST and WebSocket interfaces, via the NGINX proxy.
Content handler	a content handler is spawned by the Manager at a client's request and remains active until the client requests its destruction. It receives an essence stream from the network and produces one or more output streams, performing the required depacketization/packetization and decoding / encoding

4.5.5.1.5 Interactions

MOCG and Media Gateway

The MOCG uses the MG's REST API to request the Gateway's unique ID and uses it to register one NMOS Node on the RDS.

As needed, the MOCG requests the MG to instantiate a CH, providing the required settings, which include the input stream's SDP and the configuration of the outputs. The MG returns, among other information, the ID of the Content Handler, the ID of the input and the ID of each of the outputs. The MOCG uses these IDs to register on RDS, respectively, one Device, one Receiver and N Senders, one for each output. The Gateway also returns the SDPs for each of the outputs.

When that Content Handler is no longer needed, the MOCG:

- Deletes the corresponding entries from the RDS.
- Uses the MG API to request the destruction of the CH. The MG then releases all the resources that were reserved for that CH.

User browser and Media Gateway

The user points the browser to the MG's HTTPS URL and logs in using the default credentials. The first time, and whenever needed, the user: changes the password using the settings page and, changes the system settings appropriately.

Alternatively, the user selects the dashboard page and is presented with the current information about the system.

4.5.5.1.6 Configurations

Table 24. Needed configurations for the Media Gateway.

Element to be configured	Configuration description
Network	All the network parameters are configured using DHCP. It is expected that the Gateway will operate on a controlled environment, where a DHCP server will be present and configured with the relevant parameters including, if necessary, static mapping between the MAC addresses of the interfaces of the Gateway and the desired IP addresses.
Discovery	Since the interaction with the NMOS ecosystem will be performed solely by the OCS, the Gateway will not discover and register itself on a NMOS Registration & Discovery System (RDS). However, the Gateway will expose information via LLDP, allowing neighbour devices to retrieve relevant information such as the name, unique id and the IP address of the control interface.
Settings	The settings can be changed both using the GUI or the API. See the API specification for details.

4.5.5.1.7 Required settings

The settings that are required for the system to be fully usable are:

- PTP: Domain number and announce timeout.
- Other settings: user and password.

4.5.5.1.8 GUI - Server

The GUI is served via HTTPS on ‘/’. If the MG is running on 192.168.1.1, the URL to access the GUI is: https://192.168.1.1

4.5.5.1.9 Monitoring

- Display information about the active CH, including:
  - Input and output formats
  - Whether the input signal is active or not
- Display dynamic information about the system:
  - Input and output bitrates
  - Processor, GPU and memory utilization
- Display static information about the system:
  - Software version

4.5.5.1.10 Security

Table 25. Security elements for Media Gateway.

Security element	Configuration description
Certificates	The Gateway uses a self-signed certificate. Therefore, API clients will need to tolerate this and users accessing the GUI will usually receive warnings from their Web browsers. We may consider the

	possibility of allowing the user to install trusted HTTPS certificates on the Gateway
Firewall	The Gateway will use Ubuntu's default firewall [ufw]. The only ports that will be open normally are TCP ports 22 and 443, for SSH access and API/GUI, respectively. Other ports will be open as needed to receive UDP traffic.
User management	There is one user for the API, 'admin'. The default password is 'admin' and there is an API call to change that password but not the username. The GUI also provides a means for the user to change the password. There is one user for SSH, 'admin', with default password 'admin'. The password can be changed using the regular Linux procedures.

4.5.5.1.11 Performance

The maximum number of concurrent streams will depend on the formats and will be determined experimentally during the next phases of the project.

4.5.5.1.12 Supported formats

Table 26. Media Gateway supported formats.

Supported Formats	Configuration description
Video Resolution	• HD: 1920x1080p50 • UHD: 3840x2160p50
Audio Resolution	• 48 kHz, 24 bits per sample, 1-64 channels per stream

4.5.5.1.13 Protocols and codecs

Table 27. Media Gateway supported protocols and codecs.

Network	Supported protocol and codes
5G network	• Video: H.264/AVC (per IETF RFC 6184) and H.265/HEVC (RFC 7798) • Audio: MPEG-1 Layer2 (RFC 2250) and AAC (RFC 6416)
2110 Network	• Video: SMPTE ST 2110-20 • Audio: SMPTE ST 2110-30
Public Internet	The same codecs and payload formats as 5G Network, over RIST Simple Profile (VSF TR-06-1) [RIST-simple]

4.5.6 Component: 5G networks

5G-NR (New Radio) is the latest standard for telecommunication networks. It was first introduced in 3GPP Release-15 in December 2018. 5G standardization has re-design the whole 5G network functionalities and components from the ground-up. While LTE was considered a great evolution for already existing internet services, targeting mainly end consumers by increasing reachability and achieve higher bandwidth, 5G design had to be compatible not only with the current services but also for future services with diverse requirements.

A detailed description of 5G core network features within 5G-RECORDS (i.e. service-based architecture, standalone and non-standalone architectures, QoS, network slicing, non-public networks, Mobile Edge Compute, and network exposure) is presented in Annex A: 5G Network

4.5.6.1 *Functionality*

5G-RECORDS UC2 targets the following 5G network functionality:

Quality of Service

Quality of Service (QoS) allows the core and radio network to tailor the network resources to provide a more deterministic quality for a specific stream for the UE. Specific IP packets can be marked with a QoS flow identifier (QFI) to allow QoS handling. Each QoS is then mapped to a specific data radio bearer at the Access Network. In the core-network, different network based QoS solutions can be used to provide the required performance. For example, the network can use DiffServ and derive DSCP based on the QFIs.

In 5G-RECORDS UC2, the camera generates and receives multiple streams both on the uplink and the downlink, where each stream has its own characteristics and requirements. To ensure optimal usage of the network resources, the streams are mapped to the matching 5QI. The standard 5QI mapping to network characteristics is depicted in Figure 98 in Annex A: 5G Network.

Mobile Edge Computing (MEC)

Mobile Edge Computing brings the cloud computational capabilities closer to the end-user. 5G has adopted edge computing to enable the integration of the edge within the network. Thanks to the virtualization of the network functions, the MEC can be deployed together with the user-plane functions closer to the enterprise.

For example, in the remote media production use-case, the production chain can be distributed among multiple sites. A portion of this chain is where the director team is moderating the event, in the OB van, which requires high bandwidth, very low latency and high reliability for both the UL/DL. An edge server and user plane functions can be deployed very close to the cameras as shown in Figure 49 (examples 3 and 4) to fulfil the use case requirements. Therefore, in UC2 this capability is exploited connecting the 5G Media Gateway directly to the UPF, in order to reduce latency, which is a critical factor.

After the initial production process, the stream can be sent via SMPTE ST-2110 to the studio for further processing. Here, the studio can be few blocks away from the camera's location and another edge compute can be deployed to handle the processing as shown in Figure 49. Such flexibility enables media broadcasters to distribute both the network functions for all their 5G networks and media processing functions across multiple locations, which reduce cost and provide an overall flexible architecture.

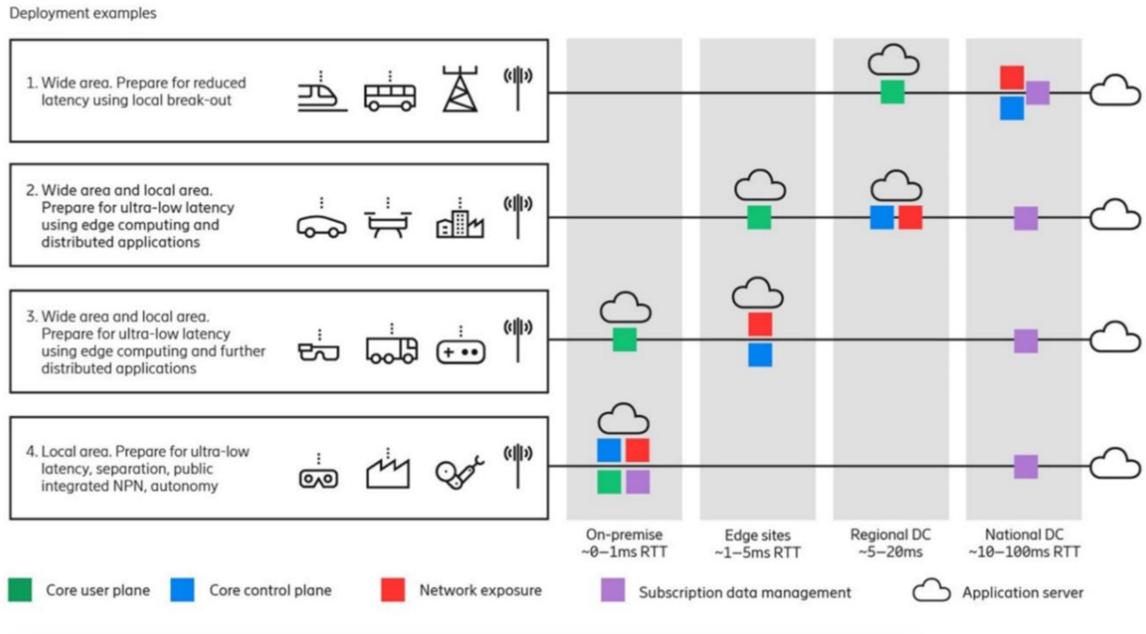


Figure 49: MEC and core deployment scenarios.

Network exposure

The 5G network introduces the Network Exposure Function (NEF) to enable interaction between external applications and the network functions. The services that are exposed to the external Application Function (AF) can be categorized into four services:

Table 28. AF service categories.

Service category	Description
Monitoring capability	It monitors a specific event for UE in 5GS and expose the available monitoring information for external exposure via the NEF
Provisioning capability	Enables external party to provision of information which can be used for the UE in 5GS
Policy/Charging capability	It handles QoS and charging policy for the UE based on the request from external party
Analytics reporting capability	It allows external party to acquire analytics information generated by 5G System

5G-RECORDS will focus on the policy/charging capability of the NEF. It enables an AF hosted in the media orchestration and control gateway (MOCG) to request specific QoS values for a given stream. The NEF APIs are exposed via RESTFUL HTTP. It supports either setting up a new QoS or updating the current QoS assignment.

An example of a call-flow for establishing an AF session with specific QoS parameters is depicted in Figure 101 in Annex A: 5G Network.

4.5.7 Component: Master Control Room (MCR) in the cloud

The public cloud provides a vast resource of compute, storage and connectivity, all waiting to be called upon. As live events are by definition time-bound (and usually of significant importance), the public cloud becomes a very interesting place to host the necessary connectivity and MCR services. Until late 2020, this was not feasible (at least not for high profile events) because the public cloud was not capable of providing the necessary real-time computing resources, the required resilience and the high bandwidth, high quality, low latency connectivity between services/components and endpoints. All that is now changing with emerging cloud technologies.

AWS CDI is the first public cloud media API that promises to supply real-time video connectivity and processing from within the public cloud. CDI is based on the standardized SMPTE 2110-20/30 defined payload, but uses unicast transport mechanisms instead of multicast, thus enabling the services to be run within the usual concept of security, availability zones, regions and virtual routing.

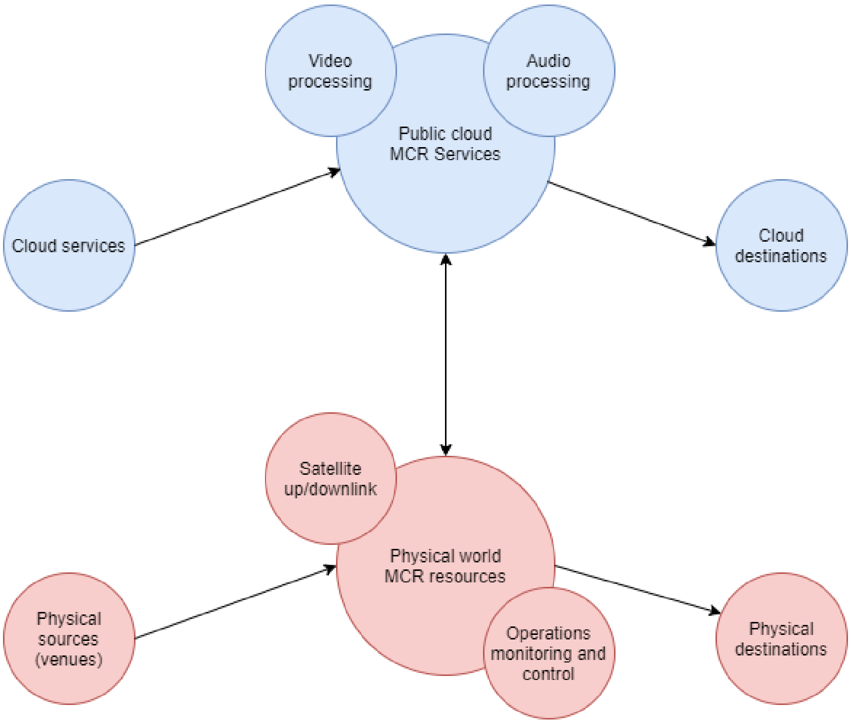


Figure 50: Cloud MCR Functionality and interaction with physical MCRs

In addition, more and more venues, companies and distribution platforms connect to or even originate in the cloud. The delivery of uncompressed streams from live events into the cloud still requires connectivity, compression, monitoring and broadcast operations to ensure the streams arrive in the right format at the right time. A traditional MCR offers these services with expertise and at scale. It can be integrated with the public cloud to offer a complete service that assures the delivery and hand-off of live video into the cloud for downstream production, playout and distribution. As such, MCR in the cloud is a true virtual representation of the physical MCR.

This technology is brand new (launched in Q4 2020 by AWS) and general industry support is still scarce and under development. Part of the 5G Records work in this area therefore is:

1. To identify the relevant connectivity and processing requirements.
2. To place this function in the 5G Records workflow.

3. To understand how to make this work reliably, in terms of PoC and production readiness.
4. To understand what the cost drivers are should this be used in a commercial situation.

Figure 51 shows the functional flow and video processing functions:

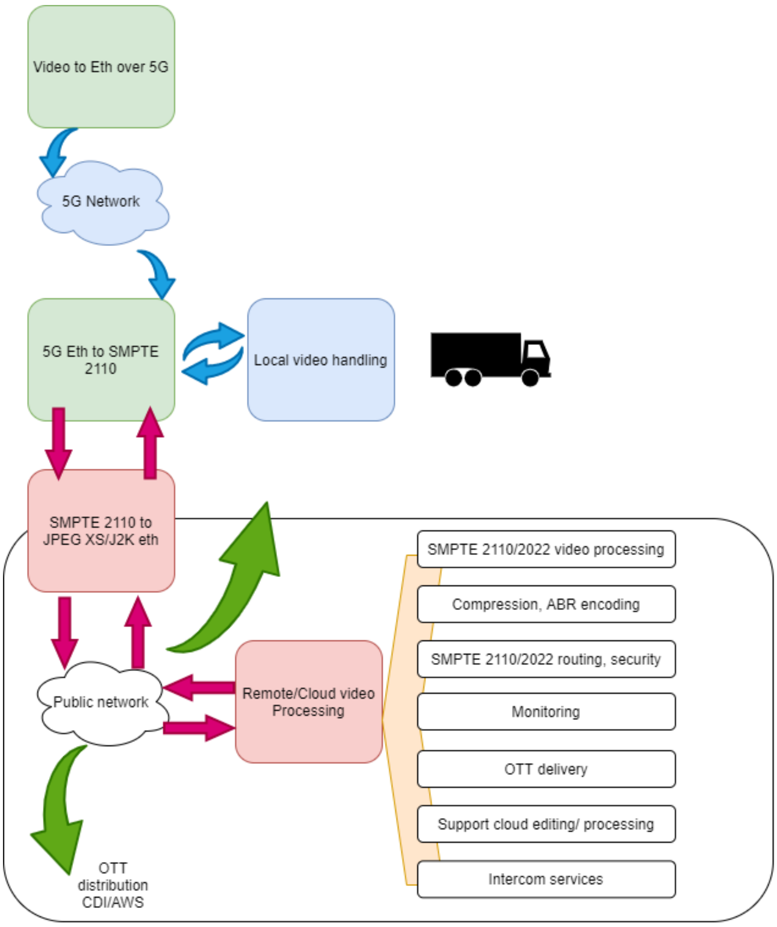


Figure 51: Functional flow and cloud video processing functions

The Cloud MCR is a natural extension or exponent of the traditional MCR. Standard functions such as booking, routing and monitoring (quality assurance) are as much essential in the cloud as they are on the ground. The main additional ingredients that are identified as cloud-specific are:

- Define a high quality, low latency, reliable connection method for interfacing with the cloud. This includes the streaming format, protocol and handshake. Here, ‘high quality’ means visually indistinguishable from baseband video and ‘low latency’ means milliseconds, not seconds. AWS CDI uses AWS MediaConnect with JPEG-XS (SMPTE 2110-22) over unicast RTP streams, for instance, meeting both requirements.
- Identify the ‘usual suspect' vendors to provide solutions for key functionalities such as routing control, confidence monitoring (visual), signal monitoring (alarms) and signal processing.

- Create the interface to the physical MCR (operations) and other components of the 5G Records (remote production and other cloud-native applications, and streaming platforms for end-user delivery).
- Secure the signals within the perimeter of the service responsibility.

The diagram below shows a possible workflow-view of the combined physical MCR (on the left) and the Cloud MCR (AWS assumed). The central region provides an overview of possible use cases that can be addressed with the setup.

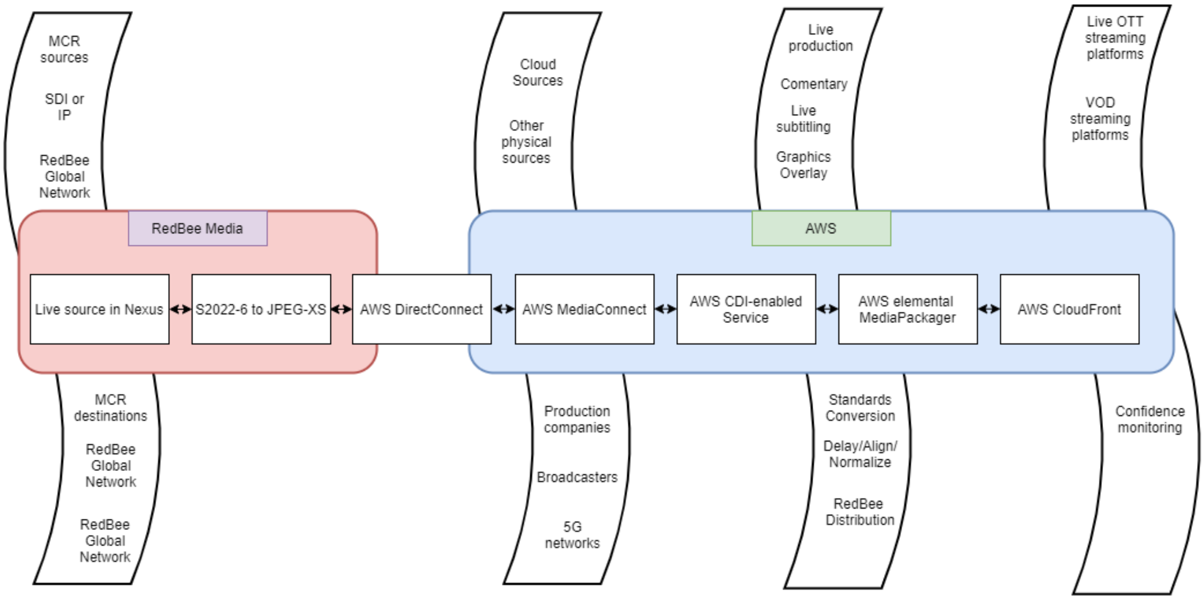


Figure 52 - Possible Workflow-view of the combined physical MCR and the Cloud MCR

4.6 Interfaces

Interface descriptions are reported in *Annex A: 5G Network*.

5 UC3 Components - Live Immersive Content Production

5.1 General architecture

The Use Case 3, for Live Immersive Media Production, comprises the deployment of the end-to-end video chain for a content production pipeline based on Free Viewpoint Video (FVV). This implies the capture of the scene with different cameras, the generation of a virtual view controlled via camera operation, the processing of the resulting live stream for its distribution, and the actual delivery to end users located in different geographical points. Additionally, the use case addresses the deployment of such chain, as well as the setup of its different elements, in a temporary way: which implies that all resources may be commissioned and decommissioned for the use case implementation. A full description of the rationale, scope, and requirements of the use case can be found in Deliverable D2.1 [2].

In the first iteration of the project, the building of the different subsystems of the end-to-end 5G platform will be focused on validating the functionality and interoperability of the elements. With this in mind, the following scenarios, as defined in D2.1 [2], will be addressed:

- **Simple content production** scenario. A small real scene is to be captured with a few cameras that will be placed at relatively short range, with few people being featured in the scene to minimize the number of necessary cameras and associated network requirements. The streams are sent to a single view renderer, which enables the system to save network resources by only sending the reference streams from the physical cameras closest to the selected virtual viewpoint.
- **Delivery scenarios 1 and 2**. Two end-to-end slices are defined for the delivery, best effort and multimedia gold, where the latter is used for production traffic and premium end users.
- **5G Theater** deployment scenario, which models the deployment where an existing[2] 5G network is configured to support the production of a specific event. Network slicing is used to guarantee uplink and MEC capacity over existing NSA.

Figure 53 describes the high-level architecture of the Use Case. The architecture is generic enough to cover the whole project. We will describe the building blocks and, in particular, how they will be implemented in this first project iteration. It is split in several locations:

- **Smart venue**. A small event room, emulating a theatre, with a deployment of 5G millimeter wave (mmW) RAN. It will include: *i)* a FVV system, *ii)* the local camera operator, who operates the virtual camera offered by the FVV platform, and *iii)* a small number of end users, which can access the live production of the event. It will be located in Segovia (Spain), using the Telefónica pilot 5G network with Nokia infrastructure and operation.
- **Near Edge***.* It hosts the VNFs required to process the FVV streams and generate the virtual views which are produced. It is also located in Segovia, to minimize latency between the venue and the processing.

[2] Even though from the use case perspective the infrastructure is modeled as "existing", from the analysis perspective or how this use case would work in a commercial deployment, in practice it will be a pilot 5G infrastructure, and part of it will be actually commissioned and configured for the project.

- **Cloud Edge** and beyond. It hosts the VNFs required to deliver the produced streams to end users and other third parties. It is located in Madrid and connected with the Near Edge.
- **Remote locations:** where users different from those located in the Smart venue will access the live production of the event either as end-consumers of the content or as remote producers that will include the media flow into an Edited Content Creation. Several locations in Spain will be used for testing.

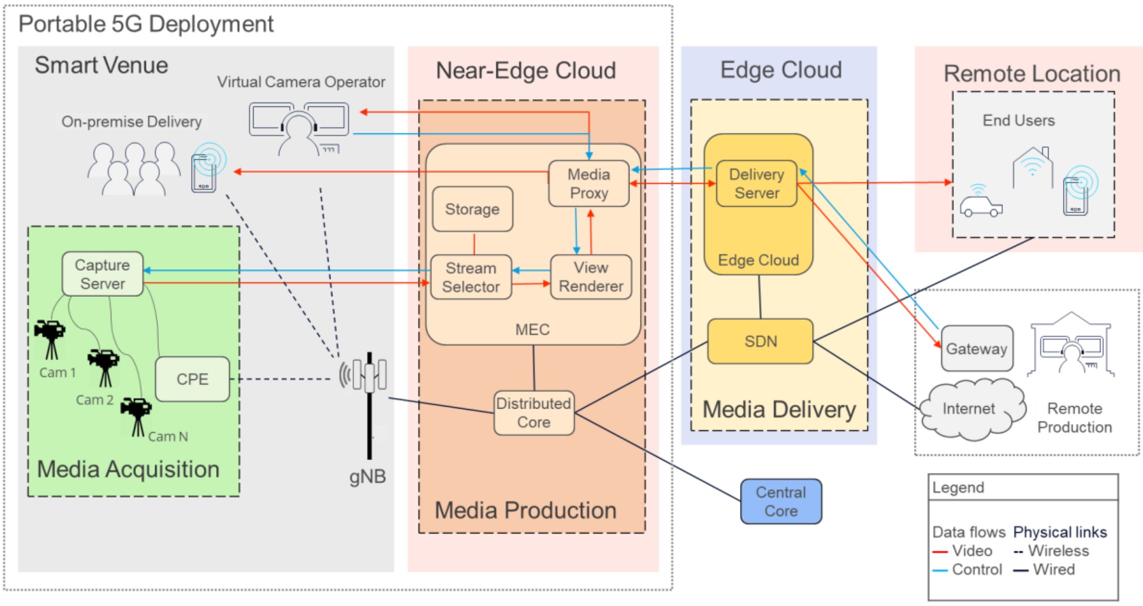


Figure 53: Use case 3 high-level architecture

The use case deployment considers three main subsystems, which implements the required Network Functions to build the end-to-end chain:

- **Media acquisition:** A set of capture servers, each one receiving the stream from up to 3 stereoscopic cameras. The capture server generates the stream for each camera (RGB + Depth) and sends it, via the 5G network, to the media production VNFs. Capture server function is implemented in *baremetal* due to the tight coupling of the cameras and the processing.
- **Media production:** The View Renderer function implements an FVV renderer and encoder for live or on-demand requests: it creates a virtual view (similar to a virtual camera) whose trajectory is remotely controlled by a camera operator. Two additional VNFs are required to support the instantiation of multiple View Renderers: a Stream Selector, which receives the flows from all the cameras and distributes them properly to the View Renderers, and a Storage system for on-demand production. Those two VNFs will not be available in this first iteration and will be addressed for the final system.
- **Media delivery:** The Delivery Server VNF receives the output of all the View Renderers and delivers it to third parties: content producers or broadcasters, in contribution quality, and end users, in streaming quality.

These subsystems work on top of the following 5G infrastructure elements:

- **5G mmW CPE**. It provides mmW 5G access to the Capture Server. Due to the high uplink requirements of FVV, using mmW is needed to provide the service with reasonable QoE.

- **5G mmW gNodeB**. It provides a mmW Radio Access Network.
- **Distributed core**. The 5G core is distributed. It includes an edge core subsystem implementing UPF/N6 interface with the near-edge cloud for the video traffic, as well as a central core for the rest of the functions.
- **Near-edge cloud (MEC)**. The Media Production VNFs run on a MEC infrastructure, which is physically close to the 5G RAN (in the same city of Segovia).
- **Edge cloud.** The Media Delivery VNFs run on Edge Cloud infrastructure in a different location (Madrid, about 100 km away from Segovia).
- **Software Defined Network**. An SDN will take responsibility of end-to-end QoS management between the edge core and the end users.

5.2 Data flows

5.2.1 5G end-to-end user plane protocol stack

The following figure shows the protocol stack for the video data flows.

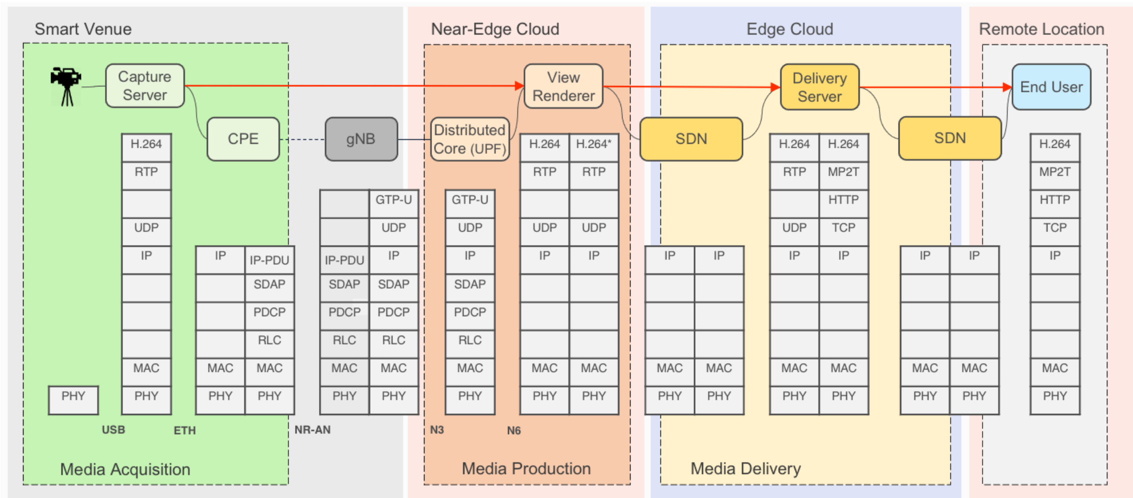


Figure 54: UC3 user plane Protocol Stack

5.2.2 Content production

Currently, content production set is based on nine stereo cameras capturing RGB + depth information. These cameras are managed by three capture servers on a three cameras per server configuration, so each server delivers three RGB video streams plus three depth streams to the edge server for view synthesis rendering.

The system is composed by five different modules:

1. Off-line training module (described in Annex C)
2. Capture module
3. Encoding and transmission module
4. Reception and decoding module
5. View synthesis module

The data flows among these modules can be classified into synchronization, stream flows and adaptive camera switching control.

Synchronization

Before the actual RGB + depth capture process starts, the capture module synchronizes the cameras and the capture servers thanks to a software-based strategy relying on the Precision Time Protocol (PTP) standardized by IEEE. This synchronization step is critical because non-synchronized multi-camera streams can create visual artifacts in the view synthesis at the receiver. The synchronization strategy is based on frame timestamps and on a hierarchical approach that involve first, a master capture server that works as reference for the rest of the capture servers. Then, synchronization among the cameras that belong to the same capture server is carried out. Figure 55 shows the synchronization process scheme for both, server-to-server synchronization and camera to camera synchronization for a three capture servers configuration, each one managing three cameras.

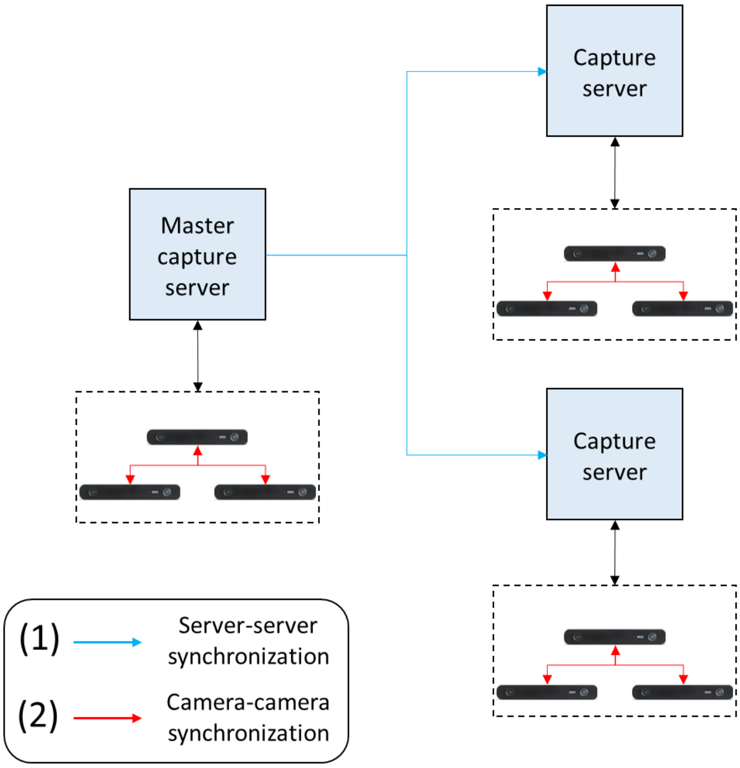


Figure 55: Synchronization process scheme for three capture servers each one managing three cameras

Streams flow

Once all cameras and servers are synchronized, the capture module performs frame acquisition, depth computation and post-processing, and then feeds the encoder with RGB frames and depth maps. The encoding and transmission module encodes both kinds of data and transmits them over RTP to the edge server through the 5G network. On the edge server side, RTP streams are received and decoded by the reception and decoding module, stored in memory and, in parallel, delivered to the real-time view synthesis renderer to compute the virtual view chosen by the virtual camera operator. Figure 56 shows a general scheme of the data flow from the capture to the view synthesis.

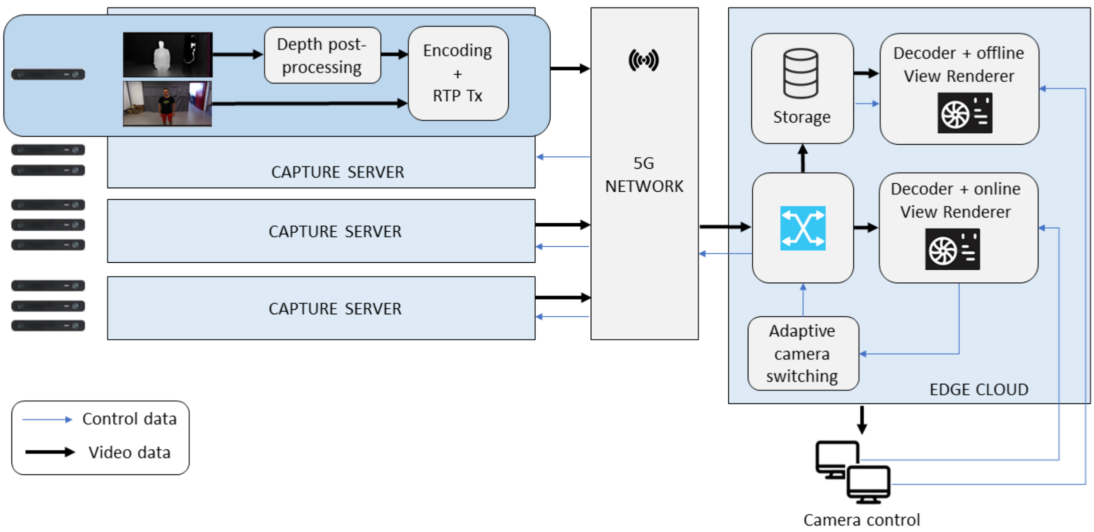


Figure 56: General architecture scheme of content production system

Figure 57 shows both RGB and depth streams data flow. Once a frame is captured, the RGB data is encoded using an H264 lossy scheme, packetized and transmitted over RTP. On the other hand, the corresponding depth map is computed and post-processed prior to its encoding and transmission. The post-processing stage consists of depth correction, 12-bit quantization of the computed depth map, and bit re-arrangement allowing to fit a 12-bit depth map frame on an 8-bit YUV 4:2:0 frame (8 bits for Y, and 4 bits for U and V). The post-processed depth map is encoded using an H264 lossless scheme, packetized and transmitted over RTP. Once all data is received at the edge server, RGB and depth streams are decoded. Then, inverse bit re-arrangement and 12-bit dequantization are applied to the decoded depth maps. The decoded RGB frame and depth map are fed to the view synthesis renderer, and the virtual view is computed. Additionally, the received streams can be stored on the edge server memory so offline view synthesis can also be computed.

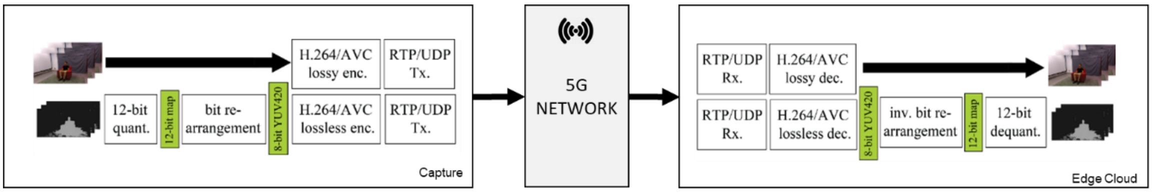


Figure 57: RGB frame and depth map data flow

Adaptive camera switching control

To optimize the available bandwidth usage, an adaptive camera switching algorithm has been developed that avoid the constant transmission of the streams from all the cameras. This approach is suitable for scenarios where only one view synthesis is demanded. In this case, there is only one online view renderer working at the same time on the edge cloud. Thus, it is possible to significantly reduce the transmitted bitrate, as only the streams from a subset of cameras are required at a given moment. This subset is chosen dynamically depending on the viewpoint chosen by the virtual camera operator/producer. A control message is transmitted from the edge cloud server to the capture servers signalling the required cameras (cameras closest to the virtual viewpoint) to be transmitted at each time.

5.2.3 Content delivery

In the Media Delivery subsystem is critical to maintain several network requirements that ensure QoE both for the traffic ingestion and for the traffic delivery. As a strategy to reach those network requirements we will explore e2e Network Slicing in the context of Content Delivery for Live Immersive Media Productions.

The delivery server will be located the Edge Cloud location (Telefónica Madrid-Peñuelas Central Office). It is a virtual machine running in server number 2 of the edge. Detailed components will be described into next chapter.

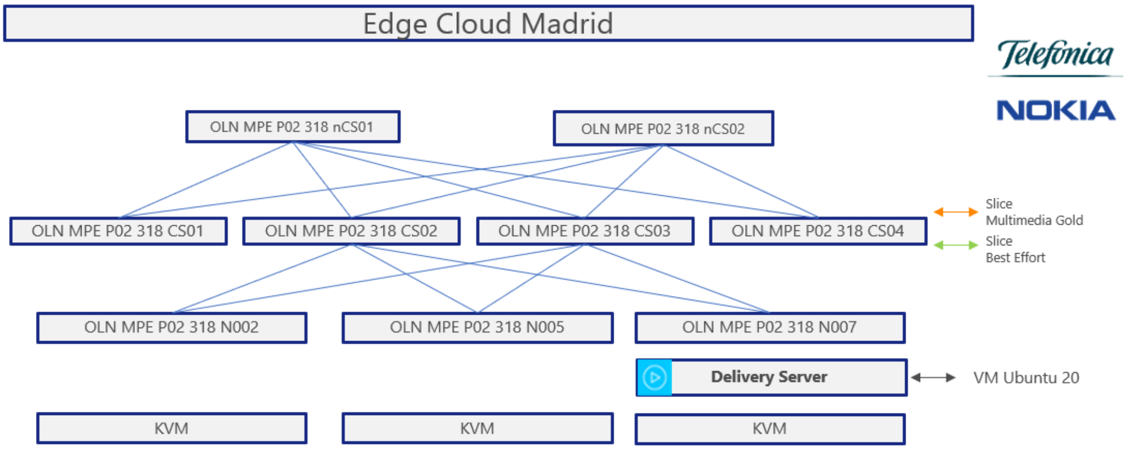


Figure 58: UC3 edge architecture

While the users will be distributed among Segovia smart venue site (on-site) and other Spanish cities: Madrid, Barcelona, Valladolid and Valencia (remote). Those users will have different network requirements based on their profile (see table).

Table 29: UC3 Network requirements

Type of User	Location	Access	Protocol	Throughput	Latency	Packet Loss
Local Producer	On-Site	5G	RTP	10 Mbps	<40ms	<1%
Remote Producer	Remote	FTTH*	RTP	50 Mbps	<150ms	<0.1%
sOn-site End User	On-Site	5G	HTTP	20 Mbps / 10 Mbps**	N/A***	N/A***
VIP End Users	Remote	FTTH*	HTTP	40 Mbps / 20 Mbps**	N/A***	N/A***
Regular End Users	Remote	FTTH*	HTTP	Best effort	N/A***	N/A***

*and Wifi-5 ** Peak / Mean ***Not Applicable

Input to the Media Delivery is received, from the Media Renderer, as H.264 / RTP / UDP, one stream per channel. Each flow goes on a different port. There are two types of outputs: RTP flows for content producers (forwarding the input), and HTTP adaptive streaming (HLS) for the end users.

To meet the different network demands, we will deploy and test performance on two different Network Slices:

- **Slice "Best-effort"**: traffic will not be prioritized on any part of the network.
- **Slice "Multimedia-Gold"**: traffic will be prioritized in as many parts of the network as possible (access, transport and core).

Content producers and VIP users (assuming on-site users are VIP too) will receive traffic through the Multimedia Gold slice, while regular users will only have access to best-effort. Network KPIs will be monitored so that producers / VIP users are moved between both slices depending on the status of the network. See D2.1 for a definition of the different test scenarios.

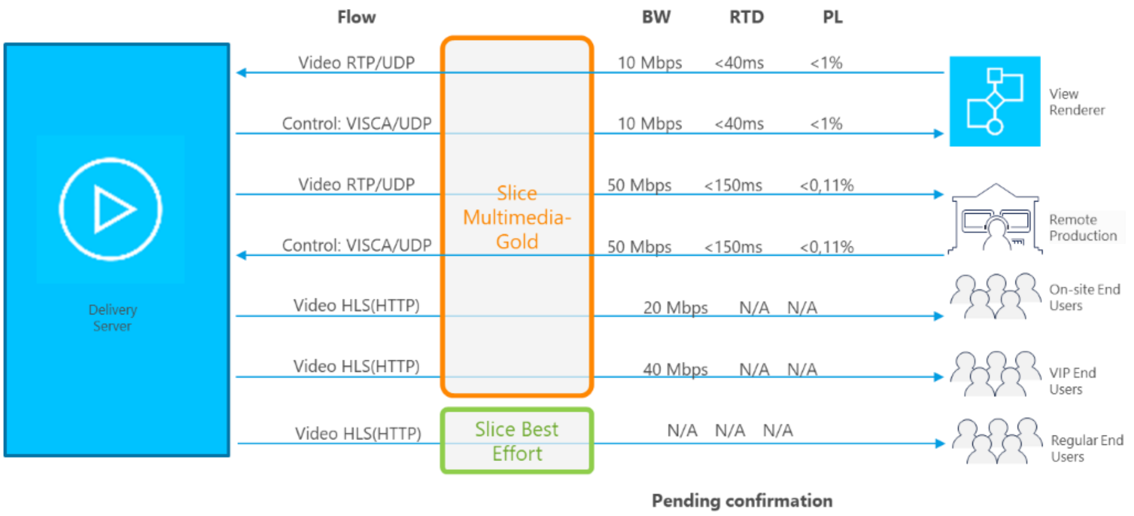


Figure 59: Network Slices flows

5.2.4 Monitoring

Systems include a monitoring module to actively monitor system performance during operation. The architecture of system monitoring is common for all monitored elements:

1. Application logs are produced with the relevant KPIs and retrieved by a Telegraf service.
2. Additionally, Telegraf service retrieves performance logs from the system (CPU consumption, network logs, disk I/O stats, etc.), both at VM and at baremetal level.
3. Telegraf logs are sent to an InfluxDB + Grafana centralized service.

The following applications will generate monitoring logs: (i) capture server, (ii) media render, (iii) media delivery, and (iv) end client. Additionally, the near-edge and edge platforms will also be monitored. More details on the monitoring capabilities for this use case will be included in deliverable D4.1.

5.3 List of Components

5.3.1 End devices

5.3.1.1 *Component: Capture server*

Figure 60 shows the main components of the capture server: (i) the cameras (currently ZED cameras); (ii) the processing server that performs all the operations associated to the capture process carried out by the capture module and the encoding and transmission module; (iii) the 5G modem.

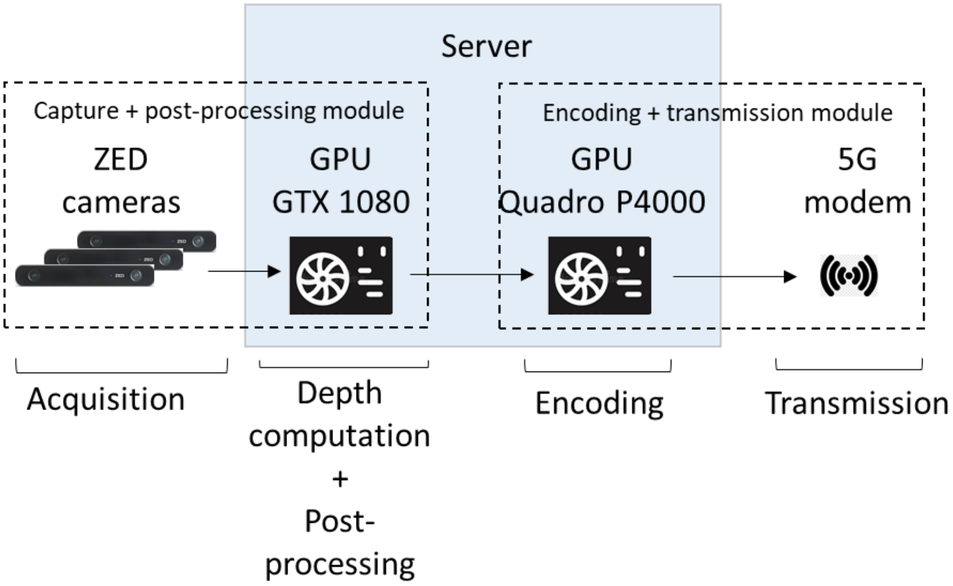


Figure 60: Capture server components associated to its main functions

Cameras

The cameras used to capture RGB + depth data are consumer electronics ZED cameras developed by Stereolabs Figure 61. These cameras capture two RGB video streams through two 2.2 MP wide-angle lenses and have a maximum 110º diagonal field of view and f/2.0 aperture. The stereo pair has a baseline of 120 mm. All cameras are connected to its corresponding capture server through an USB 3.0 port.

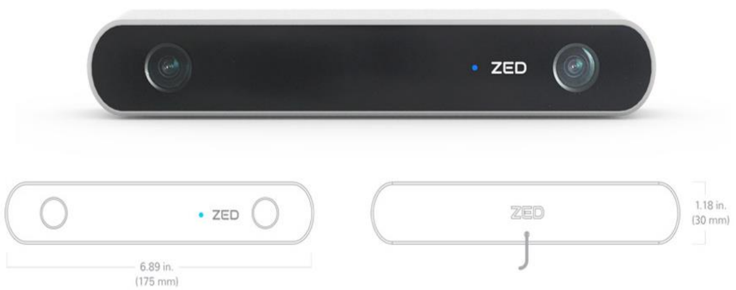


Figure 61: ZED Stereolabs camera

Processing server

Processing servers are in charge of capturing and processing RGB + depth data that will be sent through the 5G network. Each capture server is composed by the following main components:

CPU	Intel Core i7-6850K @ 3.60 GHz
RAM	64 GB
USB	2x StarTech.com PEXUSB3S44V (PCIe, four USB 3.0 ports)
GPU	NVIDIA GeForce GTX 1080 for depth computation
	NVIDIA Quadro P4000 for encoding RGB + depth data
Network card	1 Gbit/s

Currently, three cameras are connected per capture server that are bringing the capabilities of the GPUs to their limits.

5.3.1.2 *Component: Production console*

The production console is the component that allows the producer/camera operator to choose freely the desired virtual view at any moment and visualize the current view that is being rendered. It can be identified on Figure 56 labelled as “camera control” and it is composed by two main elements: (i) visualization monitor, and (ii) virtual view control.

Through the production console, the virtual camera operator is able to instantiate a virtual renderer session. Two main working modes can be instantiated: (i) online session, and (ii) offline session.

An online session allows to work on real time directly with the streams acquired by the capture servers. The received video streams are delivered to the view renderer by the stream selector module (section 5.3.2.2). On the other hand, an offline session allows to render synthetic views not based on the current real time acquisition performed by the capture servers but based on previously captured streams that have been stored on hard drive (section 5.3.2.3).

The resulting synthesized view is received from the view renderer (section 5.3.2.1) and displayed on the visualization monitor. On the other hand, the production console sends control messages to the view renderer indicating the desired virtual camera position, so the view renderer is able to render a new frame based on the new virtual camera position.

5.3.1.3 *Component: User terminal*

End users will use conventional 5G mobile phones to watch the video. In this iteration, a reduced number of users will be present in the trial location. The following is a list of tentative cell phones available for the project: Xiaomi Mix 3 5G, Huawei Mate 20 Pro 5G, LG V50, One Plus 7 Pro. They are currently in n78 band (3.5 GHz) only.

Experimental video player

For the playback of the video by the users, a specific video player will be used. This software video player has been developed by Telefonica I+D and it is based on the opensource framework Videojs. This framework uses HTML5, so the playback can be done on both personal computers and mobile devices.

This experimental video player is instrumented: it records video KPIs and sends them to the monitoring infrastructure. Details are described in D4.1.

5.3.2 Virtual Network Functions

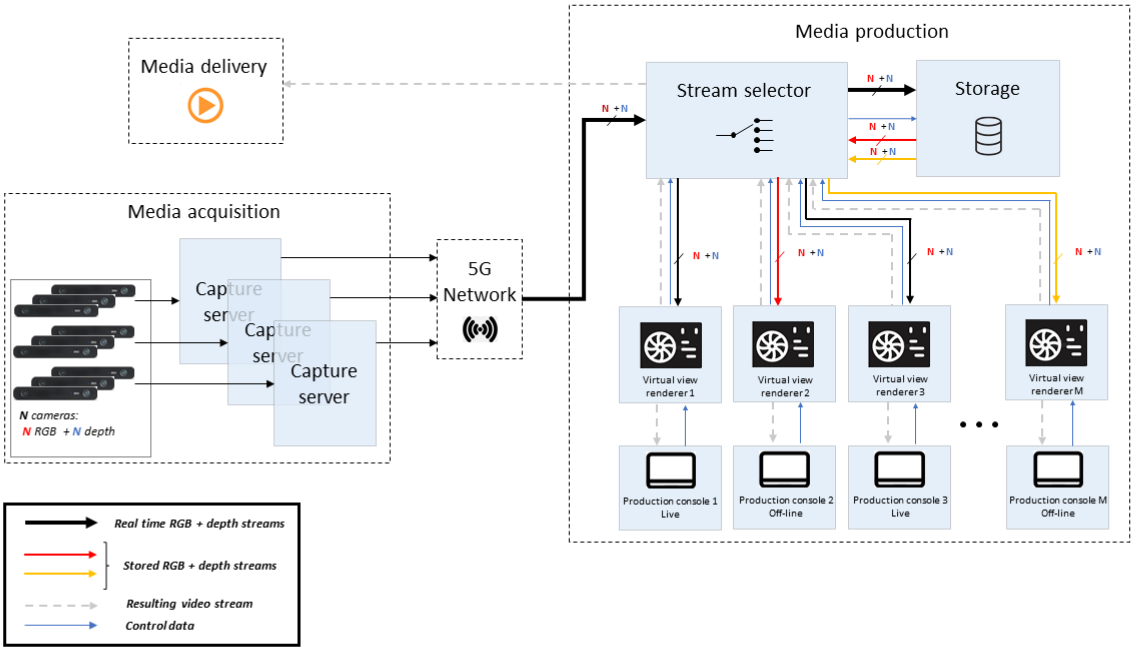


Figure 62: Virtual Network Functions integration

Edge cloud configuration details can be found in Annex B.

5.3.2.1 *Component: View renderer*

The view renderer is in charge of performing the computation of the virtual view according to the control of the production console (section 5.3.1.2). The view renderer receives as input the video streams (both RGB and depth data) and the control messages associated to the desired position of the virtual camera.

The view renderer needs the resulting information of the calibration and background modelling processes (see Annex B). The synthetic view is computed based on the information of the nearest real cameras (currently five cameras) to the current virtual camera position on a layered synthesis approach. Thus, at any moment, the view renderer needs to receive from the stream selector five RGB streams and five depth streams. The different synthesis layers are:

- Foreground: Based on the RGB and depth information captured by the cameras
- Background: Based on the background model (RGB and depth) computed on the calibration stage

Once a frame has been synthetized, it is encoded and sent to the media delivery module through the stream selector module. Also, the resulting frame is sent to the production console so the producer or the virtual camera operator is able to monitor the view of the virtual camera.

5.3.2.2 *Component: Stream selector*

The stream selector orchestrates the delivery of the FVV streams and control data among the different components of the media production, i.e., the view renderer instances, the storage module and the incoming streams from the cameras. Depending on the scenario (online or offline session) the stream selector orchestrates data delivery as follows:

- For an online session scenario, all the incoming streams from the cameras are delivered to the storage module. Also, a subset of these streams is delivered to each one of the renderers instances that are working online. The delivered streams from the cameras to the view renderer are chosen according to the desired virtual view selected on the production console.
- For the offline sessions, a subset of the stored streams is delivered from the storage module to the virtual view renderer. This sub-set is chosen according to the desired virtual view selected on the production console.

5.3.2.3 *Component: Storage*

The storage module main function is to store the captured streams to allow offline. It receives the FVV streams from the complete set of cameras delivered by the capture servers through the stream selector. On the other hand, only a subset of them (those needed to render the virtual view) are delivered to the view renderer.

5.3.2.4 *Component: Media Proxy*

The Media Proxy acts as an optional proxy module to ensure proper application-level traffic replication and connectivity at the near-edge cloud. Its main functions are:

- Replicate and route contribution traffic (output of the view renderer) to the camera control and the delivery server.
- Reverse proxy (and optionally cache) video requests from in-premise end users to the delivery server.

It uses the same modules as the Media Delivery server, but with a different configuration.

5.3.2.5 *Component: Media Delivery server*

The Media Delivery VNF performs the delivery of the rendered views to all the end users of the project, both professional (typically content producers or broadcasters) and end users. Its architecture is depicted in Figure 63. It is comprised of three main modules: (i) real-time video router and processor, (ii) multi-purpose origin server, and (iii) configuration management.

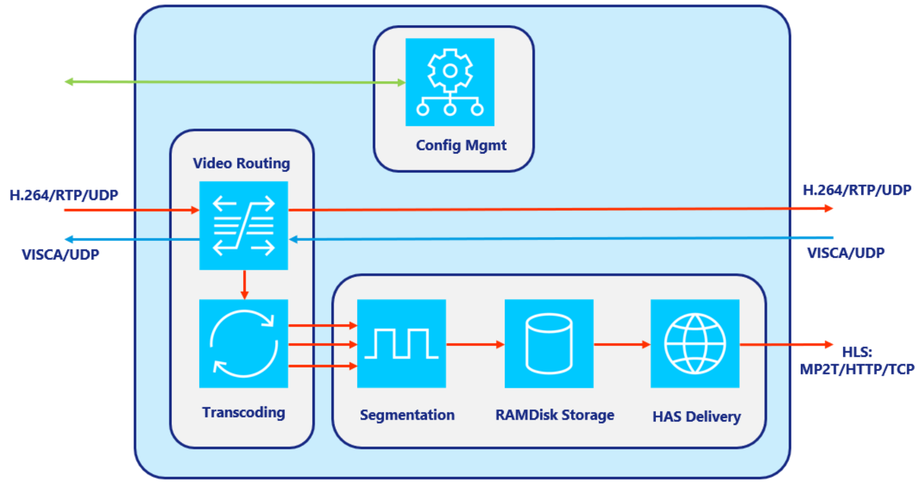


Figure 63: Media Delivery VNF

Real-time video router and processor

It is composed of two main modules:

- Video routing: a low-latency application-layer video routing module, specialized in replicating, forwarding, and adapting video over RTP transport.
- Video processing: a flexible video processing chain based on Gstreamer, which can be used to transcode and remultiplex video. In our case, it is used to generate several qualities for the video stream, and to multiplex them in MPEG-2 TS so that they can be easily ingested in hydra module.

Multi-purpose origin server

It is composed, in a simplified manner, of the following subsystems:

- Live Ingest and segmentation: It receives the live video streams (using multicast MPEG-2TS/RTP/UDP) and generates a segmented video version in a mezzanine format (MPEG-2 TS segments and a private metadata manifest).
- Local Cache: The segmented video is inserted into a local cache (in disk or memory) that can store a sliding window of several minutes of content.
- HTTP Server: The segmented video is delivered using ABR protocols (e.g., HLS, DASH) to the end client. The video can be obtained from different sources:
    - ABR protocol from an external source (the CDN). Then the HTTP server acts as an efficient reverse proxy and delivers the content to the user terminal.
    - Live ingested into the local cache. Then the HTTP makes use of a processing library ("LibPers") to dynamically generate the ABR content from the segmented mezzanine format.

Configuration management

In this first iteration of the project, it is based on static configuration files.

5.3.3 Network components

5.3.3.1 *Component: 5G CPE*

For the first iteration of the project, a RTL6305 Askey 5G mmW modem will be used. IT supports mmW and Sub 6GHz for 5G NR NSA and SA core network, offering the flexibility of combining 5G NR and 4G LTE. It can be used outdoors, as it complies with IP67 waterproof standard. It offers gigabit ethernet connectivity with the capture servers.

In later stages, Nokia FWA FastMile will be used. The Fixed Wireless Access (FWA) product family is a fixed access product family that delivers broadband and ultra-broadband (Gbit/s access) capabilities to residential and small business premises. Among this product family, FastMile 5G products use 3GPP 5G NR radio technology to further enhance the system capacity and the spectral efficiency. Additional system capacity is also achieved by supporting new frequency bands like bands n77, n78 and n79 in 3.3 – 5.0 GHz frequency; as well by supporting legacy frequency bands already supported in FastMile LTE product.

Figure 64: Askey RTL6305 (left); Nokia FastMile 5G Receiver (center) and 5G Gateway (right)

5.3.3.2 Component: 5G gNB

The configuration used for the 5G gNB is NSA 3X. It is based in E-UTRAN New Radio - Dual Connectivity (EN-DC), a technology that enables introduction of 5G services and data rates in a predominantly 4G network. UEs supporting EN-DC can connect simultaneously to LTE Master Node eNB (MN-eNB) and 5G-NR Secondary Node gNB (SN-gNB). This approach permits cellular providers to roll out 5G services without the expense of a full scale 5G Core Network.

An EN-DC enabled UE first registers for service with the 4G EPC. The UE also starts reporting measurements on 5G frequencies. If the signal quality for the UE is enough for a 5G service, the LTE eNB communicates with the 5G-NR gNB to assign resources for a 5G bearer. The 5G-NR resource assignment is then signalled to the UE via an LTE RRC Connection Reconfiguration message. Once the RRC Connection Reconfiguration procedure is completed, the UE simultaneously connects to the 4G and 5G networks. In this configuration, user data traffic directly flows to the 5G gNB part of the base station. From there, it is delivered over the air interface to the mobile device. It is also possible to forward a part of the data over the X2 interface to the 4G eNB part of the base station and from there to the UE.

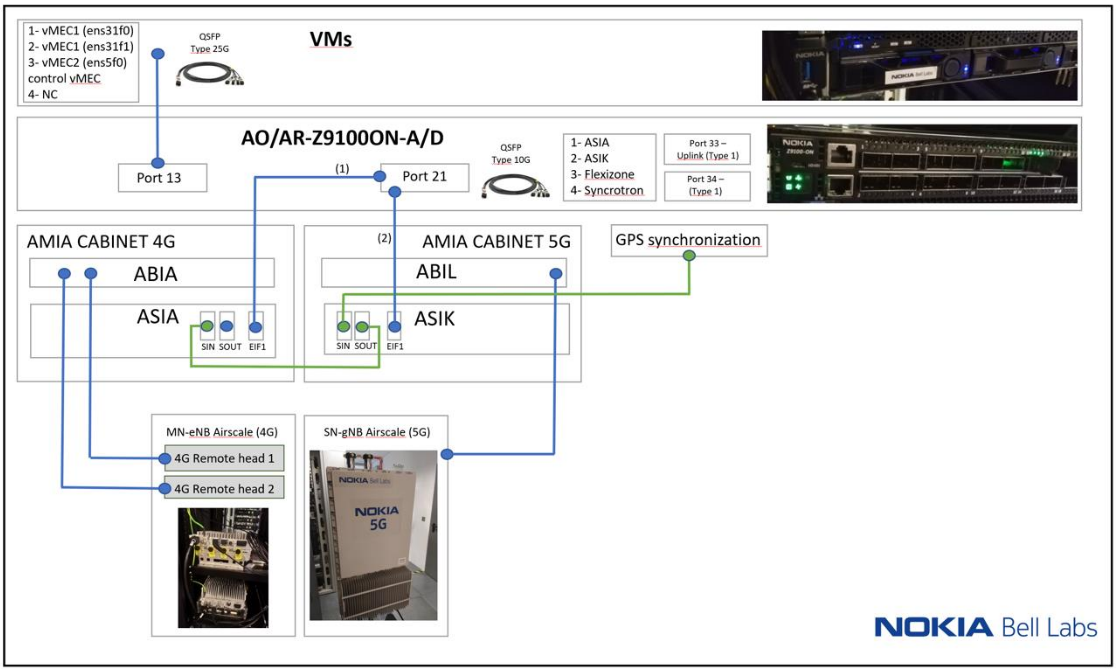


Figure 65: gNB integration within the virtualization environment

SN-gNB Airscale (5G)

The 5G antenna model used is AEUB AirScale MAA 2T2R 512AE n257.

In addition to the 5G antenna, the structure of radio 5G is also composed of ABIL and ASIK modules. ABIL is the base band module and ASIA module is the controller.

Radio 4G

The base station of 4G (eNB) is called Nokia AirScale Dual RRH 4T4R B1/3 320 W (AHEGB):

The LTE bands available are the following:

- Band 1 (2100): DL 2110-2170, UL 1920-1980. The current configuration is based in Telefónica bands.
  - EARFCN downlink: 525 (15MHz).
  - EARFCN uplink: 18525 (15MHz).
- Band 3 (1800): DL 1805-1880, UL 1710-1785. The current configuration is based in Telefónica bands.
  - EARFCN downlink: 1301 (20MHz).
  - EARFCN uplink: 19301 (20MHz).

In addition to the 4G heads, the structure of radio 4G is also composed of ABIA and ASIA. ABIA is the base band module and ASIA module is the controller.

5.3.3.3 *Component: 5G Core*

The 5G deployment for this phase of the project will be integrated with the 5G pilot infrastructure deployed by Nokia and Telefónica in Spain. This infrastructure is based on a distributed core, which includes:

- An Edge Core, located near the radio deployment (Segovia), which performs the UPF that is going to be used in the project.
- A global Central Core, shared with other LTE and 5G deployments.

The solution is implemented using the Nokia Cloud Mobile Gateway (CMG), a multi-functional packet core mobile gateway that provides increased deployment flexibility, elastic scale, high reliability, and the capacity to deliver a full range of mobile and IP services.

The CMG is deployed using CMG-a2 hardware configuration, which use the 19-inch Nokia Airframe rackmount server for rapid deployment and operational integration. The server includes Nokia-specific enhancements that make it more efficient than existing x86-based servers to run demanding services and applications. CMG implements UPF and provides SGi/N6 interface to the rest of the elements of the network, particularly to the MEC platform running the Media Production VNFs.

More details about the configuration and connectivity of the CMG are described in deliverables D4.1 and D5.2.

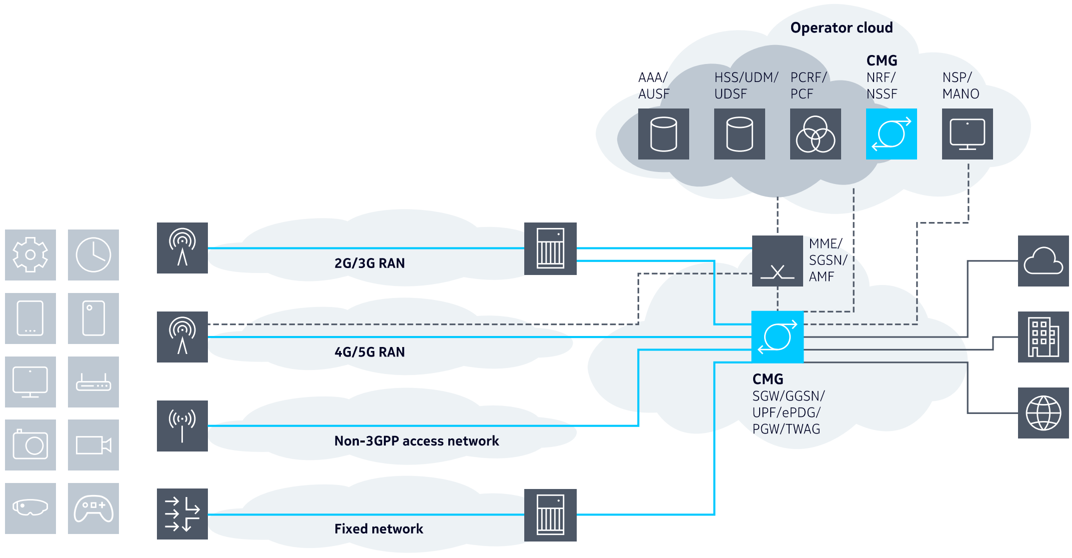


Figure 66: Cloud Mobile Gateway (CMG) in 3GPP mobile packet core

5.3.3.4 *Component: Edge MEC*

The hardware platform for the MEC is the Nokia Airframe OpenEdge server, a compact and robust x86 server platform optimized for edge computing.

Figure 67: OpenEdge 3U chassis (left) and 2U server (center), and Nvidia Tesla T4 (right)

The deployed system includes an OpenEdge 3U chassis, which can hold two 2U servers. Each 2U server has the following characteristics:

- Intel Xeon 20 cores, 192 GB RAM, 480 GB SSD
- 2x Tesla T4 GPU
- 25GbE dual port OCP NIC card CX5

The server runs Ubuntu 20.04 with an installation of MicroStack (a compact version of OpenStack) which is used to administrate and orchestrate the Virtual Network Functions (see Figure).

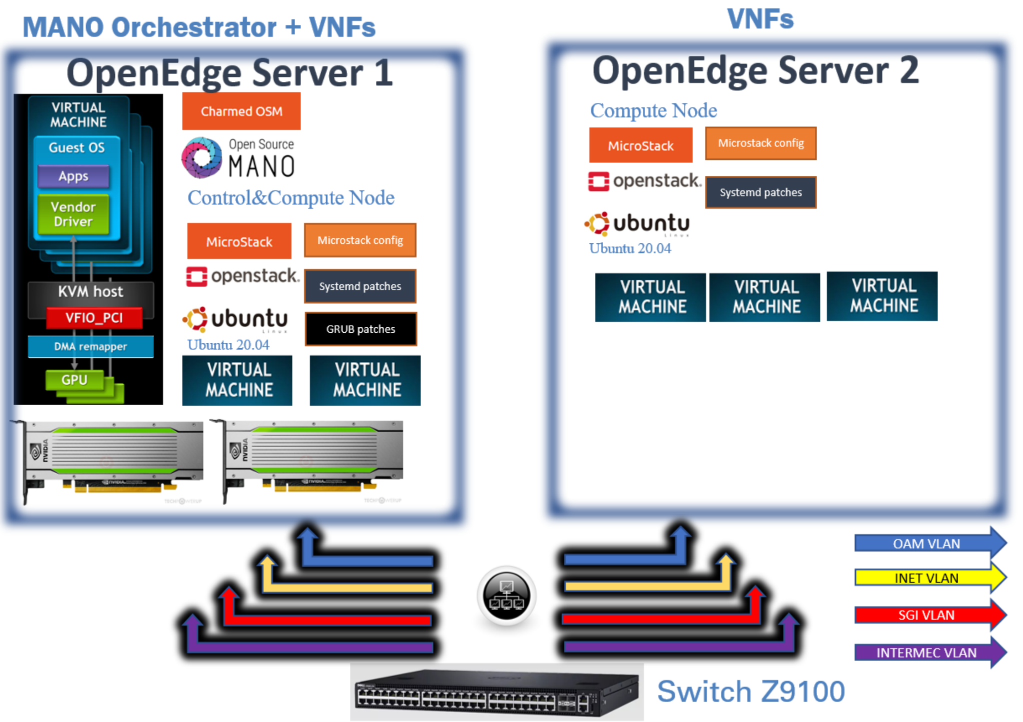


Figure 68: Diagram of the virtualization infrastructure in the framework of ETSI MEC

OpenStack software controls large pools of compute, storage, and networking resources throughout a datacenter, managed through a dashboard or via the OpenStack API. OpenStack works with popular enterprise and open-source technologies making it ideal for heterogeneous infrastructure. Microstack is a full OpenStack in a single snap package. Microstack is an upstream multi-node OpenStack deployment which can run directly on your workstation. Although made for developers, it is also suitable for edge, IoT and appliances. It is a solution from canonical (Ubuntu) which consists in a simple hardcoded configuration type OpenStack installation based on snap. Snaps mounts a read only file, so no configurations changes are allowed. As a consequence, Microstack installation has been modified to support the specific functionalities required for the project, in particular, the visibility of GPUs from the VNFs.

5.3.3.5 *Component: Delivery cloud*

Physical architecture of the Edge Cloud is made up of a layer of networking, (6 switches in leaf-spine model), a compute layer, (7 OCP servers), and a storage layer (storage server and JBOD)

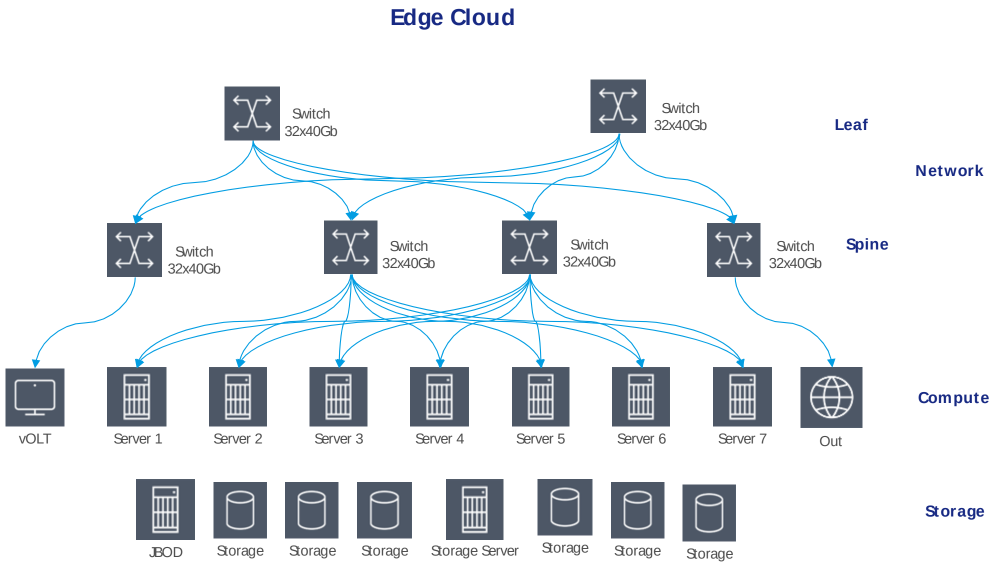


Figure 69: Leaf-Spine edge compute architecture

Edge platform is based on three main modules: (i) compute virtualization Subsystem based on OpenNebula, (ii) storage virtualization subsystem based on GlusterFS, and (iii) network virtualization subsystem based on SDN ONOS.

Compute virtualization

It is an Opensource platform based on OpenNebula software for cloud computing. It is oriented to distributed and heterogeneous datacenters, providing virtual infrastructure to build private, public, and hybrid clouds. Further, it is Opensource under Apache, and we use standard Linux KVM, as virtualization manager.

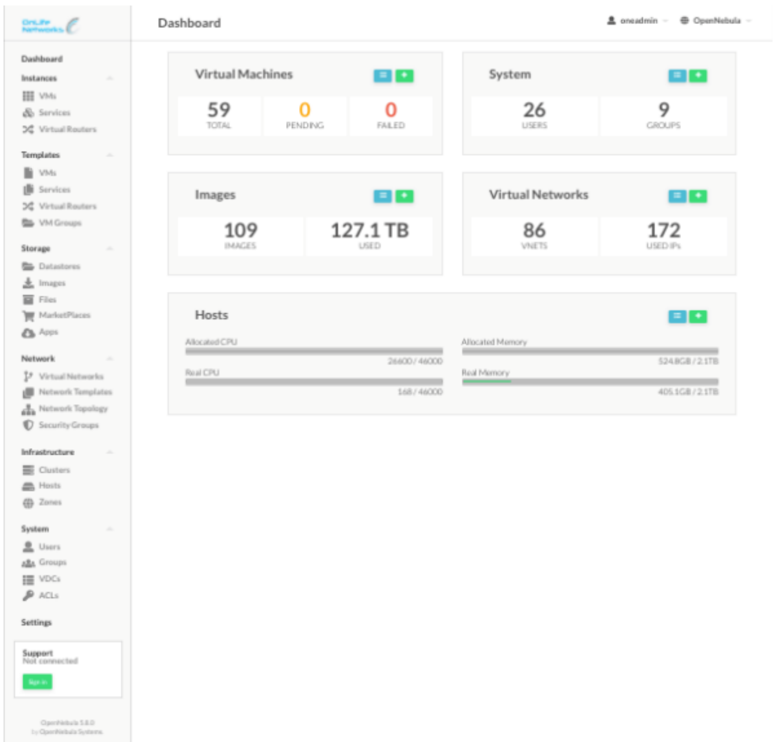


Figure 70: OpenNebula dashboard

Storage virtualization

For this layer we use GlusterFS software. This software is installed and configured into server hosts directly. For storage network GlusterFS replication is done using two separate networks. First network managed by SDN. And a second auxiliary network using the management switch. Each network has a different VLAN assigned. Also, Gluster configuration is done using at least two cluster for load balancing purposes.

Edge Cloud Storage Distribution

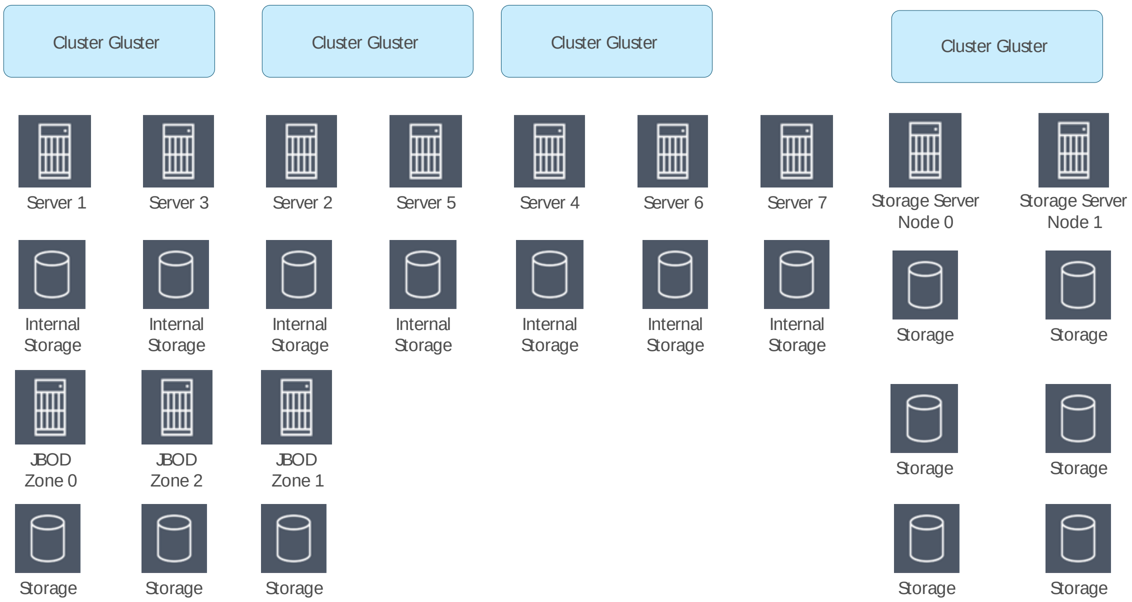


Figure 71: Edge cloud storage distribution

Network connectivity

It is a subsystem based on ONOS, Open Networking Operating System, ONOS provides the control plane for a software-defined network (SDN).

In the Edge cloud network solution, ONOS is used as the SDN controller, in charge of managing the switch fabric. ONOS provides a series of functionalities and software models to provide communication services to end hosts and networks.

We use an ONOS application called CLOSfwd, that is responsible for managing the CLOS fabric of Edge Cloud switching. In the case of the switching fabric, it is sought to create paths between pairs of end nodes.

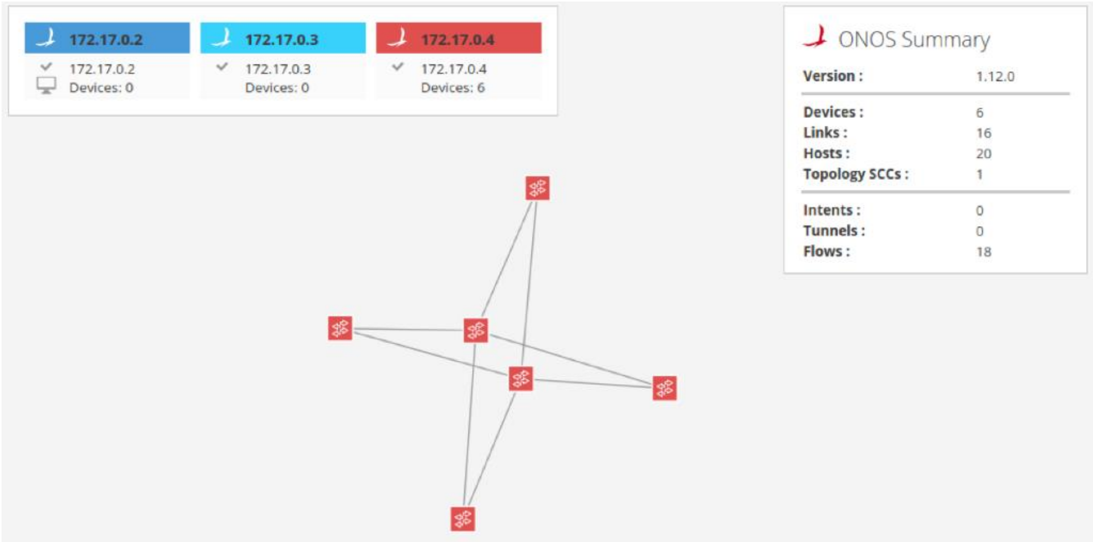


Figure 72: SDN GUI

5.3.3.6 *Component: End-to-end SDN*

Architecture of E2E SDN is defined based on these building blocks:

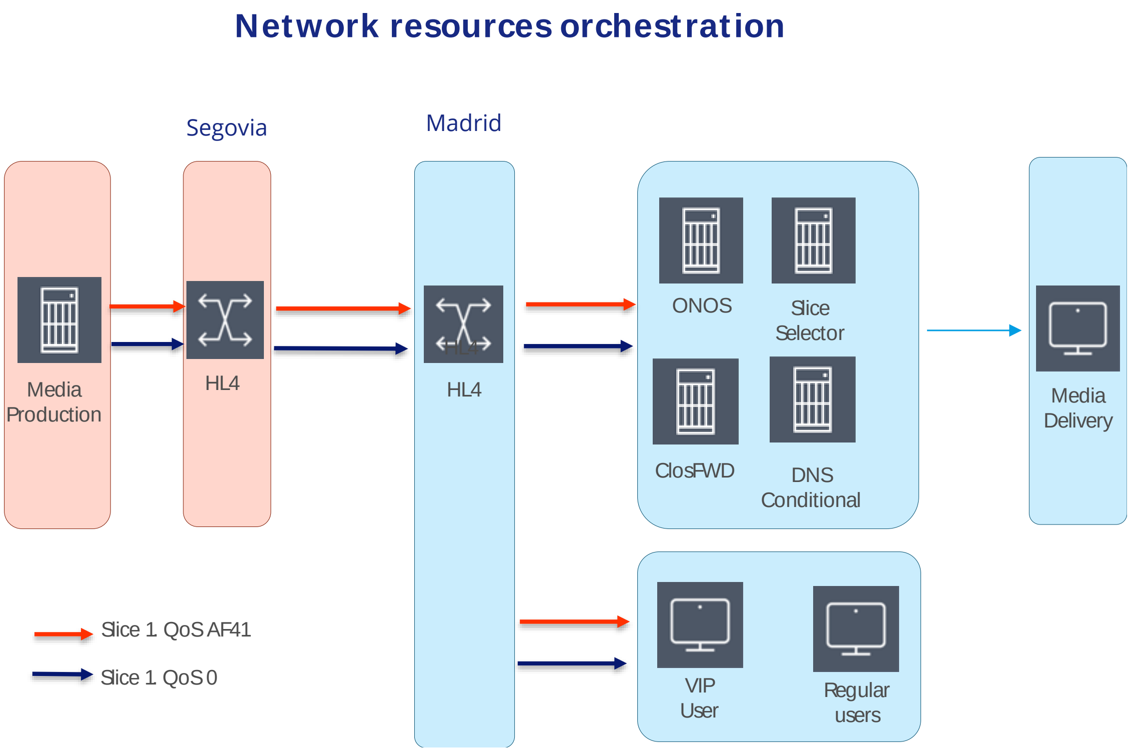


Figure 73: Network resources orchestration

- **ONOS**: used as the SDN controller, in charge of managing the switch fabric, as described in section "Network connectivity subsystem".
- **ClosFwd**: application of the ONOS environment is responsible for managing the CLOS fabric of Edge Cloud switching.

- **Slice Selector**: software application based on OVS flows with enough capacity to select the slice for the request of the users. Currently in development phase. Regarding network orchestrator slice selector, a development to insert into OVS flows needed. Logic selection will be based on IP address.
- **DNS Conditional**: implemented using opensource software bind and several views configuration in order to response correctly. That means that responses for VIP users will be different that for regular users.

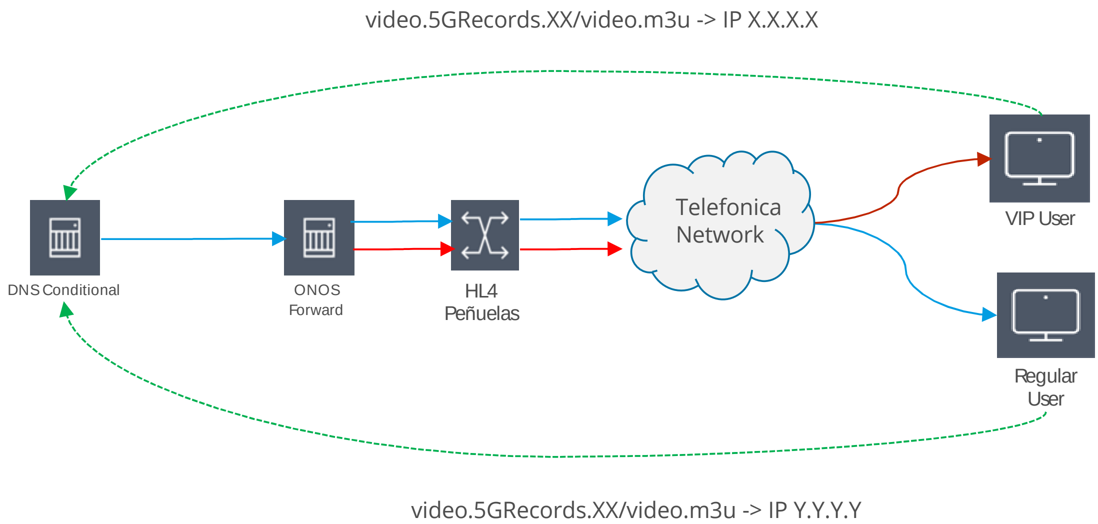


Figure 74: DNS conditional

5.4 Interfaces

5.4.1 Content interfaces

Content interfaces describe the connection between the different elements of the data path. They are depicted in the following figure. Video streams are shown in red. Some interfaces (A and B) also require control streams, depicted in blue.

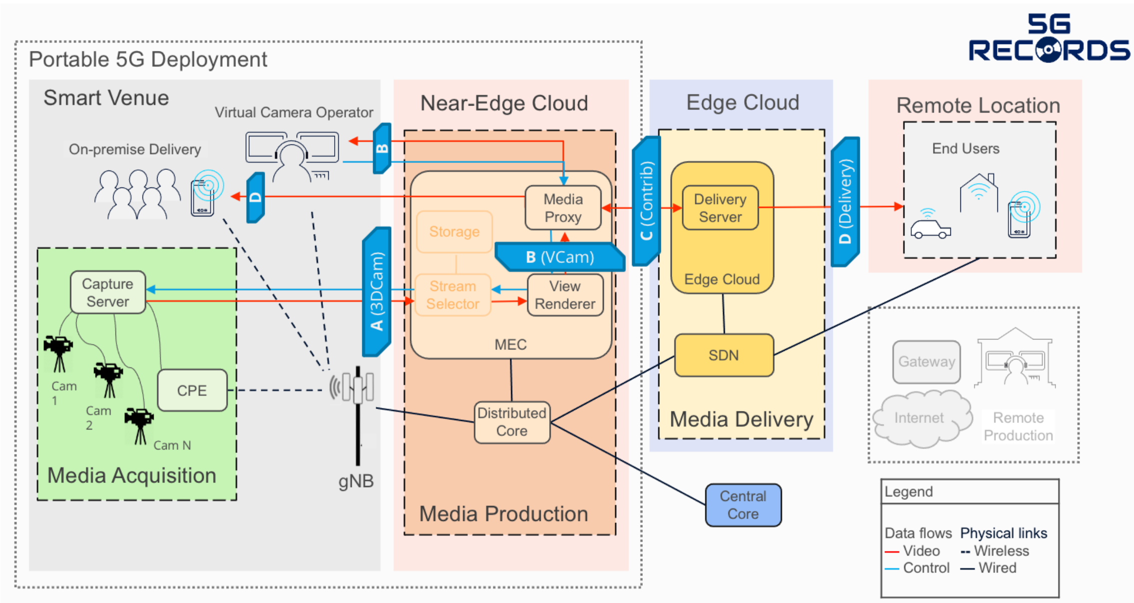


Figure 75: UC3 interfaces

5.4.1.1 *A: 3DCam interface*

This is the interface between the Capture Server and the View Renderer. It comprises:

Video (per physical view)

Each camera produces two different streams that are sent in two separate RTP streams: one for the RGB data and the other for the depth data.

The RGB encoder implementation is based on an AVC lossy encoding scheme which main configuration parameters are:

- Low latency scheme: No B frames. GOP length of 300 frames.
- Target bitrate between 5 and 15 Mb/s. Two pass VBR.

The depth encoder encapsulates s 12-bit depth on a regular 8-bit AVC lossless encoding profile. The depth encoder main configuration parameters are as follows:

- Low latency scheme: No B frames. GOP length of 300 frames.
- Lossless coding. No control of target bitrate.

The resulting NAL Units are mapped directly onto RTP packets. RTP MTU is configurable, with a maximum of 1472 bytes, to avoid IP fragmentation. RTP is send over UDP/IP/5G-RAN using unicast communication. Capture timestamps for each frame are used as the RTP timestamp after a 90 kHz clock conversion.

Control (per capture server)

These control messages imply two main categories:

- Initialization controls messages (which only occurs at the beginning of the session): configuration files for each capture server, computed background on the calibration stage, and synchronization messages between capture servers.
- Adaptive camera switching messages: every time the current virtual camera position changes, the virtual renderer sends a message to the capture servers to select the subset of cameras needed to compute the virtual view.

These communication between the different modules is performed using Message Passing Interface (MPI) standard.

5.4.1.2 *B: VCam interface*

This is the external interface of the View Renderer, which offers the interface of a virtual camera.

Video (per virtual view)

Each virtual camera is sent to the camera operator in a single unicast H.264/RTP/UDP/IP stream. The encoder configuration parameters are as follows:

- Low latency scheme: No B frames. GOP length of 300 frames.
- Contribution quality. Target bitrate between 10 and 20 Mb/s. Two pass VBR.

Control (per virtual view)

The virtual view control data transmits the desired virtual view, chosen by the producer/camera operator, from the control console to the associated view renderer instance. The message is based on TCP protocol containing the camera movement command. The virtual renderer receives the message and synthetizes a new virtual view based on the command.

5.4.1.3 C: Contrib interface

This is the contribution interface. Each virtual camera is sent as a high-quality stream (contribution quality) to the content producers and the media delivery.

Video (per virtual view)

Each virtual camera is sent to the media delivery in a single RTP stream. The stream is a replica of the VCam video stream described before, and therefore it has the same characteristics.

5.4.1.4 D: Delivery interface

This is the delivery interface. Each virtual camera is sent using HTTP Adaptive Streaming to the end users.

Video (per virtual view)

Each virtual camera is sent to the media delivery in HLS. The baseline transcoder configuration will be:

Regarding delivery, it is possible to start (as a proposal) for the first phase:

Table 30: Associated delivery bitrate and reference bitrate per resolution

Quality ID	Resolution	Reference Bitrate (Mbps)
SD	960x540	1.5
HD	1280x720	3.0
Full-HD	1920x1080	6.0
3K	2560x1440	12.0
4K	3840x2160	24.0

- Distribution-quality scheme. IBBBP GOP structure. GOP length of 300 frames.
- Number of qualities and target bitrate: configurable, a reference resolution-bitrate ladder is provided in Table 30.
- Segment size: 6 seconds.

5.4.2 Network topology and slices

The content interfaces described need to be mapped to IP interfaces in the logical network topology.

5G UEs (CPE, Camera control device, end user devices) obtain a public IP address at the core UE function. Internal traffic (e.g., for the MEC infrastructure) is routed to the SGi network at the near-edge cloud, while public traffic is routed via the central core location to the internet.

The near-edge cloud contains several sub-networks (VLANs):

- SGi: connectivity between UEs and MEC VNFs
- MEC: internal connectivity between VNFs
- Internal Gold: connectivity with edge cloud with guaranteed QoS
- Internal Best Effort: connectivity with edge cloud without guaranteed QoS

In the edge cloud, besides end points to internal Multimedia Gold / Best Effort networks, there are two additional VLANs:

- Public Multimedia Gold: connectivity with global end users, with guaranteed QoS
- Public Best Effort: connectivity with global end users, without guaranteed QoS

End users in remote locations can access the delivery server via either network. The SDN will manage the allocation of each user to its specific slice. Both are publicly available on the internet (i.e., they provide public IP addresses) via routing; however, requests coming from the public internet will always be handled by the best effort slice.

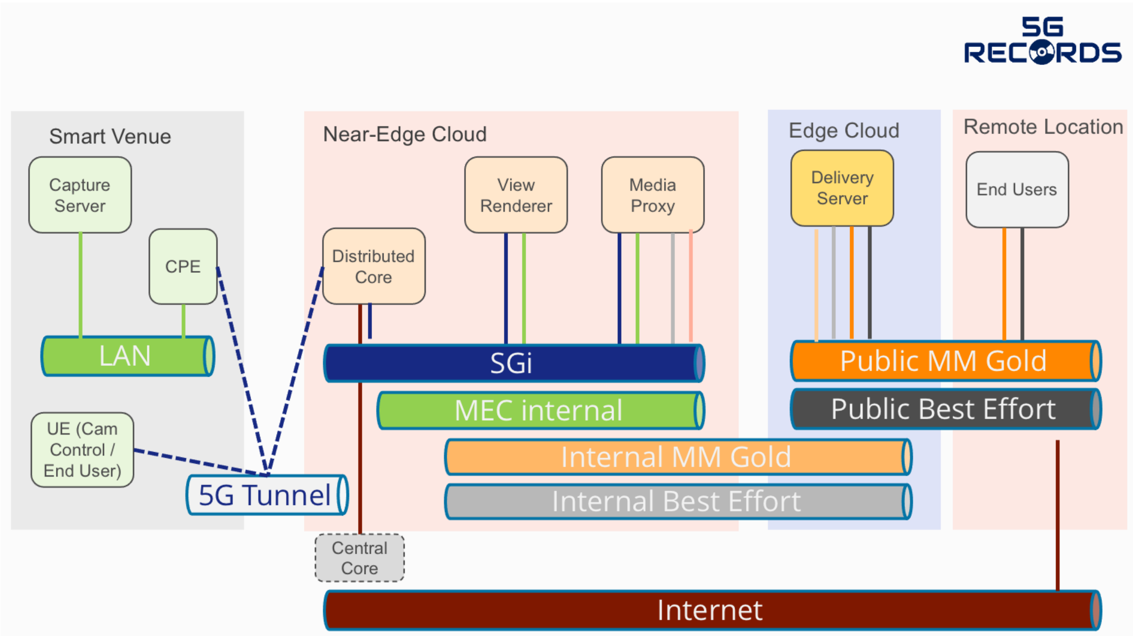


Figure 76: UC3 slices within the network topology

5.4.3 Management and orchestration interfaces

Combining all the elements, the flow is presented in Figure 77. Interface Orchestration is done thanks to a communication between the Influx DB as monitoring component and two other elements Conditional DNS and slice selector. An alert, previously configured (TBD) will be trigger two actions, change a DNS record inside DNS and internal tables in Slice selector. All the communication is TCP/IP in a dedicated and private monitoring network. The relation between components is:

- **SDN:** ONOS is used as the SDN controller and it´s connected to all networks.
- **Slice Selector**: Software application based on OVS flows with enough capacity to select the best slice suitable. Connected to slices gold and best effort internal and external, and to the internal video delivery.
- **DNS Conditional** connected to best effort slice. Response will be different based on network status and type of user.
- **Video Player** connected to best effort slice. Only to download the player.
- **Media delivery:** delivery of the video, connected to the internal video network.

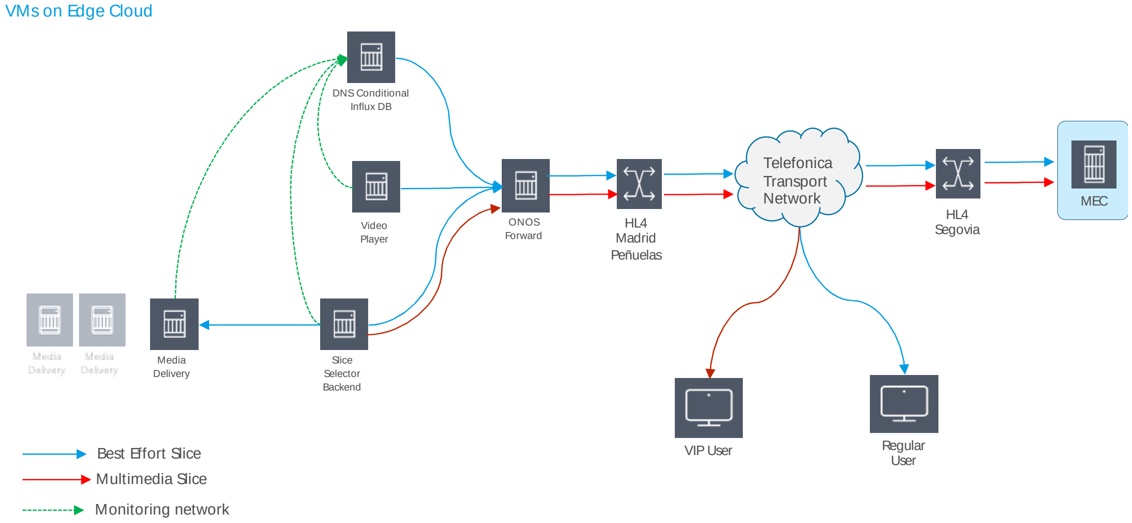


Figure 77: Network management architecture

5.4.4 Monitoring interfaces

UC3 monitoring interfaces are described in deliverable D4.1.

6 Multiple UC shared component: Media Operational Control Gateway (MOCG)

6.1 Task purpose

Task T3.4 is concerned with providing a joined-up approach to “Media Orchestration and Control” (MOC) functions that allow effective use of media devices on 5G networks. These functions include:

- Media device registration
- Discovery of media devices
- Connection of media devices
- Configuration and Control of media devices
- Monitoring of media devices
- Synchronisation of media and data flows
- Authentication and authorization
- Resource management and provisioning

Professional media production will in the future require more flexibility over where and how its operations are provided, so “effective use” includes operation where there are multiple networks, which could include 5G, wired, cloud-based and home networks (Figure 78).

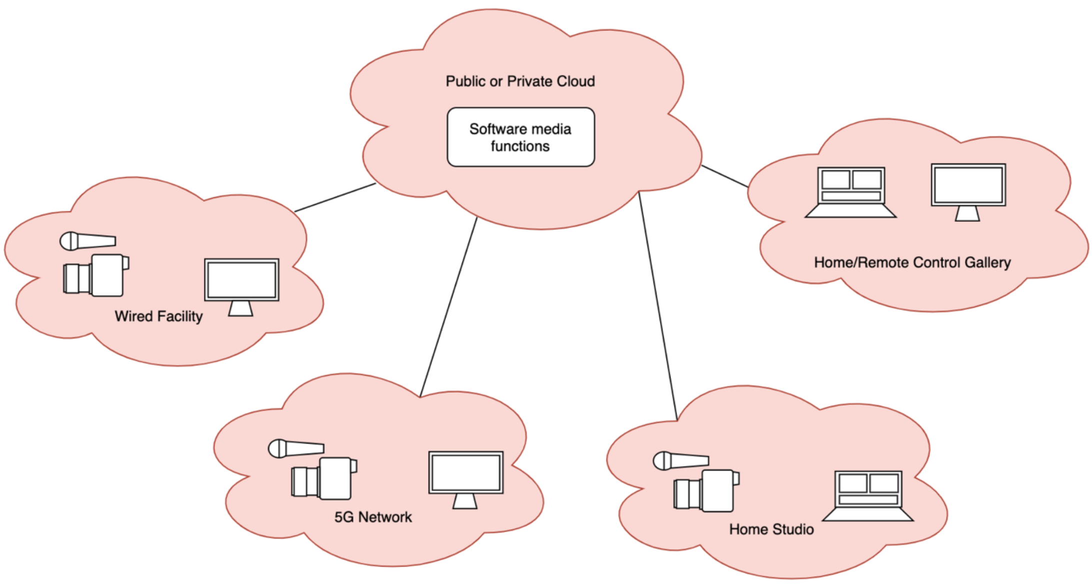


Figure 78: Production across multiple networks

5G-RECORDS’s Use Case 2 is an example of this, and Figure *79* shows a vision of a consistent set of media orchestration and control functions being available to a production that is using both wireless 5G and wired media networks.

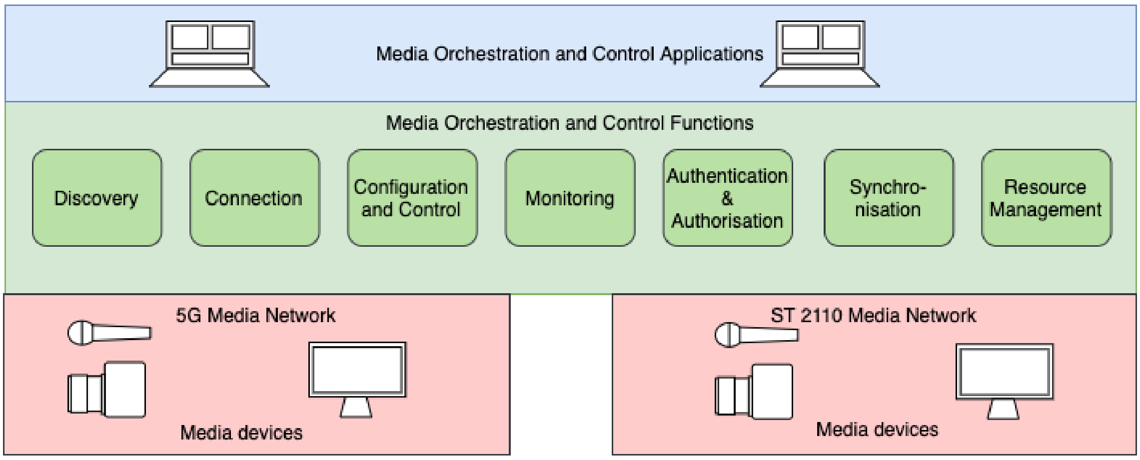


Figure 79: Common Media Orchestration and Control (MOC)

6.2 Architectural Approach

6.2.1 Models for Interoperability

The broadcast industry's adoption of live networked media systems is still relatively immature, and there is not yet a consistent and complete approach to describing architectures and the interfaces required for interoperability. However, there are some helpful steps towards this including:

- The Joint Taskforce for Networked Media's Reference Architecture [7]
- The EBU Technology Pyramid for Media Nodes [8]
- The AMWA Network Media Systems Template [9]

The last of these is a recent development and worth further discussion here as it is related to the 5G-RECORDS MOC work. The Advanced Media Workflow Association developed the Network Media Systems to provide developer, integrators and end users with an overview of the elements required for a networked media system to be practical. Figure 80 shows the main blocks of functionality that are required and Figure 81 provides more detail. The functionality is represented to show stacked layers:

- Media and Infrastructure
- Control
- Monitoring
- Security

The Network Media Systems Template provides a good fit for the required functionality identified for 5G-RECORDS MOC and will help the project identify any further functionality required and ensure consistency of terminology.

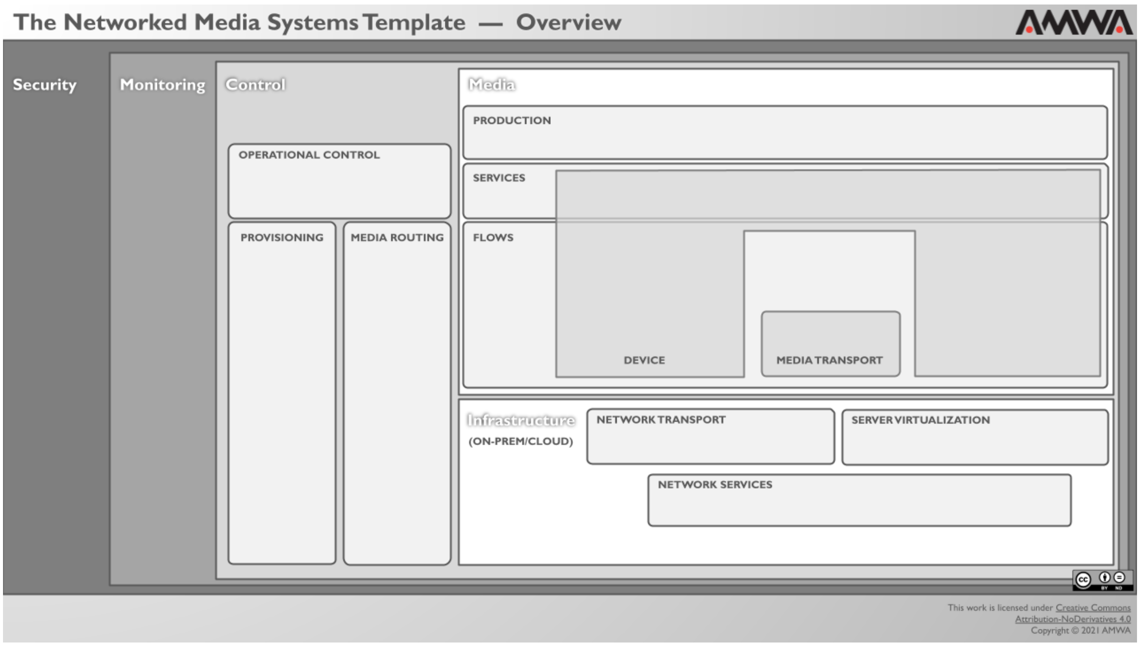


Figure 80: AMWA Network Media Systems Template – Overview

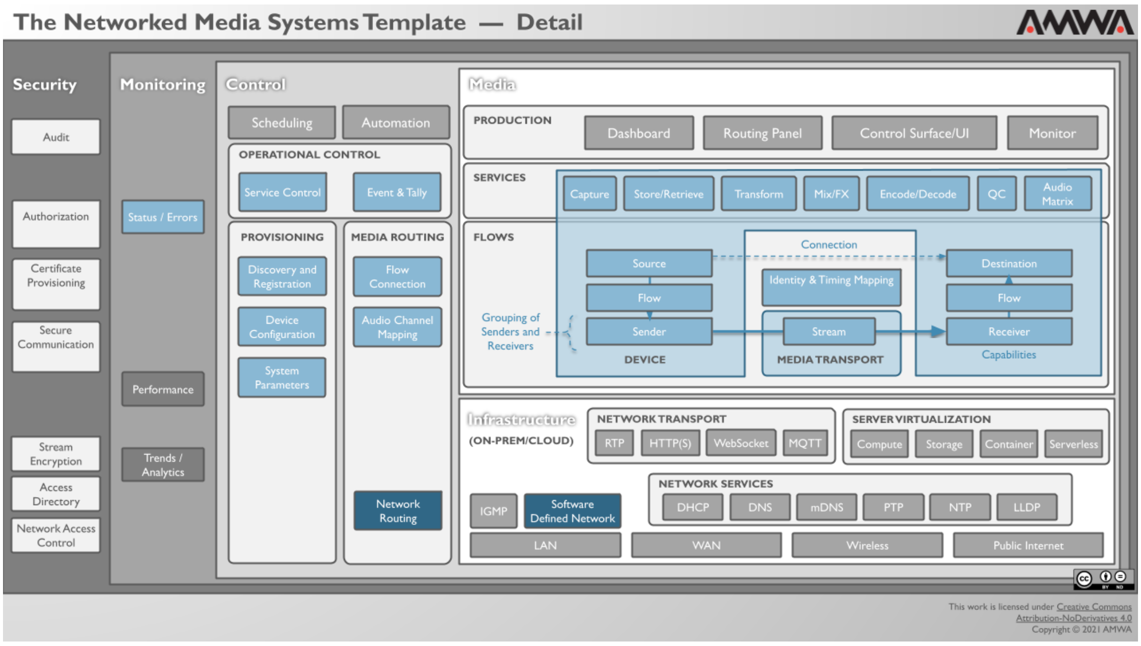


Figure 81: AMWA Network Media Systems Template – Detail

6.2.2 Gateways

5G-RECORDS's architecture includes Gateways to provide interfaces between systems and networks. The Figure 82 below shows a simple example.

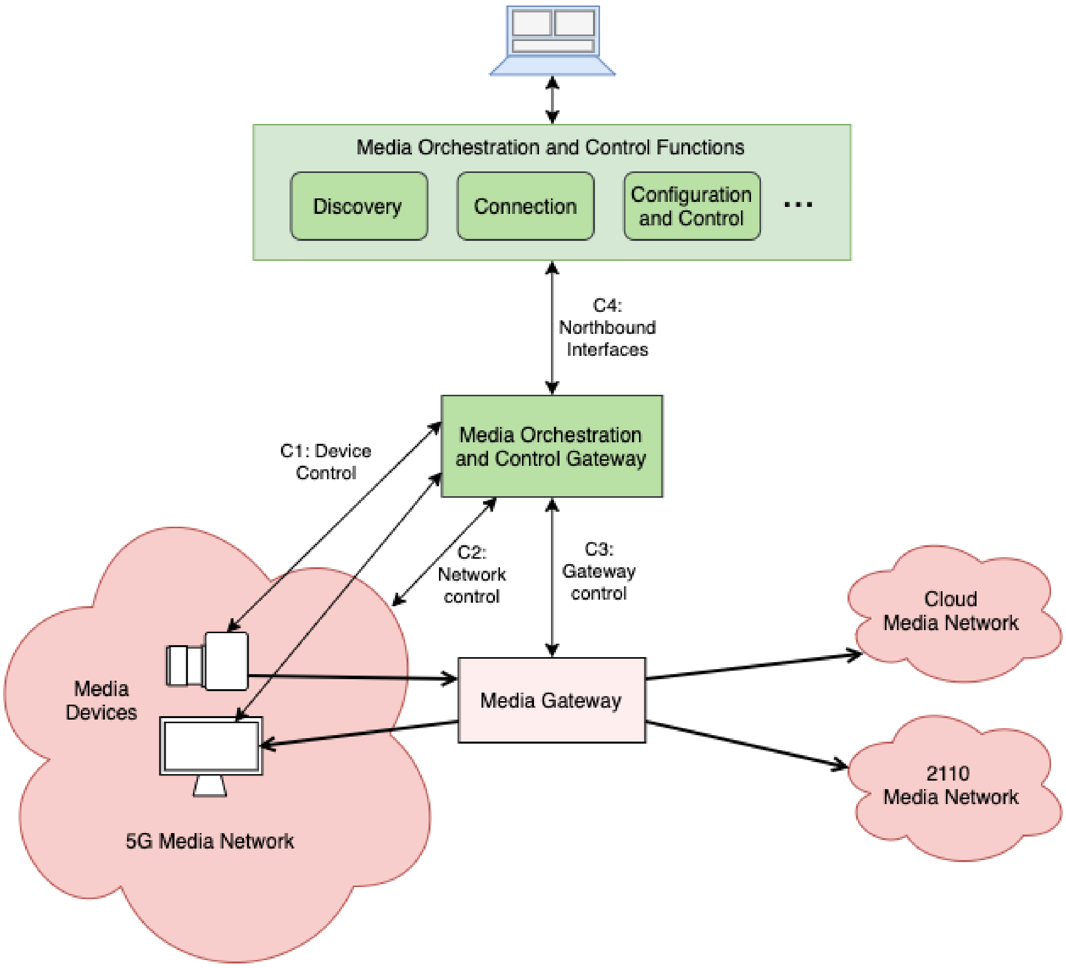


Figure 82: Media Gateway & Media Orchestration and Control Gateway

The Media Gateway is responsible for:

- Receiving media streams from Media Devices on the 5G network and sending to other networks, or vice-versa,
- Handling address translation,
- Conversion to/from multicast,
- Conversion between video and audio compression formats (and uncompressed),
- Conversion between streaming formats (e.g., to/from RIST).

The Media Orchestration and Control (MOC) Gateway is responsible for:

- Ensuring Media Devices are authorised for use,
- Ensuring Media Devices are registered and available,
- Ensuring that the network resources are available to connect Media Devices,
- Providing a connection of Media Devices through Media Gateways,
- Providing a connection for control of Media Devices,
- Providing a means to monitor Media Devices and their use of network resources,
- Providing control/monitoring protocol and message translation.

The Media Gateway and MOC Gateway are software components that can be deployed as best fits the use case, and there may be multiple Gateways. Some scenarios may require specialist hardware, as is the case for the high bandwidths and packet timing of ST 2110 video in Use Case 2.

6.2.3 Gateway Handlers

Both the Media Gateway and MOC Gateway will create Handlers on demand to support Media Devices and the Streams going to / from them, as they are connected and disconnected. See Figure 83 for an example.

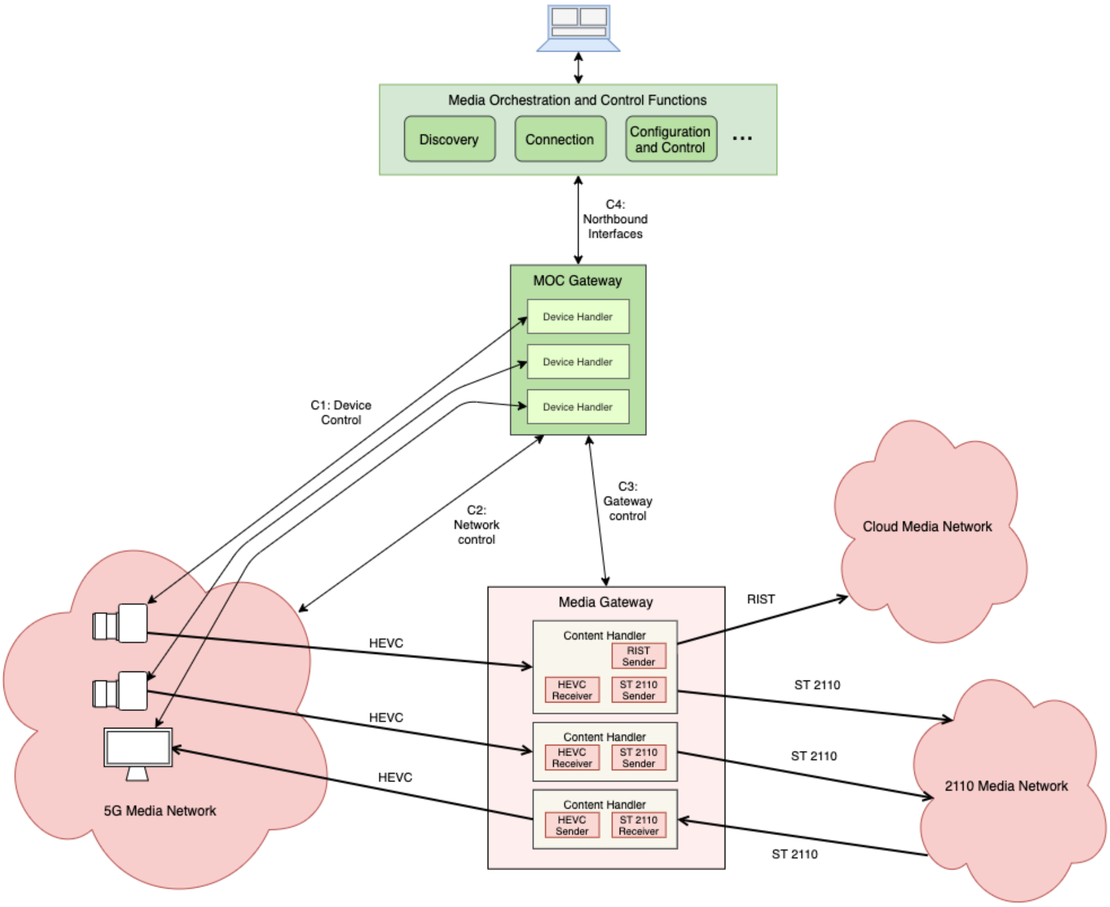


Figure 83: Gateways and Handlers

Each Content Handler in the Media Gateway handles a single media element, e.g., video or audio (so if a camera sends audio multiple Content Handlers will be used).

5G-RECORDS has decided to use HEVC within a native RTP Payload format (RFC 7798) as video codec on 5G. AAC with a native RTP payload format (RFC 6416) is considered as audio codec. Audio, video and other elementary streams are handled separately in the Media Gateway; this is suitable for typical production operations and helps support componentised and cloud-friendly design.

The system design is flexible, and the codec can be simply changed to other, even vendor specific codecs. Synchronisation between component streams is achieved by timestamps (see 6.3.5).

The Media and MOC Gateways are implemented using separate software components, deployed using containers and communicating using lightweight message queues. This allows flexibility in where the gateway functions can be deployed and is suitable for load-balancing and other scalability and resilience measures (see 6.3.7). This approach also helps with development of the different functions required withing the gateway across the consortium as the message APIs formalise the interface between the components.

Figure 84 show more detail of the functional parts that are expected for MOC Gateway, and the following sections go into more detail of functionality.

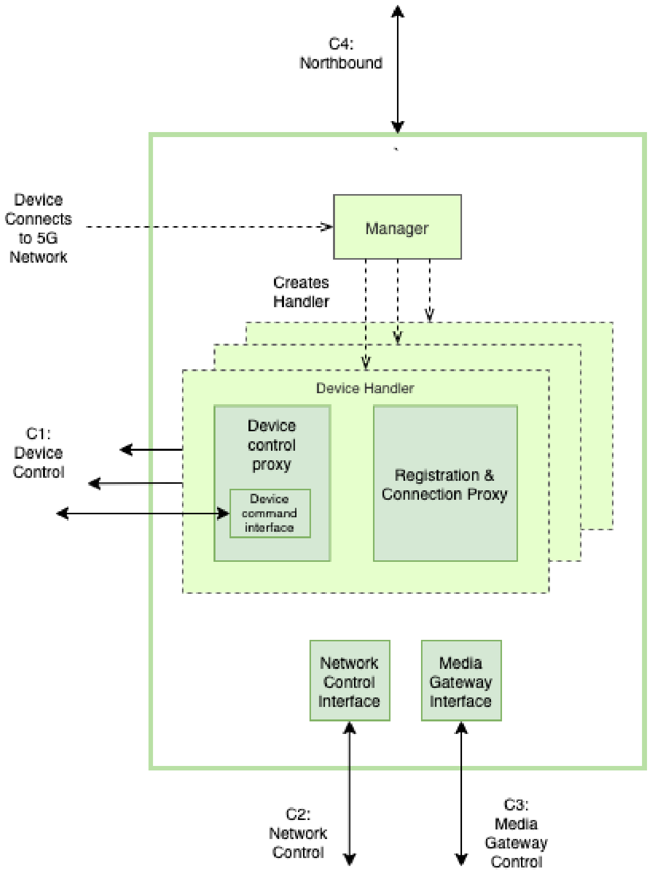


Figure 84: Logical Architecture of MOC Gateway

6.3 Media Orchestration and Control Functions

6.3.1 Registration and Discovery of Media Devices

To make a media device such as a camera available for use in a production it first needs access to the network. For a 5G network, the device's modem requests and is granted authorization to access to the 5G network, which establishes a data link (PDU Session) to the device modem, providing it with an IP address.

The 5G-RECORDS project has yet to study in detail how media producers want to manage access to the 5G Network. The 5G Network is setup as a private network, which can be a Standalone Non-Public Network (S-NPN) or a Public Network Integrated Non-Public Network (PNI-NPN). Traditional 5G devices contain a SIM or an embedded SIM card, which contain network access credentials. Depending on device configuration, the 5G device may establish one or more IP connections to data networks (PDU Session). PDU sessions are comparable to VLANs, which also allow the creation of multiple IP connections on a single physical interface. See 6.3.6.

Once the device has an IP address it contacts the MOC Gateway to register. Ideally the device would use the NMOS IS-04 Registration API directly, and the project is talking with camera manufacturers about their plans to support this. In the interim, as currently few cameras support NMOS natively, the device control interface (C1) will initially build on the camera control protocol discussed in section 6.3.4.1.

The MOC Gateway creates a Device Handler to communicate with the device and finds its identity and initial configuration. The Device Handler registers information about the device using the IS-04 Registration API, allowing the device to be used in the production. A controller application can then discover the device and query its information using the IS-04 Query API. In effect this provides an "NMOS proxy" for the device, handling any message and address translation required to access the IS-04 Registry. Figure 85 and Figure 86 show an example of Registration and Discovery. 5G-RECORDS will build upon open source NMOS implementations:

- The MOC Gateway will use NMOS Node and Registration code from Sony's C++ implementation (https://github.com/sony/nmos-cpp, [10]). Modifications will be made as required to support proxy operation.
- The C++ implementation also provides an NMOS Registry, and a BBC Python implementation (https://github.com/bbc/nmos-registration, [11]) may also be used for testing.
- Sony's JavaScript client (https://github.com/sony/nmos-js, [12]) provides a starting point for testing discovery.

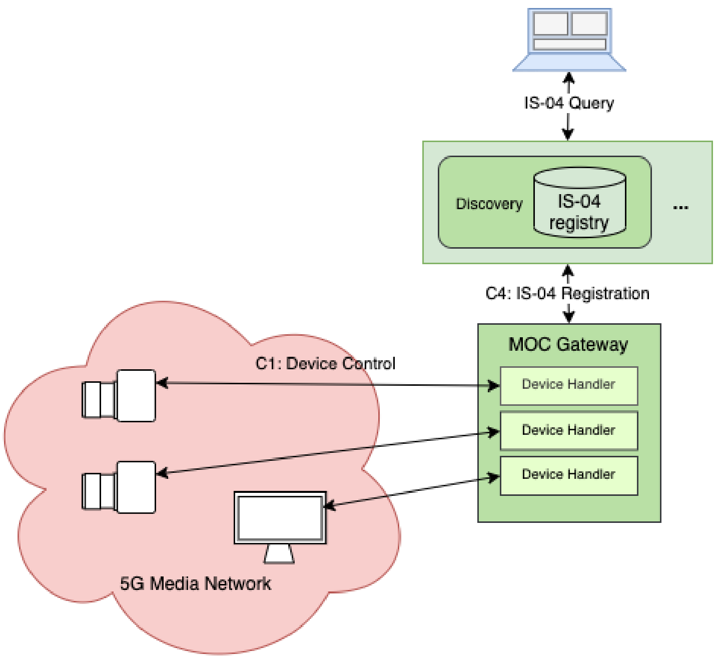


Figure 85: Registration and Discovery

Although not shown on these diagrams, the MOC Gateway also can register the Media gateway using IS-04. The Project will determine whether any extensions or modifications to the IS-04 specification should be recommended as a result of this work, for example to help with address translation.

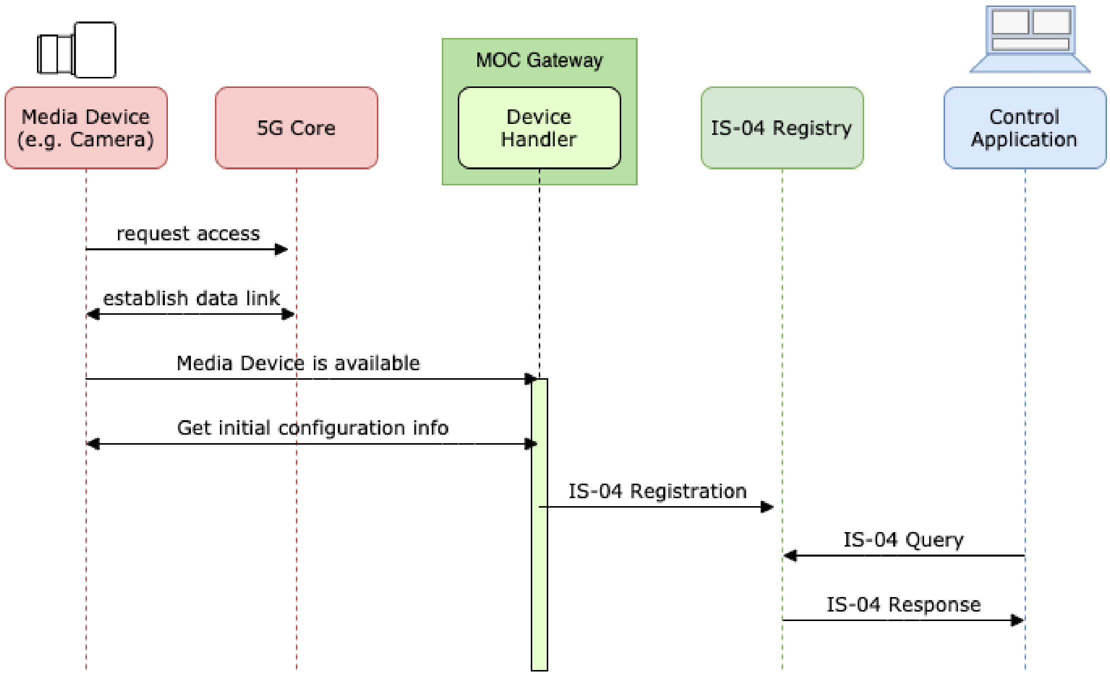


Figure 86: Registration and Discovery Sequence

6.3.2 **Connection of Media Devices**

Once the control application has discovered the media device it can connect it by using the IS-05 Connection API. IS-05 is a flexible API, capable of managing different types of protocol, although currently is typically used to make connections of multicast RTP streams:

- The controller provides the sending device with information about the required connection.
- The sending device returns a "transport file" (typically an SDP document) with details of the stream that is being sent, or will be sent.
- The controller sends the transport file to the receiving device.
- The controller activates the connection (for a multicast RTP connection the receiving device will perform an IGMP join).

A detailed sequence diagram for the above is shown at AMWA's website: https://specs.amwa.tv/is-05/v1.1/docs/1.0._Overview.html [13].

Where the devices do not understand IS-05, the MOC Gateway makes any necessary device control changes for the connection, e.g. to enable the output of a camera, or enable an input of a video monitor.

The Sony open-source NMOS control client (nmos-js) provides basic connection control functionality. 5G-RECORDS is extending this for its use cases.

Where a connection is made between networks, as in Use Case 2, the MOC Gateway and Media Gateway work together to handle any format and address conversion required, in a way that their presence is hidden to devices and other systems in the external network (a design goal of the MOC Gateway).

Figure 87 illustrates such an example for a connection to an ST 2110 media network, with programme video from a camera and a return video to a monitor.

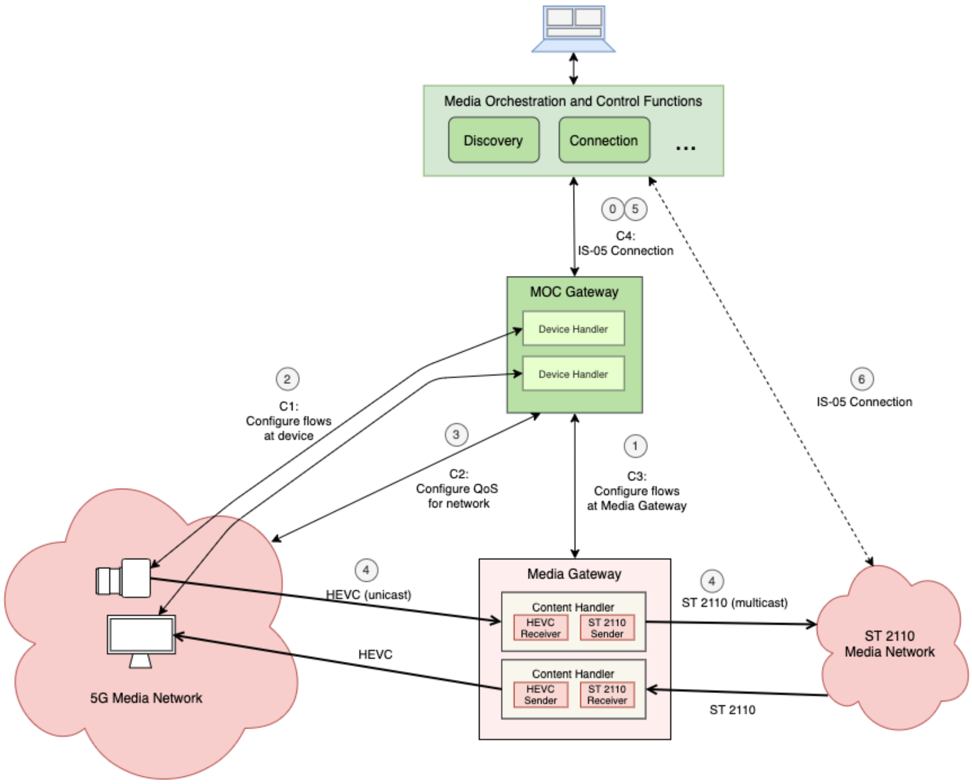


Figure 87: Media Device Connection

In this example, to connect the camera's video output:

(0) A control application discovers the camera by querying the IS-04 registry. This provides a link to a proxy for the camera's sender provided by the MOC Gateway (see discussion of proxy above – the idea is that the control application can treat the sender as though it were on the ST 2110 network). It uses the link to request a connection through IS-05 stage and activation messages.

(1) The MOC Gateway's Device Handler for the camera asks the Media Gateway to provide a Content Handler for the video connection. It provides the ST 2110 sender configuration, including multicast IP address and port information, and information about the protocol and format used on the 5G network. The Media Gateway returns the receiver unicast IP address and UDP port(s) (for RTP streams, a second port may be required for RTCP).

(2) The Device Handler provides the camera with the sender configuration, including codec configuration and the IP address and UDP port(s) on the Media Gateway. The camera returns its IP address and sender port.

(3) The Device Handler now has information it needs to configure the QoS flows withing the 5G system. This is discussed in section 6.3.3.

(4) The camera streams (unicast) to the Media Gateway, which in turn streams (multicast) onto the ST 2110 network.

(5) The Device Handler sends the control application information about the multicast stream's codec, address, port etc. This is an IS-05 "transport file" containing SDP information.

(6) The control application uses this to connect the receiver(s) in the ST 2110 Media Network.

Figure 88 is a sequence diagram showing this in more detail, including creation of a Content Handler:

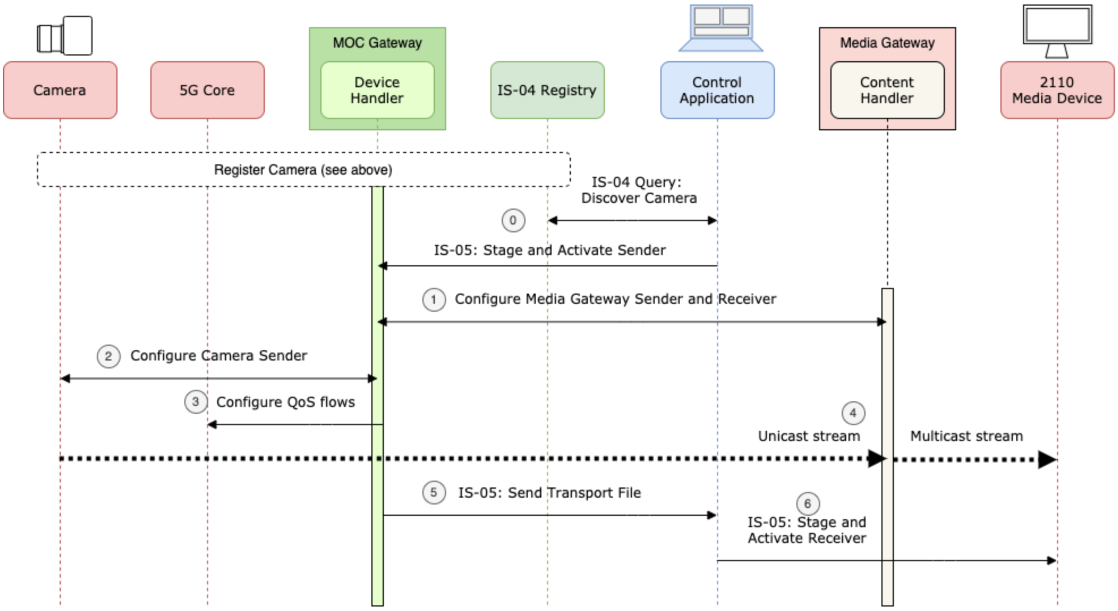


Figure 88: Connection Sequence

The above only shows the video connection but a similar sequence applies to audio and other connections from the camera.

To connect "reverse feeds" such as video to the camera viewfinder or monitor in the 5G network, the configuration is reversed, so:

- The IS-05 *sender* is in the ST 2110 network the *receiver* is in the Media Gateway.
- The Device Handler gets the *receiver* address and port information from the camera and configures the Media Gateway's *sender* to use this.
- The Device Handler configures the ST 2110 multicast *receiver* address and port on the Media Gateway,

In our example the connection happens immediately; IS-05 also supports delayed activation of connections to allow several to be made at the same time (this is sometimes called "salvo" connection). For audio connections, channel mapping operations may also be included using IS-08. Once the IS-05 connections are established, the MOC can interact with 5G Core to request needed QoS support.

6.3.3 Network Control and QoS

Where QoS and/or network slicing are required to prioritise the media flows in the user plane, the MOC Gateway will invoke appropriate APIs in the 5G control plane. In 3GPP's 5G terminology, the MOC acts as an AF (Application Function) to activate network services or a QoS Flow. In some cases, the MOC Gateway may directly interact with the PCF (Policy Control Function). In other cases, the MOC Gateway interacts with the PCF though the NEF (Network Exposure Function).

The 3GPP PCF and NEF expose HTTP REST APIs and are described in 4.5.6 in more detail.

Figure 89 extends the previous example (Figure 87) to show a simplified view on the 3GPP QoS Model including Network Slices. Step (3) of the previous sequence can now be written as follows:

> (3) The MOC Gateway now has all the information required to configure the QoS flows within the 5G System to provide QoS separations. For each flow, the MOC Gateway provides a 5-tuple with sender and receiver IP addresses and ports, and the protocol. The MOC Gateway can then request a specific QoS treatment. The UPF and the UE map the traffic flows using lower layer mechanisms.

Note, as well as the 5-tuples, the 5G System may use other criteria for traffic detection, for example a Type of Service value or a flow label (IPv6 only).

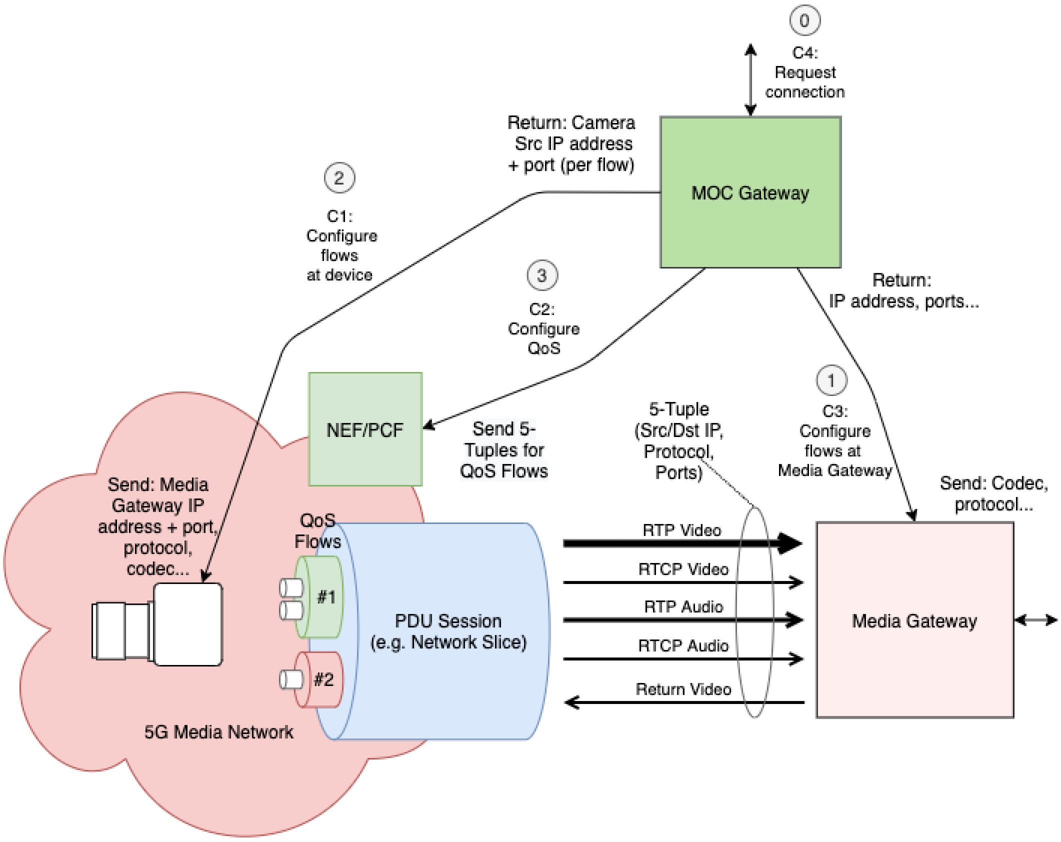


Figure 89: Simplified 3GPP QoS Model

6.3.4 Configuration, Control and Monitoring of Media Devices

At present, the NMOS specifications do not implement general camera control, and there are many protocols in use in the broadcast industry, from simple serial port protocols to API frameworks. It is unrealistic to expect an open and fully standardised approach to be used for everything, but the project will demonstrate a common approach where appropriate.

6.3.4.1 *Camera Control*

An important aspect of the 5G-RECORDS project is the control of remote devices during operation from a different location (e.g., a main studio, a smart-working studio) connected through a 5G network. With the aim to find a solution as universal, extensible and device-agnostic as possible for this task, 5G-RECORDS is defining a novel way of exchanging a core set of camera operational controls commonly used in live production environments, possibly integrating consolidated technologies in use today.

In fact, one of the main requirements for this novel approach is that the camera control messages exchanged with this protocol will be designed to be consistent with and reuse as much as possible the NMOS messaging format, the current state-of-the-art for IP-based productions. Unfortunately, NMOS specifications and APIs (including IS-07), do not currently specify camera controls. AMWA has an ongoing activity to define models for camera device control and 5G-RECORDS is in liaison with AMWA to find an appropriate technical solution for this.

NMOS IS-07 currently supports MQTT and WebSockets for transport of event flows, which can be used to carry control information. Based and these considerations, we adopted a lightweight protocol created for IoT projects: MQTT.

MQTT has a lot of advantages:

- The protocol is lightweight enough to allow embedded devices to parse and respond quickly.
- It is flexible enough to support diversification of devices and cameras.
- It is designed as an asynchronous message protocol instead of the asynchronous protocol. This could be useful from the network point of view; latency could be unstable.
- Usage: simple to use and supported with a lot of libraries available.
- Already deployed at Ericsson labs and used by Cyanview products.

To evaluate of the most suitable MQTT broker for the project, we started with a performance comparison among three main solutions on the market conducted in [14] by confluent.io. In this comparison, their choice fell upon Kafka, Pulsar and RabbitMQ and their test focused primarily on system throughput and latency, since these are the main performance metrics for event streaming systems in a production environment. To be more specific, the throughput test measures how efficient is each system in utilizing the hardware, specifically the disks and the CPU. The latency test, on the other hand, measures how close each system is to delivering real-time messaging including tail latencies of up to p99.9th percentile, a key requirement for real-time and mission-critical applications as well as microservices architectures.

The performances (reached on a Virtual machine with 8 vCores, 64 GB RAM, 2 x 2,500 GB NVMe SSD, and with a Network card with high 25 Gbps network bandwidth) are listed in Table 31.

Table 31: performances comparison

	Kafka	Pulsar	RabbitMQ (Mirrored)
Peak Throughput (MB/s)	605 MB/s	305 MB/s	38 MB/s
p99 Latency (ms)	5 ms (200 MB/s load)	25 ms (200 MB/s load)	1 ms (reduced 30 MB/s load)

The figures in the table, clearly show how, in a broadcast environment with limited number of cameras and remote-control panels (RCPs), all the brokers tested are able to fulfil our target performances with a good margin. From these results, and from similar tests conducted in [15], [16] and [17], we assumed that the choice of the MQTT broker for the project should not be based solely on throughput and latency but should involve also other parameters, such as, for example:

- Opensource code,
- Availability of a C/C++ implementation for easy integration with other software developed in the project,
- Ease of use/configuration,
- Adoption by developers' community,
- Previous experience by the project's members.

An excellent candidate, that fits perfectly all the previous requirements, is Apache Mosquitto and, therefore, it was chose as the reference MQTT broker for the project.

Regarding camera control, after a preliminary study phase, a few main controls and functional requirements were identified and grouped into two main classes to allow a more organized planning of the development phase:

- Basic:
  - Shutter and iris
  - White Balance
  - Pan, tilt and zoom (PTZ)
  - Focus
- Advanced:
  - Saving pre-set
  - Smart tracking

Those messages will be sent from the controller (typically a remote control panel) to the camera, and vice versa, as shown in Figure 90.

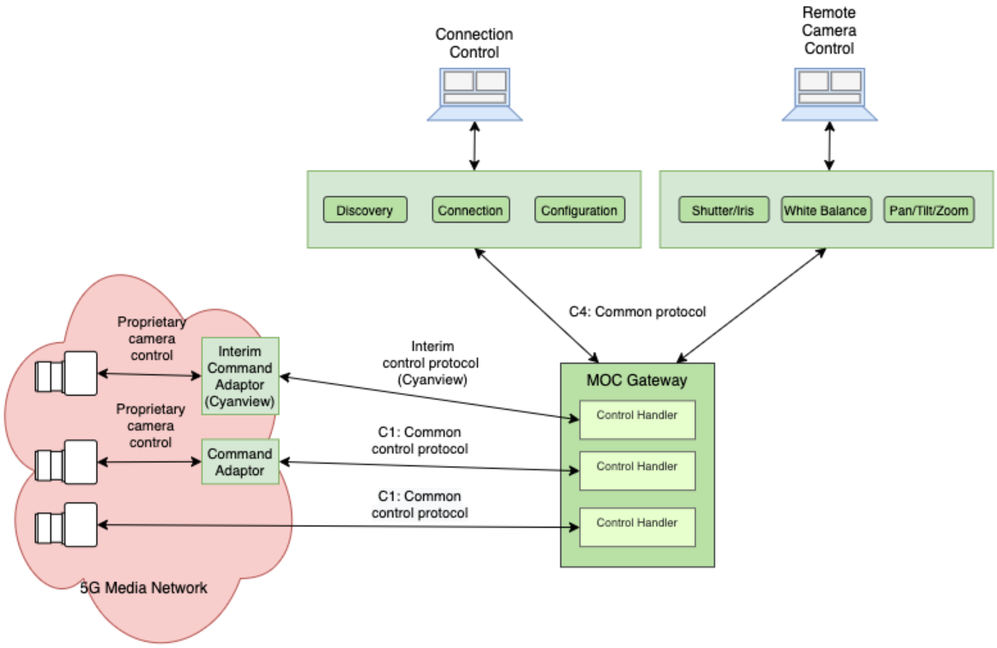


Figure 90: Camera Control

In a long-term vision, the cameras support a common set of control commands, using a common protocol (illustrated in the lowest camera in Figure 90). The MOC Gateway does not need to reformat any of the commands. The middle camera in Figure 90 illustrates the need of a Command Adaptor, which is attached to a camera to convert to proprietary camera control commands.

The upper camera illustrates an interim solution, based on the Cyanview RIO proprietary command adaptor and its corresponding protocol. The interim solution will be tested standalone (using Cyanview's control client) at the May 2021 Aachen testing (initially without a MOC Gateway).

In this schema, the MOC Gateway is the most important part and is in charge of acting as a proxy and potentially as translator for the specific communication required for the cameras.

The MOC Gateway deals with the specific communication required for the cameras. Until all relevant cameras support a common operational control protocol and some common core control messages (lower camera in Figure 90), this will require a command adaptor which converts the common operational control messages to camera vendor specific command sets (middle and upper camera in Figure 90).

Cyanview uses an MQTT-based protocol to carry operational control messages. MQTT is a lightweight broker-based publish-and-subscribe protocol, which supports handling of arbitrary size messages. The MQTT broker does not process the message itself, thus, a message can be text or any binary formatted message. The pub/sub model allows a single device to send to multiple devices with a single message. The MQTT Broker use a so called "topic" to filter and forward messages to subscribers. A publisher sends a message to a specific topic. Interested devices need to subscribe with the MQTT broker on a per- MQTT topic bases in order to receive messages. This is because a device sends its message to a broker, which then distributes the data to the subscribers on the so called "topic". The MQTT client library candidate would be the Paho C++ from Eclipse Foundation.

Cyanview has defined a specific MQTT topic hierarchy for the operational control purpose. For example, to control a camera tally, a numeric value is published to an MQTT topic such as the following (simplified) example:

```
/<id>/camhead/set/tally
```

where `<id>` uniquely identifies of the camera.

Figure 91 shows a simplified control sequence using this approach, where the MOC Gateway converts a generic operational control command set into Cyanview message format (e.g., using MQTT) and then sends Cyanview formatted message using MQTT to the Cyanview Command Adapter.

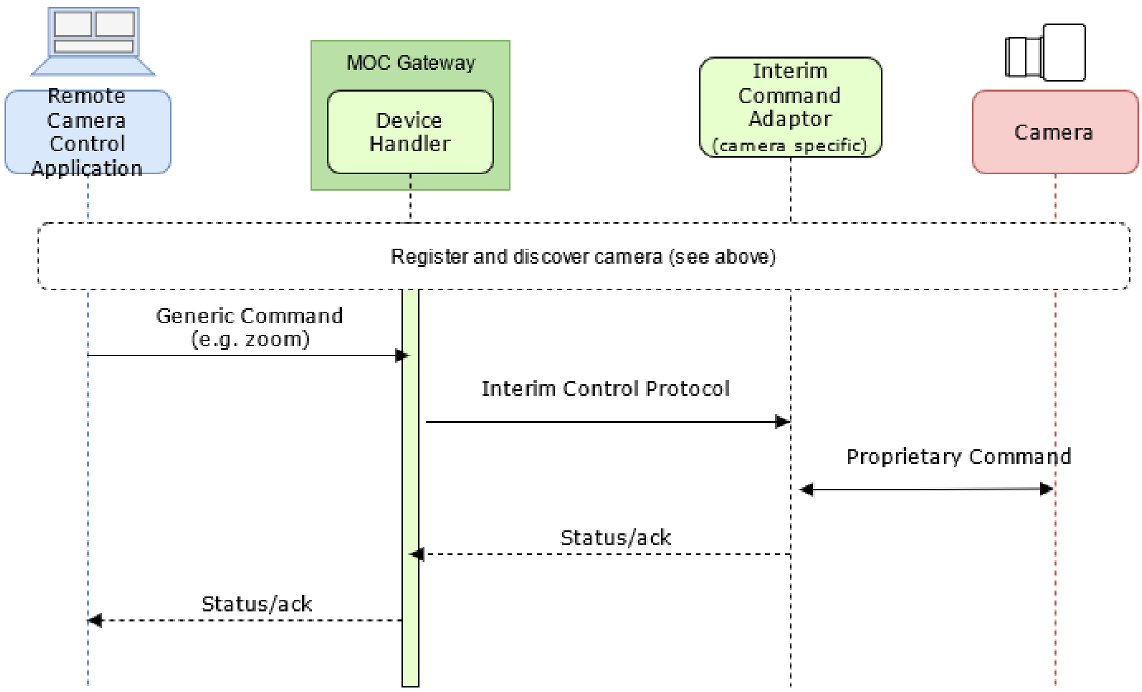


Figure 91: Example camera control sequence

During the analysis of the operational control ecosystem, a wide number of operational control protocol variants were identified. In this context three main scenarios were identified. The shared base of each control architecture is the Sony nmos-cpp open-source code.

1. *Hybrid Nmos-Cyanview*
2. *Nmos IS-07 adaptation based on Sony nmos-cpp code*
3. *New control protocol design based on Sony nmos-coo code*

The *hybrid Nmos-Cyanview* scenario is the least complex case whereby a Cyanview CY-RIO is used as Command Adapter as illustrated in Figure 92. The role of the Command Adapter is to translate the Cyanview MQTT messages into camera vendor proprietary commands. On right side of Figure 92 there is the IS-07-based chain. The role is to pass control messages from the remote virtual RPC Client NMOS node sender to the Control Handler receiver, then the Control Handler can translate these messages into MQTT messages (based on Cyanview's MQTT topic structures) and forward them to the MQTT broker. In this scenario is possible to attach virtually any IS-07 compatible sender to a compatible receiver. The registration with the NMOS registry and the activation of the IS-07 are allowed by using the Nvidia NMOS controller (which uses the Sony nmos-js open-source client code).

One of the most important advantages of this scenario is the reuse of the work performed by the Cyanview team with the proprietary camera controls protocols. A simplified diagram of this architecture is presented in Figure 92.

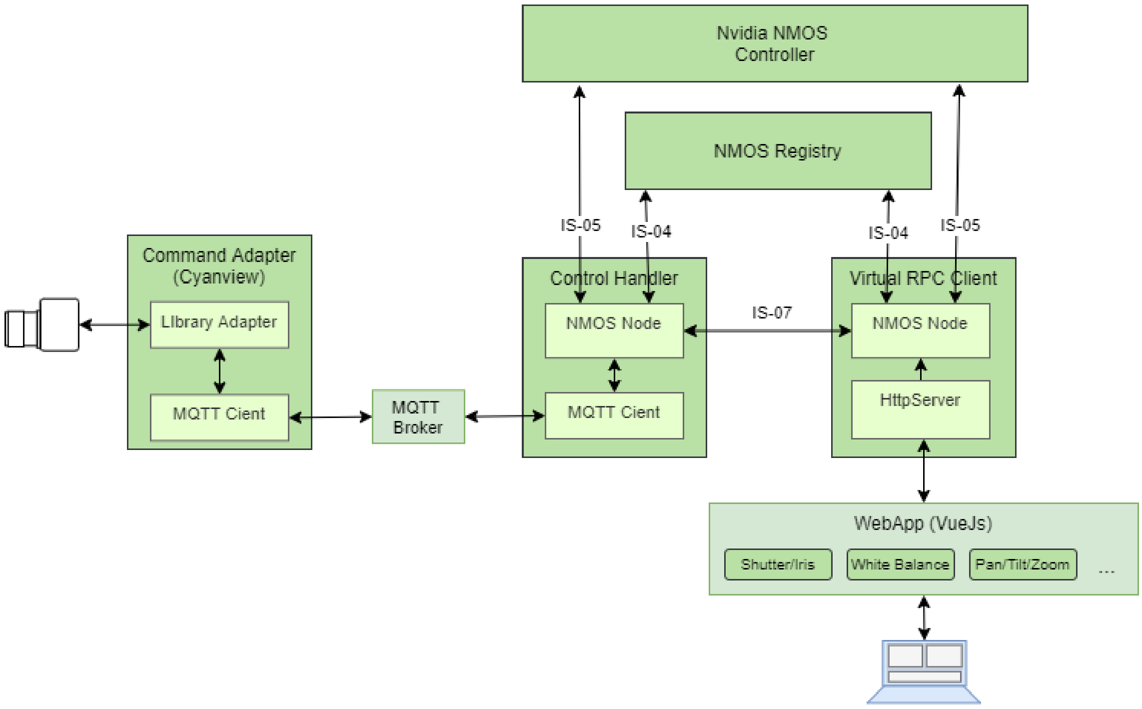


Figure 92: Cyanview scenario

The second scenario under evaluation is *NMOS IS-07 adaptation based on Sony nmos-cpp code.* In this particular scenario the aim is to reimplement the whole chain from the remote WebApp virtual RPC app to the proprietary control protocol. The platform identified for the development of the camera interface is under evaluation, at the moment the best candidate is the Xilinx FPGA programmable platform, that has many advantages and could be a good solution for both camera control and audio/video handling. The NMOS chain in this scenario is based, like in the other case, on IS-07 usage, a NMOS registry, and the nmos-js based Nvidia NMOS controller to activate the data flows fromNMOS sender to the receiver. A simplified diagram of this architecture is presented in Figure 93.

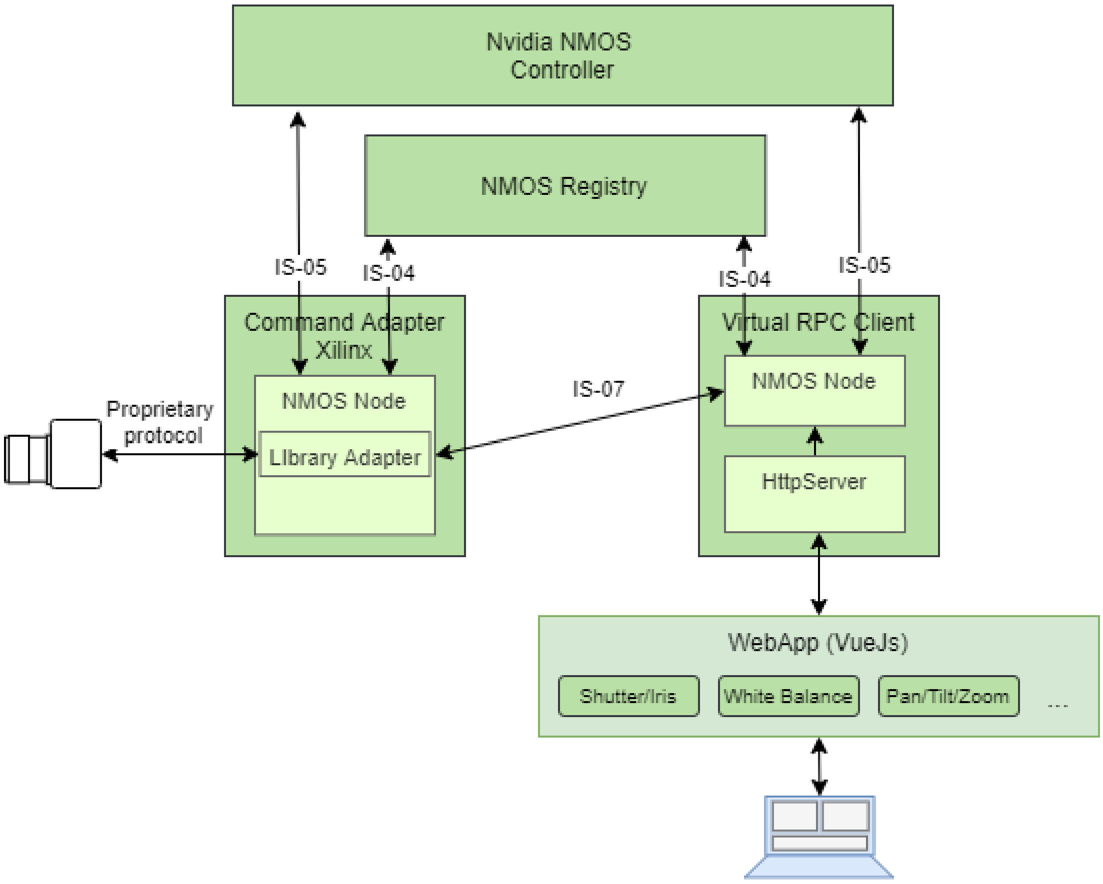


Figure 93: Nmos IS-07 adaptation based on sony-nmos code deploy

The last scenario under evaluation is the *new control protocol design based on Sony nmos-cpp code.* In this scenario an architecture with a new control protocol design could be analysed and developed. The main difference between this last scenario and the previous two is that in this case the Camera Control team would develop a new protocol instead relying on IS-07. In fact, the IS-07 standard does not currently suggest use as a controlling or sending commands protocol.

In this particular camera control architecture in Figure 94, the idea is to rely on WebSockets or Web API to exchange NMOS-compatible JSON messages between a virtual RPC application and the NMOS capable node, Command Adapter (for example a Xilinx FPGA board [18], already used in the project, see 4.5.2). The command adapter device would be both the handler of the control capabilities and the audio/video capabilities; again, with this architecture the best fit is a Xilinx FPGA board. The command adapter is the responsible component of the proprietary control translation, and the type of protocol depends on camera brand.

The project is considering approaches based on NMOS because no NMOS specification yet provides camera control, and AMWA are currently modelling device control; therefore 5G-RECORDS is well-placed to influence ongoing specification work. Although device control is not in the current scope of IS-07, its ability to carry time-stamped messages could be suitable for scenarios where a varying control parameter value needs to be sent accurately with respect to a time base, such as a time-dependent camera pan/tilt move. In other scenarios this level of detail may not be needed, and the simpler approach of the last scenario may be more applicable.

When the command adapter comes up and running register itself on the NMOS registry with a new type of capability defined for camera control use case:


```
{
    "senders": [
        "43e476ba-01eb-4ee8-b56d-e0f2ba4812a4"
    ],
    "receivers": [],
    "controls": [
        {
            "type": "urn:x-mos:control:sr-ctrl/v1.1",
            "href": "http://hostname/x-nmos/connection/v1.1"
        },
        {
            "type": "urn:x-mos:control:sr-ctrl/v1.0",
            "href": "http://hostname/x-nmos/connection/v1.0"
        }
    ],
    "tags": {},
    "type": "urn:x-mos:device:generic",
    "label": "Camera Adapter Xilinx",
    "node_id": "b1e6fde5-fc17-455f-8f4a-e7af388f51e4",
    "id": "31a6da29-105a-430d-9f68-ac49534bc429",
    "description": "Camera Adapter Xilinx"
}
```


The WebSocket endpoint on the command adapter is responsible for offering the control aspects mentioned above, including:

- PTS
- Shutter/Iris
- White balance

At the other endpoint of the WebSocket, the virtual RPC WebApp application would send a flow to find and connect to all the compatible and controllable NMOS nodes (using IS-04 Query API as discussed previously). It could be feasible to control more than one remote command adapter, depending on protocol design and boundaries. The IS-04 Query API provides filtering mechanisms to aid client find the device. Figure 94 shows a simplified diagram of this architecture.

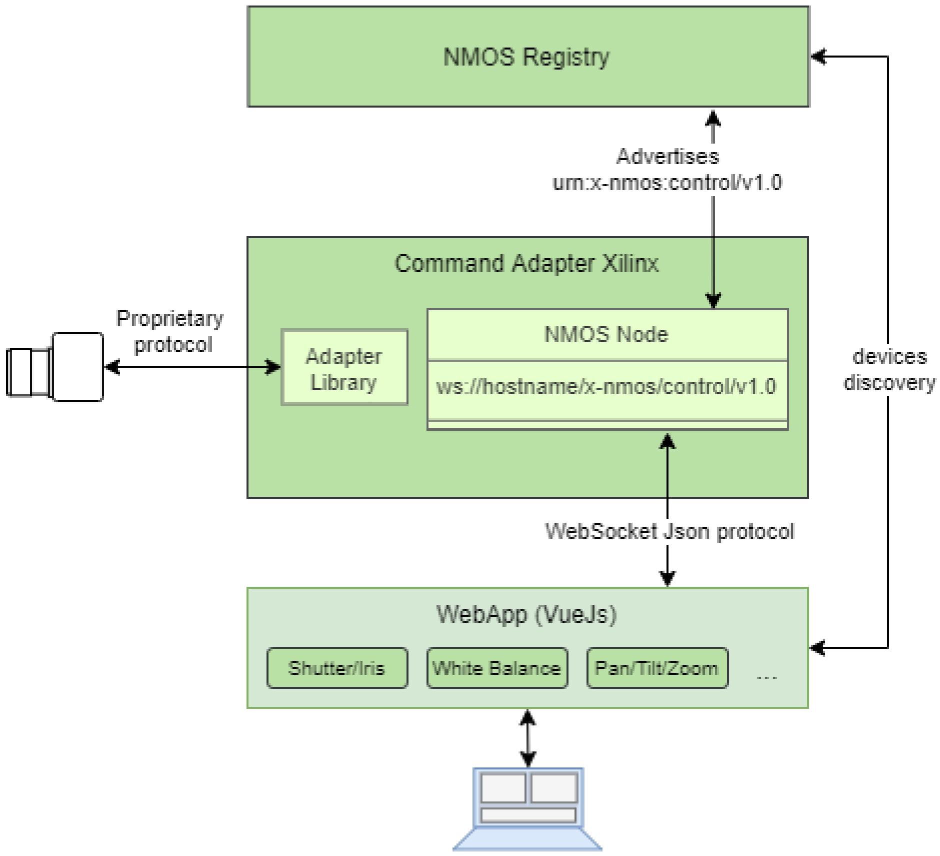


Figure 94: new control protocol design based on sony-nmos code deploy

6.3.5 Timing and Synchronisation

Some operations, such as pan/tilt/zoom require low latency for responsive operational control. 5G-RECORDS will investigate the choice of protocol and topology on this latency. For example, MQTT uses a message broker rather than a direct point-to-point approach, and it may be necessary to have this located locally or at the "edge" rather than in a remote cloud (i.e. ensuring short path message routing).

Control messages should include a timestamp of when they were issued, and optionally when they are to take effect (if not immediately). This helps with monitoring control latency, as well as scheduled control. This approach has been adopted by AMWA IS-07, whose messages can contain creation, origin and action timestamps [19], and IS-05 allows connections to be activated at a specified TAI time

6.3.6 Authentication and Authorisation

A media device's modem is authorised on the 5G network as described in section 6.3.2. A further layer of authorisation can also be provided in the MOC Layer, to control access to production resources. Examples include:

- Only allowing known media devices to be registered.
- Only allowing authorised users to connect devices.
- Only allowing authorised users to control devices.

This will build on the AMWA IS-10 Authorisation API provides a mechanism to control access to NMOS API calls, including IS-04 Registration and Query and IS-05 Connection

APIs. IS-10 uses the OAuth 2.0 open authorisation protocol (https://oauth.net/2/) [20], that is widely used for authorising web services, with JWT access tokens. Further work will address authorisation that is beyond the current scope of NMOS specifications, such as camera control.

6.3.7 Provisioning, Resource Management and Monitoring

The ability to provision cloud resources on demand increasingly important part of production operations. To support this:

- The MOC Gateway dynamically creates Device Handlers when media devices become available
- The Media Gateway dynamically creates Content Handlers as required
- Registration of a sending/receiving device can trigger provisioning of a corresponding stream receiver/sender in the cloud, along with any media services needed to process content

NMOS services can be deployed and automated using container technology (Docker and Docker Compose) and this approach will be used for the MOC Gateway, together with continuous test and integration methods, and appropriate status logging.

6.4 Interfaces

The MOC Gateway functionality as described in Chapter 6 includes four main interfaces used by its Device Handlers. The section numbers referenced below provide more information.

6.4.1 C1: Device Interface

C1 is an interface with media devices in the 5G network. It provides the following functionality:

- Registration of the media device (6.3.1),
- Configuration and connection of media flows from/to the media device (6.3.2),
- Configuration, control and monitoring of the media device (6.3.3, 6.3.4).

In practice cameras and other media devices use various approaches to this interface, including proprietary protocols. 5G-RECORDS is investigating the most appropriate protocol(s) and messages for possible definition of a core control set. This may build on work happening in AMWA on device modelling and how that can be mapped into a NMOS-compatible API (see Annex E).

6.4.2 C2: Network Interface

C2 is an interface with the NEF or PCF of the 5G network to provide QoS and/or network slicing functionality. It includes a 5-tuple for each stream flow (sender and receiver addresses and ports, and protocol). (6.3.3).

6.4.3 C3: Gateway Interface

C3 is an interface with the MG to create Content Handlers and configure senders and receivers on the 5G and ST 2110 networks (6.3.2). It will be implemented using RESTful API and publish-subscribe messaging.

6.4.4 C4: Northbound Interface

C4 represents a set of interfaces with production control plane applications and services, including:

- Registration of the 5G media devices and MG using AMWA IS-04 (6.3.1).
- Handling AMWA IS-05 connection requests from control applications (6.3.2).
- Handling device control messages from control applications (6.3.4).

Annex A: 5G Network

Annex A presents the main features that define the 5G core network. Given that these characteristics may apply to the three use cases, Annex A aims to provide a general description of the following features: (i) 5G service-based architecture, (ii) standalone and non-standalone architectures, (iii) advanced QoS, (iv) network slicing, (v) non-public networks, (vi) edge cloud, and (vii) network exposure.

A.1 5G service-based architecture (SBA)

For the 5G system to offer such great flexibility and diversity in use cases, it had to adopt the modern paradigms used in today's cloud offerings. The 5G architecture transitioned the telecom infrastructure from running on fixed hardware components to running as a set of cloud-native services. It was possible using Virtual network functions (VNF) and software-defined networks (SDN) that results in what is known as service-based architecture.

The 5G network architecture allows the separation between User-plane and Control-plane functions, which results in high flexibility in the network design and function distribution. It is possible to choose an architecture where some user-plane functions are deployed closer to the UE, which results in lower latency and bandwidth offload from the network backhaul.

The usage of network functions in the architecture design allows the functions to communicate directly, which simplifies the design and reduces latency. The NFs can operate in a stateless mode, where the NF computation is separated from the storage. Figure 95 depicts the 5G system architecture.

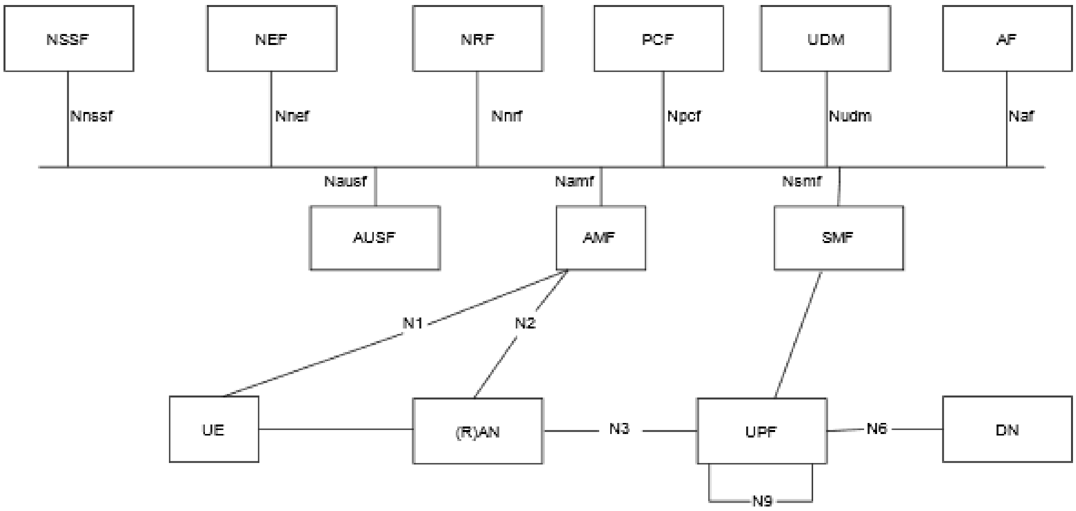


Figure 95: 5G system architecture (Ref.TS.23.501)

Table 32 below is a definition for the 5G network functions:

Table 32: 5G Network Functions

Function		Description
AUSF	Authentication Server Function	Implements the authentication server for the 5G network.
AMF	Access and Mobility Management Function	It provides the interfaces towards the RAN and handles connection and mobility management tasks.
DN	Data Network	Internet, an operator services network or a third-party network connected to the 5G network. It can be either trusted or untrusted.
UDSF	Unstructured Data Storage Function	It stores dynamic data within the network which enables stateless architecture.
NEF	Network Exposure Function	It exposes network capabilities and services to third-party applications via RESTFUL APIs.
NRF	Network Repository Function	It allows other network functions to register their capabilities and discover other functions within the network.
NSSF	Network Slice Selection Function	It assists slice selection: it provides the network slice instance to the UE based on its authorization.
PCF	Policy Control Function	Define the policy for the control plane and enforce it in the user plane.
SMF	Session Management Function	It keeps track of the ongoing sessions for a user. It is responsible for session management (establishment, modification, and release).
UDM	Unified Data Management	It generates Access and key agreement credentials; it also authorizes UE access to services and manages its subscription. It keeps the subscriber profiles.
UDR	Unified Data Repository	It is a database for different subscription information such as subscription data, policy data, exposure data, and application data.
UPF	User Plane Function	It is responsible for initiating a user plane connection between the UE and DN. It also routes packets between both ends and enforces policy on the initiated session.
AF	Application Function	Any third-party application that implements the RESTFUL APIs to interact with the 5GC network via NEF.
UE	User Equipment	Client with 5G support. It connects to the network via the radio interface.
(R)AN	(Radio) Access Network	It is responsible for connecting the UE to the 5G core via the radio interface. It has a separate set of functions that interacts with the core via AMF in the control plane and UPF in the user plane.

5G reference points

Table 33: 5G reference points

Reference point	Description
Uu	The air interface between the UE and the 5G RAN
N1	It is non-access stratum NAS layer signalling interface that connects the UE to the 5GC via AMF
N2	It is a control plane interface between the 5GC and the AN. It connects the RAN to 5GC via AMF
N3	It is a user plane interface that transfers user data from the RAN to the UPF.
N6	It transfers user data from the UPF to external data networks such as cloud data center.
N9	It connects multiple UPFs
N4	It connects the SMF to the UPF. It transfers information about packet handling, forwarding and usage report from SMF to the UPF

A.2 Standalone (SA) and non-standalone (NSA) architectures

5G network allows backward compatibility with the LTE network. The current architecture allows the operators that did not upgrade their whole infrastructure to 5GC to benefit from using the NR band. 5G enables such backward compatibility by using two modes of network operation: Stand-alone (SA) and Non-Stand-alone (NSA).

Non-Standalone architecture

The 5G non-stand-alone architecture allows the usage of 5G Access Network (AN) to be used in combination with LTE AN and EPC without deploying the full 5GC. Figure 96 depicts a simplified NSA architecture; the network operator can offer connectivity using the 5G-NR spectrum via EPC, allowing access to higher bandwidth and low latency, while the e-NB carries out the signalling. This architecture limits the usage of the full 5G functionality to only the functions available in EPC.

Standalone architecture

The stand-alone architecture enables the full functionality of the 5G network. The 5GNR and the AN can carry both the UP and CP without the need for LTE. The 5GC is deployed with full functionality, allowing features like virtual end-to-end slicing and NEF. The 5GNR can also operate in LTE bands, so operators that deploy 5G networks can still benefit from the existing LTE bands.

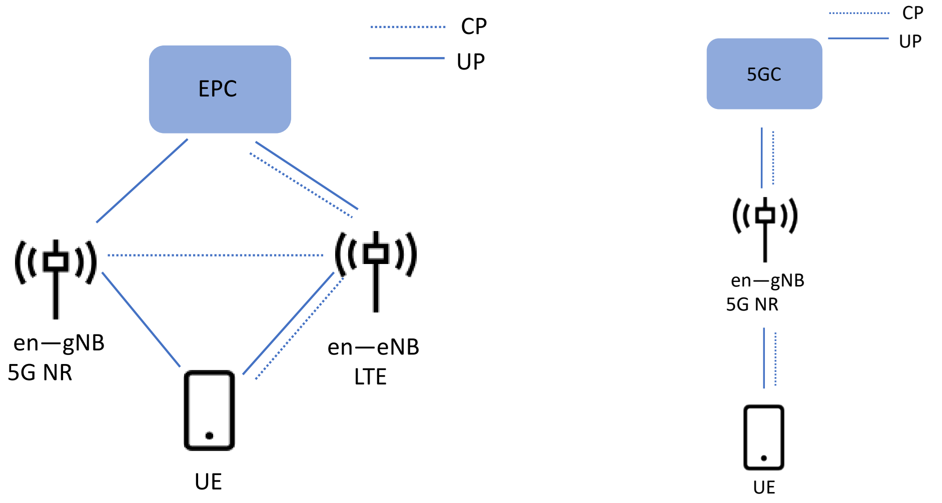


Figure 96: (a) 5G NR NSA architecture, (b) 5G NR SA architecture.

A.3 Quality of Service (QoS)

5G has inherited the concept of QoS from LTE with few modifications to fit the new 5G architecture. QoS allows the core and radio network to tailor the network resources to provide a more deterministic quality for a specific stream for the UE. Specific IP packets can be marked with a QoS flow identifier (QFI) to allow QoS handling. Each QoS is then mapped to a specific data radio bearer at the AN as shown in Figure 97. In the core network, different network based QoS solutions can be used to provide the characteristics. For example, the network can use DiffServ and derive DSCP based on the QFIs.

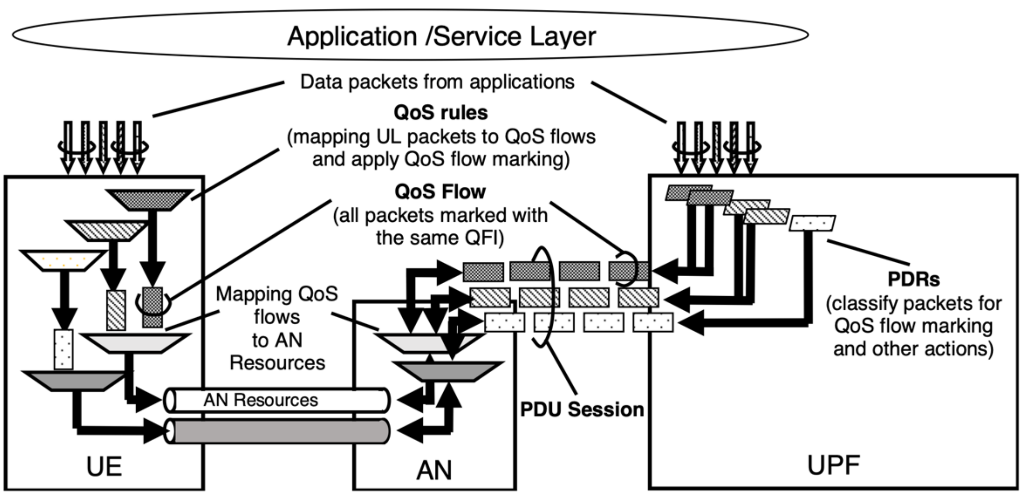


Figure 97: QoS mapping between AN and 5GC [Ref. TS23.501]

The 5G QoS characteristics are associated with a table of network characteristics called 5QI. A set of 5QIs is standardized. The addition of deployment specific 5QIs is supported. The 5QI are categorized according to the following parameters:

1. **Resource type** (GBR, non-GBR): GBR refers to Guaranteed Bitrate. Admission control acts as a gating function to allow QoS bearers up to a supported threshold.

2. **Priority level**: it defines a relative priority between different QoS flows.
3. **Packet delay budget** (PDB): it defines an upper bound for the time that a packet may be delayed between the UE and the UPF. The PDB contains delay contributions for the core network and radio network.
4. **Packet error rate** (PER): it defines an upper bound for the rate of IP packets that have been processed by the sender of a link layer protocol (e.g. RLC in RAN) but that are not successfully delivered by the corresponding receiver to the upper layer (e.g. PDCP in RAN).
5. **Averaging window**: each GBR QoS Flow is associated with an Averaging window. The Averaging window represents the duration over which the GFBR (Guaranteed Flow Bit Rate) and MFBR (Maximum Flow Bit Rate) is calculated in the RAN, UPF and the UE.
6. **Maximum data burst volume** (MDBV): each GBR QoS Flow with Delay-critical resource type is associated with a MDBV. It denotes the largest amount of data that the RAN is required to serve within a period of a PDB.

5QI Value	Resource Type	Default Priority Level	Packet Delay Budget	Packet Error Rate	Default Maximum Data Burst Volume (NOTE 2)	Default Averaging Window	Example Services
1	GBR (NOTE 1)	20	100 ms	10^{-2}	N/A	2000 ms	Conversational Voice
2		40	150 ms	10^{-3}	N/A	2000 ms	Conversational Video (Live Streaming)
3		30	50 ms	10^{-3}	N/A	2000 ms	Real Time Gaming, V2X messages Electricity distribution – medium voltage, Process automation - monitoring
4		50	300 ms	10^{-6}	N/A	2000 ms	Non-Conversational Video (Buffered Streaming)
65		7	75 ms	10^{-2}	N/A	2000 ms	Mission Critical user plane Push To Talk voice (e.g., MCPTT)
66		20	100 ms	10^{-2}	N/A	2000 ms	Non-Mission-Critical user plane Push To Talk voice
67		15	100 ms	10^{-3}	N/A	2000 ms	Mission Critical Video user plane
75		25	50 ms	10^{-2}	N/A	2000 ms	V2X messages
5	Non-GBR (NOTE 1)	10	100 ms	10^{-6}	N/A	N/A	IMS Signalling
6		60	300 ms	10^{-6}	N/A	N/A	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
7		70	100 ms	10^{-3}	N/A	N/A	Voice, Video (Live Streaming) Interactive Gaming
8		80	300 ms	10^{-6}	N/A	N/A	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
9		90					
69		5	60 ms	10^{-6}	N/A	N/A	Mission Critical delay sensitive signalling (e.g., MC-PTT signalling)
70		55	200 ms	10^{-6}	N/A	N/A	Mission Critical Data (e.g. example services are the same as QCI 6/8/9)
79		65	50 ms	10^{-2}	N/A	N/A	V2X messages
80		68	10 ms	10^{-6}	N/A	N/A	Low Latency eMBB applications Augmented Reality
82	Delay Critical GBR	19	10 ms (NOTE 4)	10^{-4}	255 bytes	2000 ms	Discrete Automation (see TS 22.261 [2])
83		22	10 ms (NOTE 4)	10^{-4}	1358 bytes (NOTE 3)	2000 ms	Discrete Automation (see TS 22.261 [2])

Figure 98: 5QI values [21]

A.4 Network slicing

Network slicing enables 5G service providers to create an E2E virtual network that fulfils the requirements for a specific use-case or industry sector. Despite sharing the same physical infrastructure and resources, the Mobile Network Operator (MNO) can separate the slice traffic and resources from the other traffic in the network. The virtualization also allows the MNO to deploy network functions for the dedicated slice when needed. The Service-Level Agreement (SLA) between the MNO and the enterprise governs the resources assigned to each slice.

Figure 99 depicts end-to-end network slicing. Each slice can be divided into RAN, transport, and 5G Core. RAN slicing allows flexibility in resource allocation and prioritization across multiple slices so that it can fulfil the SLA. Transport slicing ensures that the MNO infrastructure has sufficient capacity to transfer data from multiple RAN slices to the 5G Core. It is possible for multiple RAN partitions to use the same physical transport, where traffic is separated via a VPN tunnel. 5G Core enables the MNO to assign dedicated or shared NFs to each slice based on the SLA. Multiple slices can share both user-plane and control-plane network functions. For latency-critical use-cases, It is possible to deploy some user-plane network functions such as UPF on the edge site as shown in Figure 99 in Slice D.

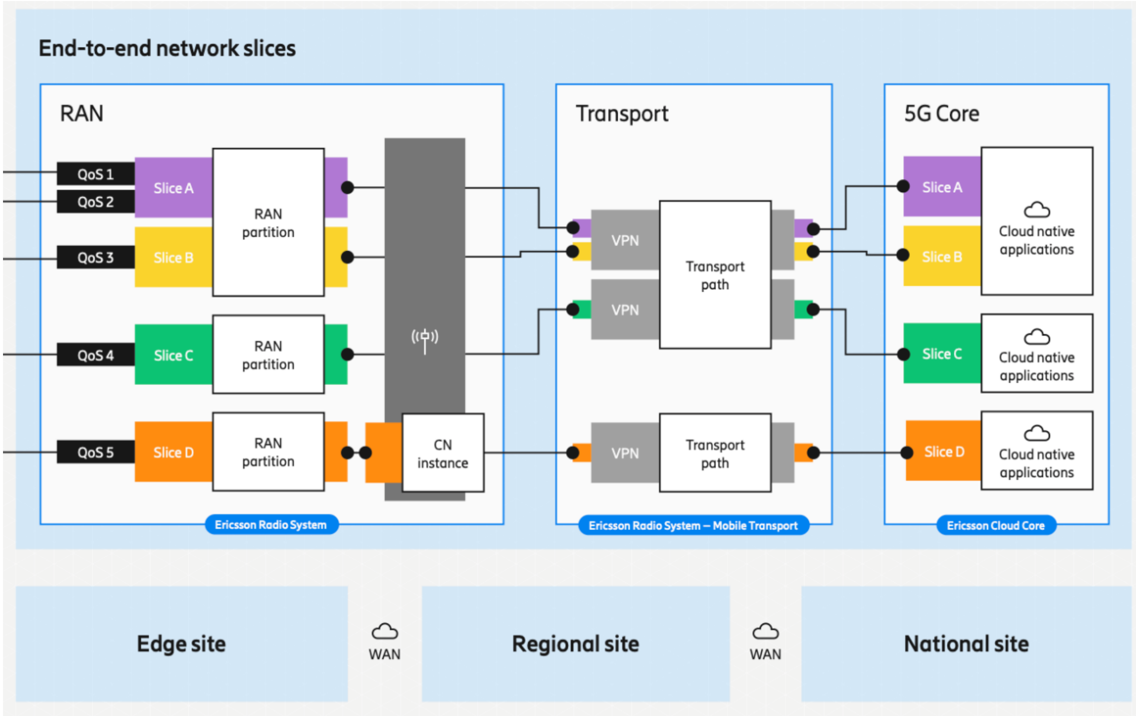


Figure 99: E2E network slicing

A.5 Non-Public Networks (NPN)

One of the fundamental changes in 5G standardization compared to previous generations is Non-Public Networks (NPN). 5G enables different industry verticals to deploy a private 5G network within their premises. The installed network can be logically or physically separated from a Public Network with its infrastructure and resources. Enterprise partners can use NPN to guarantee a specific QoS and execute specific tasks without influence from external factors. It also guarantees a high level of security because part or full network can be hosted within the factory/studio premises.

3GPP categorizes NPNs into two solutions: Standalone NPN (S-NPN) and Public Network-Integrated NPN (PNI-NPN).

Standalone non-public network (SNPN):

Standalone NPN is a full 5G network where the network operator hosts both the user plane and control plane within its premises. The network operator can separate the NPN from any PLMN. Therefore, it has its own NPN Id. The virtualized network functions allow the enterprise to host the user plane functions very close to the operation. This setup is essential in some use cases to reduce latency and increase security. Here, the enterprise is responsible for managing the whole 5G network. SNPN operators can benefit from network slicing and QoS to serve varying traffic within their network. For example, in 5G based media production network, video streams and control signals can be carried on two separate slices, each slice has its unique characteristic to transfer its corresponding traffic. while QoS is used within each slice to prioritize critical traffic as shown in Figure 100

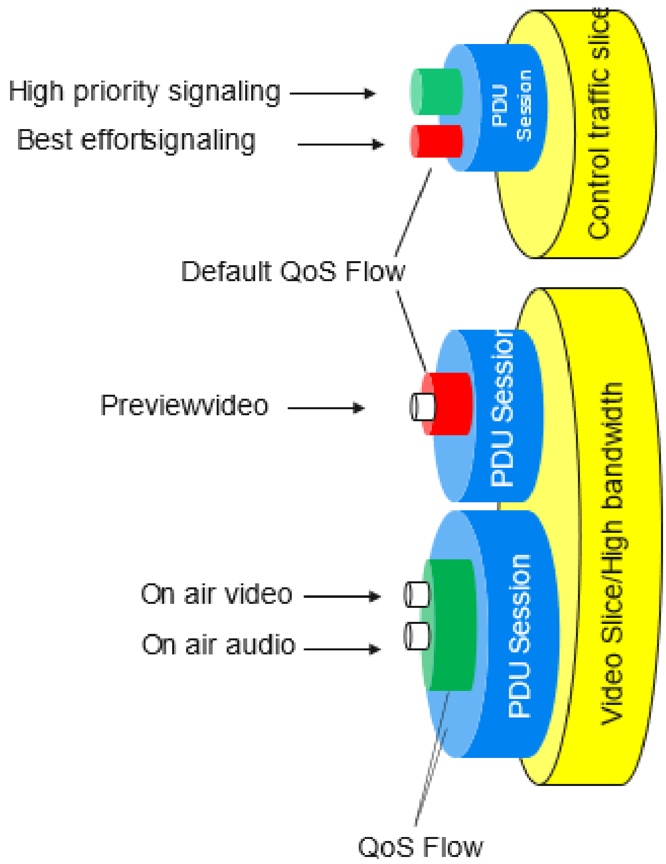


Figure 100: QoS and network slicing within S-NPN

Public network integrated NPN (PNI-NPN):

Public Network Integrated NPN is a private 5G network that shared functionalities with a Public Network. Using VNF architecture, private networks can re-use some functions of the 5G Core or the RAN hosted by the PLMN. One realization for separating the private network related traffic and network resources is network slicing. The MNO hosts and operates the NPN functions outside of the enterprise premises. Other functions can be hosted in the enterprise premises to guarantee the fulfilment of the use case requirements.

A.6 Mobile Edge Computing (MEC)

Mobile Edge computing is a distributed cloud architecture that adapts the location of deploying services according to the use case requirements. It can bring the cloud computational capabilities closer to the end-user. 5G has adopted MEC to enable the integration of the edge within the network. Thanks to the virtualization of the network functions, the edge cloud can be deployed together with the user plane core functions closer to the enterprise. It will reduce latency and give the enterprise more flexibility in designing their solutions and reduce deployment costs. Mobile Edge Computing is often also called Multi-Access Edge Computing.

A.7 Network exposure

The 5G network introduces the Network Exposure Function (NEF) to enable interaction between application functions and the network functions. Application Functions can reside within the mobile network or in external networks. The services that are exposed to the external AF can be categorized into 4 general services:

- **Monitoring capability:** It monitors a specific event for UE in 5GS and expose the available monitoring information for external exposure via the NEF.
- **Provisioning capability**: Enables external party to provision of information which can be used for the UE in 5GS.
- **Policy/Charging capability**: It handles QoS and charging policy for the UE based on the request from external party.
- **Analytics reporting capability**: It allows external party to acquire analytics information generated by 5G System.

A complete list of services, exposed by the NEF can be found in [22]

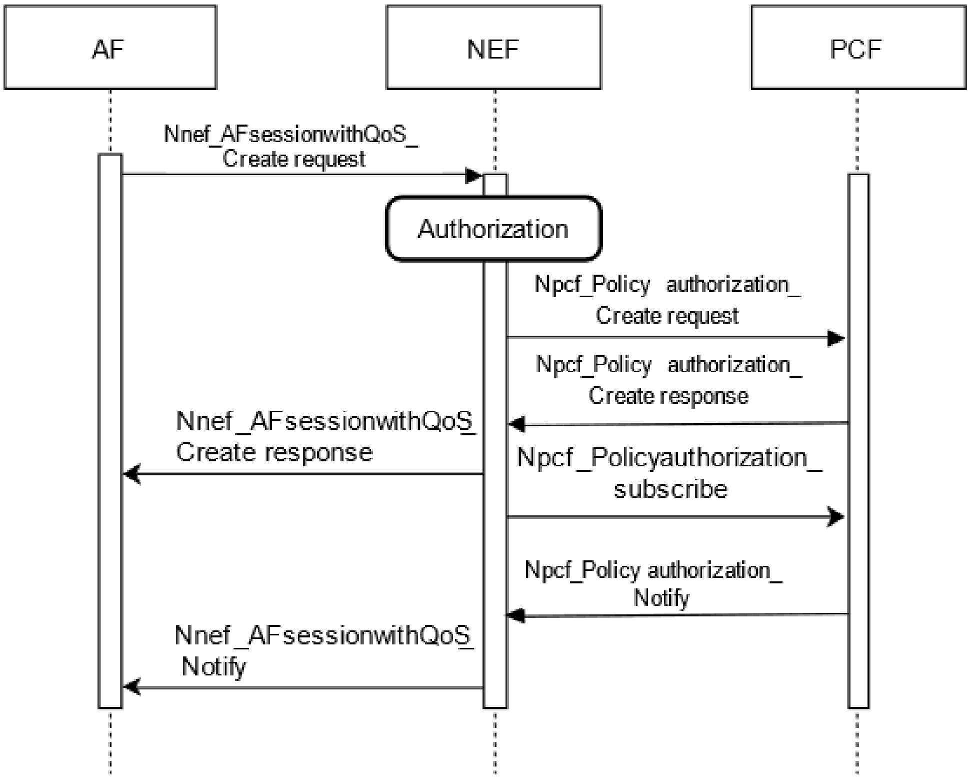


Figure 101: Example NEF procedure for setting up an AF session with QoS

Figure 101 depicts an example call-flow for establishing an AF session with specific QoS parameters:

1. The AF sends a RESTFUL API to the NEF requesting the creation of QoS for an upcoming session. The request is pre-defined as Nnef_AFsessionWithQoS_Create. The request must contain AF identifier, UE IP address, description of application flows and QoS reference. It can also optionally include time period and traffic volume.

2. The NEF authorize the received request and check for the allowed pre-defined QoS for the corresponding AF.
3. If authorization is granted, the NEF sends a Npcf_PolicyAuthorization_Create to the PCF. The request should include all the parameters sent from the request in step 1.
4. The PCF authorize the request and check if the QoS requested is allowed and send the response to the NEF.
5. The NEF sends a Nnef_AFsessionWithQoS_Create response to the AF.
6. The NEF sends a Npcf_PolicyAuthorization_Subscribe message to the PCF to get notified once the resources are allocated to the AF session.
7. Once the resources are allocated, the PCF sends Npcf_PolicyAuthorization_Notify message to the NEF to notify it.
8. The NEF sends Nnef_AFsessionWithQoS_Notify to the AF once it is notified by the PCF.

A.8 5G network radio system

This subsection discusses the radio solution deployed in the Ericsson test network.

Radio dot system

The Ericsson Radio Dot System (RDS) is providing a flexible, cost-effective architecture and superior network performance for indoor deployment. It can be easily installed at confined spaces. The test network is hosted indoors, therefore the usage of RDS for RAN solution is the optimal solution. The RDS has three main components: The Indoor Radio Unit (IRU), the Radio Dots (RD) and the digital unit containing the baseband processing.

The **Radio Dot (RD)** is a radio front-end including the radio antennas and the radio frequency (RF) part. Figure 102 depicts the radio dot used in the 5G infrastructure. The RD generates and receives the radio waves enabling radio coverage. It is connected to an IRU with a dedicated LAN cable. The IRU can manage multiple radio dots.

Figure 102:Ericsson radio dot

The **Indoor Radio Unit (IRU)** acts as an aggregator of signals sent by the Radio Dots and provides the digital interface towards the BBU. Moreover, it provides power to the RDs, radio processing functions, and Analog-to-Digital Conversion (ADC) / Digital-to-Analog Conversion (DAC). It is connected to the RD via a LAN cable and connected to the digital unit via fiber. The IRU is depicted in Figure 103

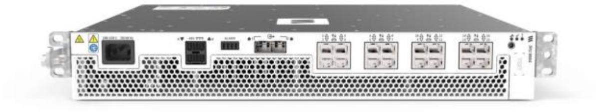


Figure 103: indoor radio unit

The **Baseband Unit (BBU**) is responsible for the baseband processing. The BBU performs radio resource handling, encoding, decoding of uplink and downlink radio signals, radio control signal processing and radio network synchronization.

Annex B: Additional information Edge Cloud.

This annex contains some additional, but relevant information of the Edge Cloud used within UC3.

B.1 Delivery cloud hardware

Hardware elements of Edge Delivery Cloud platform used in UC3 are:

- **Rack. Rack OCP**: One OCP Rack powered at –48DC, is separate in two zones, but with only one central bus bar at 12V.
- **Power**: 2 Power Shelf, one per each zone of the rack, provided by Bel Power. In each power self is installed 3 converters of 3kW to DC/DC 12V
- **Management switch. Celestica Pebble E1050**: Currently we use a Celestica switch for management purposes. It´s an ICOS managed switchi wih 48 RJ45 ports and 4x SFP+ ports. Model is based on Hurricane2, with fixed PSU12V F2B.
- **FTTH access OLT Ruby S1010 vOLT:** 48 GPON ports.
- **Service Switches. SDN Openflow switches**: For SDN service currently are instaled EdgeCore and Smallstone Celestica whitebox switches. 4 x Smallstone Celestica WhiteBox switches, and 2 x EdgeCore Switches. Both based on chipset Broadcom BCM56850 Trident II, and with 32xQSFP 40G ports.
- **Compute.** For compute service and management is installed:
    - **3 Nodes Intel Relion XO1132g Server**
      Intel Skylake OCP Server Relion XO1132g Server
      1OU (1/3rd Width) w/ 2x 2.5" Fixed 6Gb SATA Bay
      Dual Intel Xeon Gold 6132 (14C, 2.6GHz, 140W)
      384GB, RAM, DDR4-2666MHz REG, ECC, 2R (12 x 32GB)
      Integrated AHCI, Intel C621, 6Gb SATA: Linux RAID 0/1/5/6/10/50/60
      480GB SSD, 2.5", 6Gb SATA, 1 DWPD, 3D TLC
      Integrated NIC, Intel X550, 2x RJ-45/10GbE (1-Port Shared with BMC for IPMI)
      NIC, Mellanox ConnectX 3 Pro EN, 2x QSFP+/40GbE, OCP Mezz

    - **4 Nodes AMD Altus XO1132g Server**
      1OU (1/3rd Width), 2x M.2 NVMe Slot w/ 2x 2.5" Fixed SATA Bay
      Dual AMD EPYC 7351 (16C, 2.40GHz,155/170W, DP)
      256GB RAM, DDR4-2666MHz REG, ECC, 2R (16 x 16GB)
      Integrated AHCI, AMD SoC, 6Gb SATA: Linux RAID 0/1/5/6/10/50/60
      480GB SSD, 2.5", 6Gb SATA, 1 DWPD, 3D TLC
      Integrated NIC, Intel I350, 2x RJ-45/GbE (1-Port Shared with BMC for IPMI)

NIC, Mellanox ConnectX 3 Pro EN, 2x QSFP+/40GbE, OCP Mezz

- **Storage.** To diferent options are available for storage:
  - One JBOD. It is an Open Vault (SAS12G JBOD/JBOF) ST7110G2-30A Wiwynn.
  - One StorageServer. It´s a ByceCanyon, up to 72 HDD.

B.2 Edge Cloud Software Components

Edge platform is based on three main modules: (i) compute virtualization Subsystem based on OpenNebula, (ii) storage virtualization subsystem based on GlusterFS, and (iii) network virtualization subsystem based on SDN ONOS.

B.2.1 Cloud Virtualization Subsystem

OpenNebula is an Opensource platform for cloud computing. Is oriented to distributed and heterogeneous datacenters, providing virtual infrastructure to build private, public, and hybrid clouds. The key aspects are as follows:

- Opensource under Apache
- Lightweight, easy to maintain, operate, use and upgrade
- Flexible., can be adapted to the needs of each datacenter
- Production ready
- Ready to use in private and public clouds

Internally manage all elements separating the functions of the different technologies, called subsystems:

- Storage subsystem: Datastores configuration and image management
- Network subsystem: Network configuration and virtual network management
- Host subsystem: Monitoring configuration and host management
- Cluster subsystem: Cluster configuration and management
- Virtualization subsystem: Hypervisor configuration and VM management
- Scheduler: Scheduling and configuration policies
- Users Subsystem: AAA (Authentication, authorization and accounting)

Storage terminology inside OpenNebula:

- **Storage images:** images of the cloud virtual machines, decoupled from VMs. They are classified into several types like: OS, DATABLOCK, CDROM, CONTEXT, Volatile disk (cannot be saved), and Snapshots.
- **Storage Datastores**: containers in which are stored all kind of objects.
  - System. Container for running live images.
  - Default. Container for images.
  - Files. Container for files, usually for configuration of running virtual machines.
- **Images Datastores Types:** technologies supported to configure datastores like Filesystem, Ceph, Gluster, LVM, vmfs and dev.
- **Storage transfer managers:** internally images are transferred or copied using one of those drivers (shared, ssh, fs_lvm, qcow2, ceph, dev).

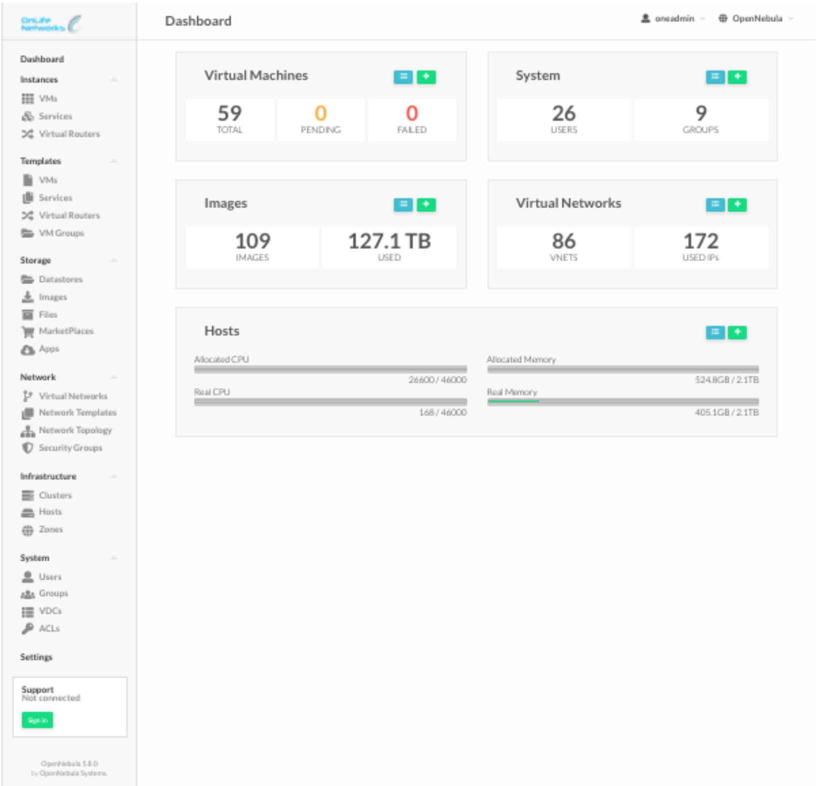


Figure 104: OpenNebula administrator view

Currently for 5G-RECORDS are three datastores reserved:

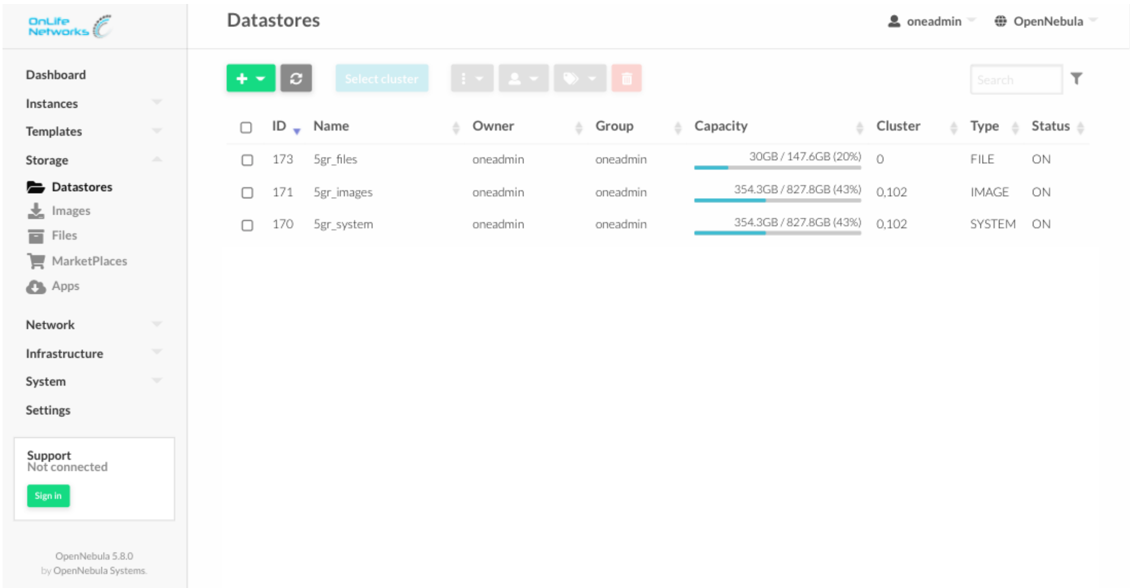


Figure 105: 5G-RECORDS datastores

Network subsystem inside OpenNebula
Regarding network subsystem, it supports these technologies:

- Virtual Networks: bridge, OVS, 802.1Q and VXLAN networks.
- Security group (rules): In/out, Global or protocol, Ports, Source/destination
- Virtual routers

In the Edge Cloud bridges and OVS are defined and managed due an integration between OpenNebula and the SDN controller.

Host Virtualization and scheduling

The hosts are the physical machines that will run the VMs. Currently using KVM hosts.

Scheduler is in charge of the assignment between pending Virtual Machines and known Hosts.

OpenNebula's architecture defines this module as a separate process that can be started independently of oned (OpenNebula daemon). Policy:

POLICY	DESCRIPTION
0	Packing: Minimize the number of hosts in use by packing the VMs in the hosts to reduce VM fragmentation
1	Striping: Maximize resources available for the VMs by spreading the VMs in the hosts
2	Load-aware: Maximize resources available for the VMs by using those nodes with less load
3	Custom: Use a custom RANK
4	Fixed: Hosts will be ranked according to the PRIORITY attribute found in the Host or Cluster template

Users: Users in an OpenNebula installation are classified in four types:

1. Administrators, an admin user belongs to an admin group (oneadmin or otherwise) and can perform manage operations
2. Regular users, that may access most OpenNebula functionality.
3. Public users, only basic functionality (and public interfaces) are open to public users.
4. Service users, a service user account is used by the OpenNebula services (i.e. cloud APIs like EC2 or GUI's like Sunstone) to proxy auth requests.

Quotas: The quota system tracks user and group usage of system resources, and allows the system administrator to set limits on the usage of resources. Datastore, compute, network and storage.

ACLS: The ACL authorization system enables fine-tuning of the allowed operations for any user, or group of users.

Other Features:

- **OneFlow.** Server that takes the requests and manages the Service (multi-tiered application) life cycle.

- **OneGate**: component allows Virtual Machine guests to pull and push VM information from OpenNebula. Users and administrators can use it to gather metrics, detect problems in their applications, and trigger OneFlow elasticity rules from inside the VM.

B.2.2 Storage Subsystem

Edge cloud software define storage is based on **GlusterFS.** Gluster is a scalable, distributed file system that aggregates disk storage resources from multiple servers into a single global namespace. The advantages are as follows:

- Scales to several petabytes,
- Handles thousands of clients,
- POSIX compatible,
- Open Source,
- Uses commodity hardware,
- Can use any filesystem that supports extended attributes,
- Allows optimization for different workloads,
- Accessible using standard protocols like NFS, SMB,
- Provides replication, geo-replication,
- Provides quotas, snapshots, etc.

GlusterFS software is installed and configured into server hosts directly.

Storage network. GlusterFS replication is done using two separate networks. First network managed by SDN. And a second auxiliary network using the management switch. Each network has a VLAN assigned.

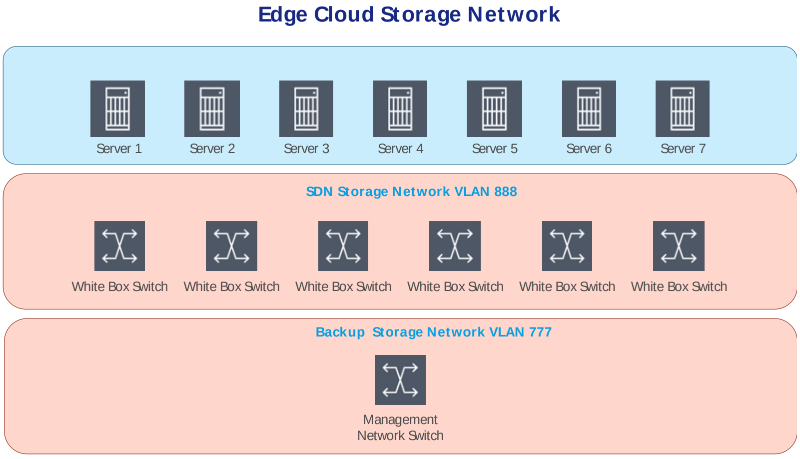


Figure 106: Edge Cloud Storage Network

Gluster configuration is done using at least two cluster for load balance purposes.

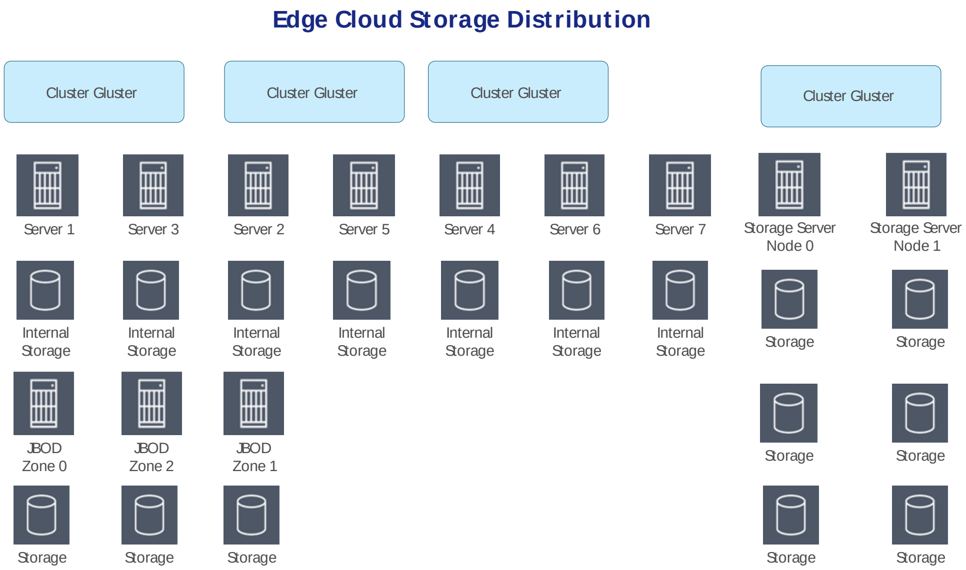


Figure 107: Edge Cloud Storage Distribution

B.2.3 Network connectivity subsystem

Network connectivity subsystem is based on ONOS, Open Networking Operating System, ONOS provides the control plane for a software-defined network (SDN). In the Edge cloud network solution, ONOS is used as the SDN controller, in charge of managing the switch fabric. ONOS provides a series of functionalities and software models to provide communication services to end hosts and neighbouring networks. The following diagram show ONOS subsystems that make up ONOS.

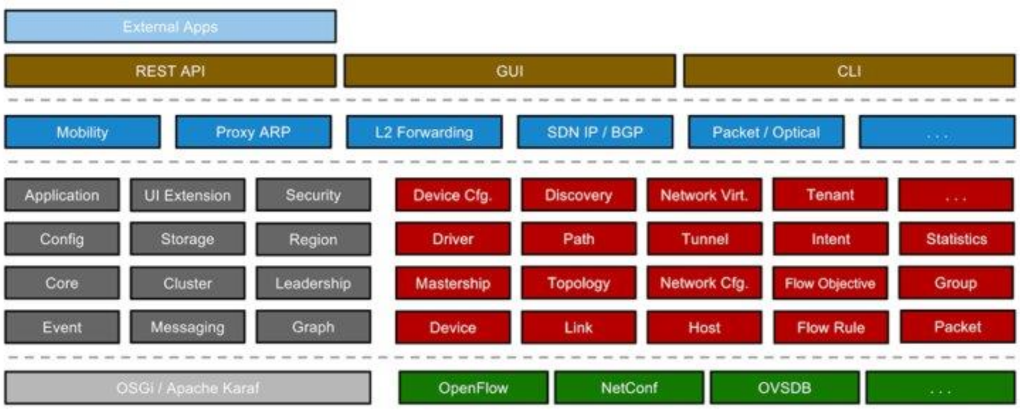


Figure 108: Network Connectivity subsystem based on ONOS

We use a CORD approach with these characteristics:

- Overlay and Trellis (underlay) independent layers,
- Trellis provides infrastructure 'basic' IP connectivity to hosts,
- Service connectivity implemented as VxLAN over Trellis,
- OVS dependent for:
  - Access control,
  - Load balancing.

Our basic rules for design in the SDN are:

- Scale L2 to thousands of users,
- Avoid flow-oriented approach,
- No more one flow to connect A and B,
- Minimize ONOS API actions required to move hosted VMs,
- Keep logic in CLOS Switches (no OVS flows).

The capacity per instance is: CPU: 1vCPU, Memory: 16Gb, Disk: 30Gb.

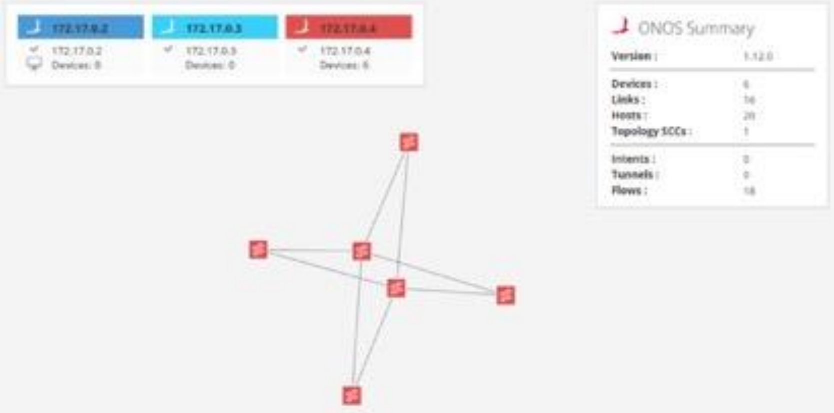


Figure 109: Example after installation

Annex C: FVV capture system calibration

The capture system has been designed to be portable and based on stereo consumer electronics cameras. These cameras are placed at fixed (discrete) points around the scene using variable height stands, aligned to cover the desired field of view, and wired to the capture servers using USB cables. Each capture server manages three cameras and performs the operations described on section 5.2.2: synchronization, RGB + depth data capture, depth post-processing, encoding and transmission.

To correctly render synthetic views, two offline tasks must be performed previously to the actual capture process: (i) a complete cameras calibration and (ii) background modelling. Both operations are performed using custom software developed specifically for this purpose.

(i) Calibration

Once the system is deployed, to correctly synthesize views, a complete and accurate characterization of both the internal and external parameters of the reference cameras is necessary to know where the cameras are placed in a common reference system. The internal parameters of each camera are characterized, using the common pinhole camera model, by the intrinsic matrix:

$$A = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

where f_x and f_y describe the focal length, and c_x and c_y the principal point of the camera (all of them in pixel units). The external parameters of the camera are characterized by the extrinsic matrix, which describes the location of the camera in the 3D world and the direction it is pointing to:

$$(R|t)$$

where R is the 3x3 rotation matrix, and t is the 3x1 translation vector.

This set of parameters is necessary to unproject points from the image plane to the 3D space and then project those 3D points back to the image plane of the virtual camera.

To obtain the described parameters, a reference object/pattern calibration technique has been chosen for accuracy. The calibration process is performed by placing a pattern with known geometry in front of the camera. Different 3D or planar patterns can be used to obtain matching points between the images captured by the cameras. For this specific case, a checkerboard (Figure 110) has been chosen as the calibration pattern to ease point detection and obtain multiple matches between a large set of captured images, hence leading to the creation of a dense 3D point cloud.

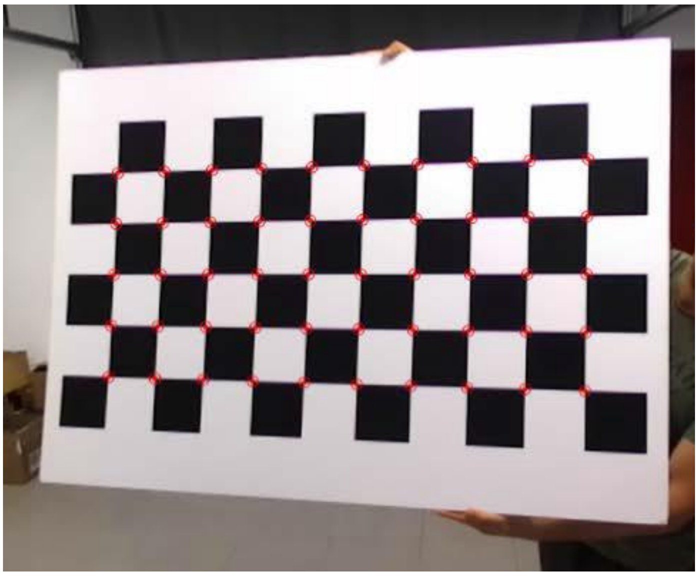


Figure 110: Calibration checkerboard

The first stage of the calibration process implies obtaining a point cloud by detecting the corners of the pattern (red points of Figure 111). The complete checkerboard needs to be viewed by several cameras at the same instant so that geometric constraints remain consistent among all different views. Our checkerboard pattern has 6 rows and 11 columns, and with a square edge of 101.6 mm (it must be that big to fill a significant portion of the image).

The estimation of the initial camera parameters results from computing a Euclidean calibration. 3D points are first organized in equal-sized triplets of aligned points (one triplet per checkerboard column, thus, some detected points are discarded). Figure 111 shows the points (in red) and triplets (in yellow) used in the optimization step.

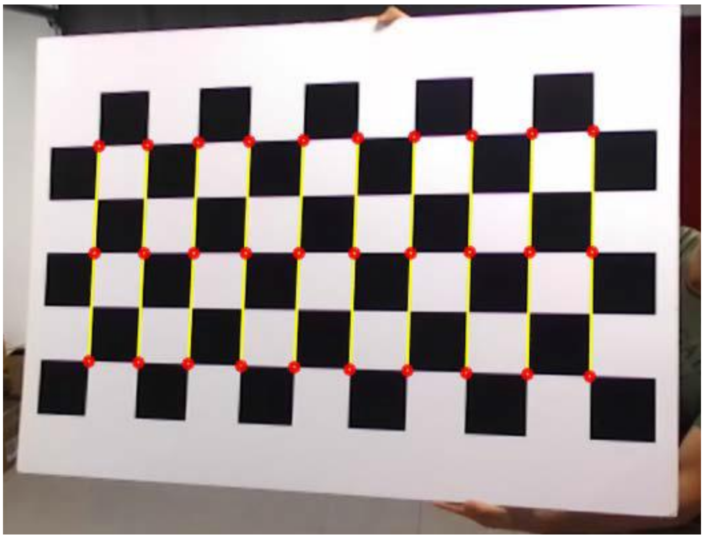


Figure 111: Detected points (red) and triplets (yellow) over the calibration checkerboard

Once the initial parameters have been obtained, a bundle adjustment optimization (specifically, the partitioned Levenberg-Marquardt) algorithm is used for the minimization of a cost function that includes the reprojection error and the variance in the triplet lengths. In practice, it is preferable to give more weight to the adjustment of the triplets, to prioritize their correct 3D reconstruction.

Figure 112 shows two examples of results obtained with the calibration procedure, using different arrangements of the cameras. The positions of both the cameras and the triplets are plotted.

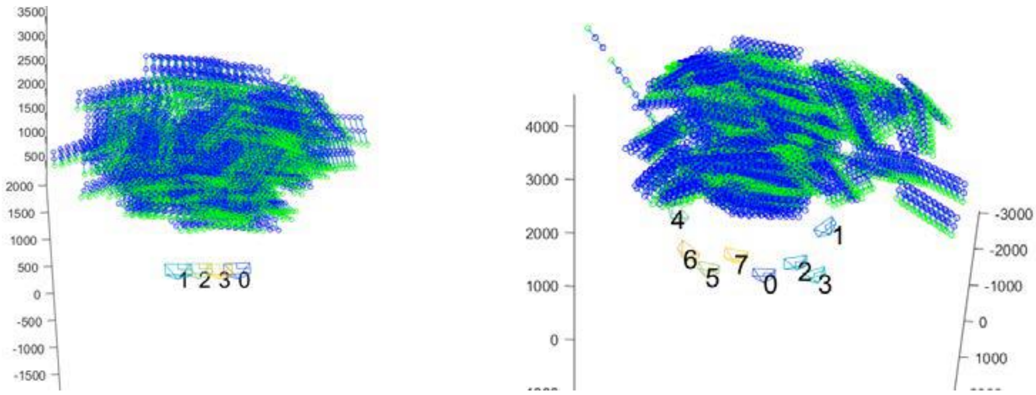


Figure 112: Calibration results example: detected points (green and blue) for different camera arrangements

In addition to the described computation of camera parameters, the calibration procedure allows to refine (to some extent) the depth data provided by the cameras, during the capture process, during the post-processing stage described in section 5.2.2

(ii) Background modelling

Once the cameras calibration process has been performed, the next step implies a background depth modelling. Assuming a controlled environment, the background of the sequences remains stationary; we take advantage of this to transmit just the foreground (i.e., the part of the image that changes every frame) to reduce data traffic. Therefore, a pre-calculated background model for both colour and depth is needed at the receiver, and a foreground segmentation must be performed at the transmitter in real time.

The creation of the background model is performed offline after the calibration process and the depth correction. Due to the noise inherent to the depth estimation of the cameras, a single/joint depth background model is calculated to have a complete and unique depth image that will provide a stable background for the synthesis. As for the colour background model, it is not unique, since one is created per camera, independently of the rest, by capturing several frames of the background (empty room) and obtaining the mean for each pixel. The colour models are used for both foreground segmentation (performed on-line during the capture process) and depth modelling.

The depth model of the background is acquired by using the structure from motion (SfM) paradigm that permits recovering the 3D structure of the scene by using 2D information from the different captured images and using the calibration parameters previously computed.

The first step is to extract key points from the colour images of the background. To enhance the input images so as to have the feature point extraction find in them more key points, a retinex filter is used. Then, key points are detected, and feature points extracted in the filtered images using the Accelerated-KAZE features extractor. With all the key points that have been detected on the input images, a 3D point cloud can be obtained using a regular SfM pipeline.

However, since the cameras are calibrated, a simpler and faster triangulation algorithm can be used: a point on an image, together with the calibration parameters of the corresponding camera, determine a line in 3D space; key points that are identified to be the same on several images ideally determine pencils of lines with a common 3D point, so the set of such common points would be the 3D point cloud wanted. In practice, due to errors both in the estimation of the camera parameters and the location of the key points, rays do not usually intersect, and the location of each 3D point is chosen such that it minimizes the sum of squared distances to the possible candidates.

Direct estimation of the implicit surface of the scene can be obtained because multi-scale retinex with colour restoration and A-KAZE key point extractors are used to build the point cloud, yielding denser point clouds than usual SIFT-based workflows. After an initial estimation of the surface 0, it is refined using a multi-view stereo variational algorithm based on photo-consistency constraints 0, yielding the final mesh (see Figure 113).

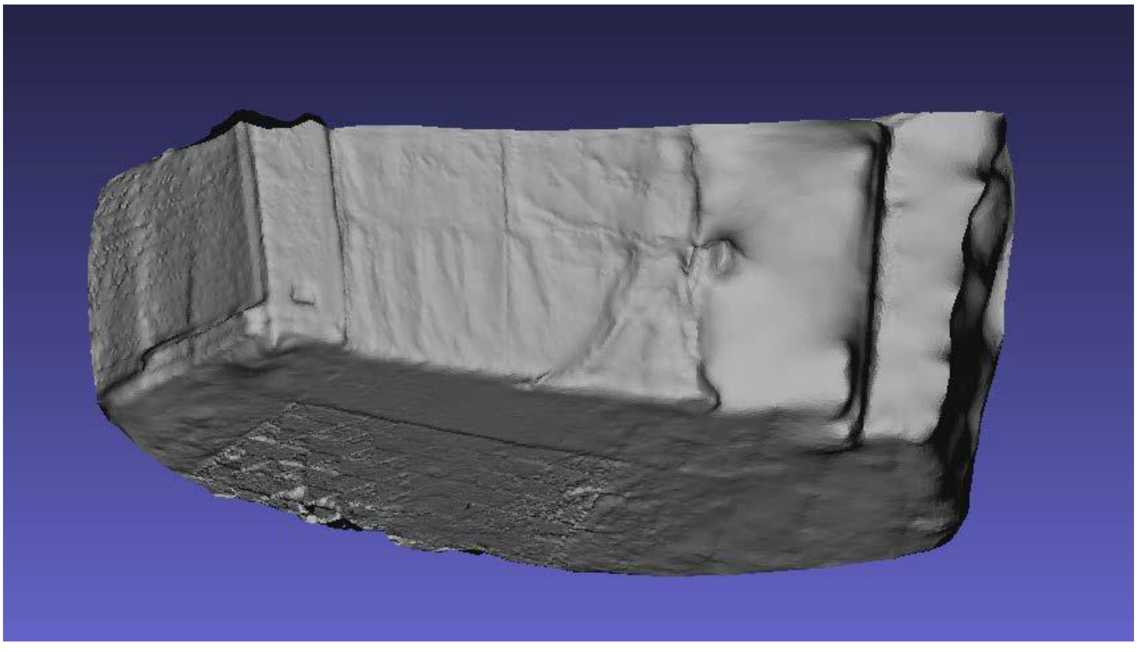


Figure 113: Background mesh example

Computer graphics pipelines typically include a “hidden surface removal” stage to determine the visibility of objects in a scene, commonly known as Z-buffer. The idea is that all the geometric elements in the scene are projected onto the screen and the closest element to each pixel is recorded, as well as its distance to the camera. A computer graphics rendering engine is used to project the 3D reconstruction of the background model to each of the cameras. Therefore, a depth map of the background is generated for each camera, and these depth maps are used in the edge server for the view synthesis rendering.

Annex D: Considerations about deployment scenarios

The modular architecture of 5G-RECORDS’s UC2 enables several possible deployment scenarios, each targeted to different business constraints and network setup conditions.

The section intends to identify the most relevant deployments of the 5G RECORDS components for a set of typical production scenarios, including deployment of security-related functions like NATs and Firewalls. Considering that many 5G Systems prevent reaching mobile devices from the outside due to security and misuse protection, provision of simple integration mechanisms as an HTTP REST Server on a wireless device may be impossible in some scenarios due to various operational security reasons.

Four main scenario categories are identified, namely: (i) Local production, (ii) Remote production using SNPNs, (iii) Remote production using PNI-PNPs, (iv) Integration of Multi-Production sites.

D.1 Scenario 1: Local Production

Scenario 1 is the most straight forward, since it represents a traditional local production. Here, the program director is sitting at the event location (on-premises) typically in an OB-Van (Figure 114)

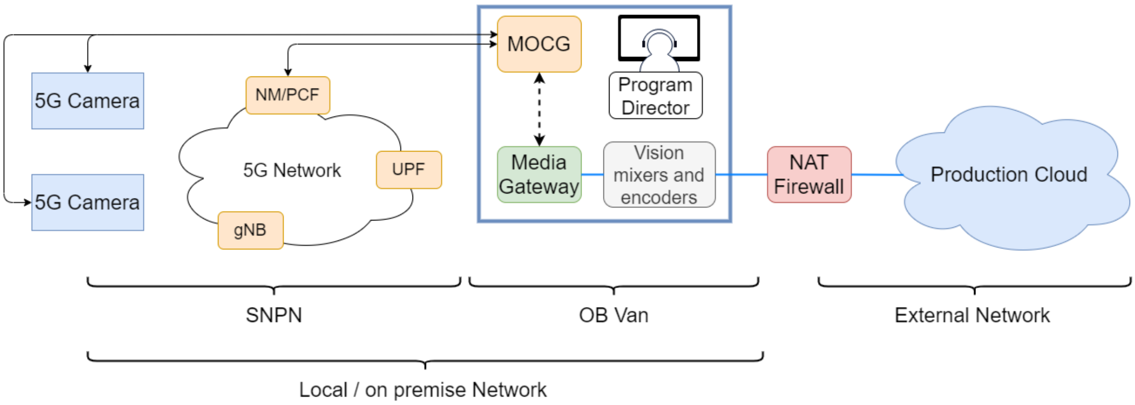


Figure 114: Scenario 1: Local Production

The local media production network, including all media production-related functions, is protected with a NAT / Firewall from the outside world. In this very controlled setup, no additional security restrictions are needed for mobile devices and the Media Operational Control Gateway may directly contact 5G cameras as in regular NMOS-enabled environments. The 5G cameras (and other devices) may expose an HTTP REST server, offering, in particular NMOS IS-04 and IS-05 interfaces for node discovery and registration and node inter-connection.

D.2 Scenario 2: Remote Production using SNPNs

In Scenario 2 the deployment is related to a case of remote production. In particular, the program director and the vision mixer are located in a remote production environment, like a physical production environment or a cloud-based environment.

Due to limited bandwidth with respect to normal requirements of SMPTE 2110 environments, media streams from 5G cameras should be transmitted as compressed video streams over the external network to the remote production environment.

Scenario 2a (Figure 115) represents a case where media equipment (and the MOCG and the Media Gateway) are deployed locally within the S-NPN system. Thus, the Media Gateway and the MOCG are deployed within the trusted boundaries of the S-NPN on-premises network. Thus, the MOCG may directly interact with 5G cameras and also with the Network Management functions like the PCF.

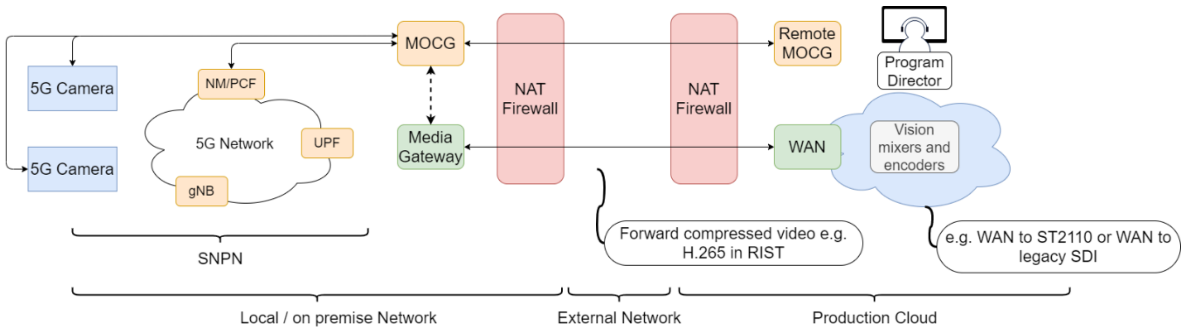


Figure 115: Scenario 2a: Remote Production

The Media Gateway in Scenario 2a may transcode or reformat the video stream between the 5G system and the external network. However, transcoding should be avoided due to latency issues.

Scenario 2b (Figure 116) represents a scenario, where media equipment (the MOCG and Media Gateway) is not deployed locally within the Standalone NPN system. Thus, the amount of on-premise functionality is reduced and only IP level functionality is deployed within the local S-NPN system.

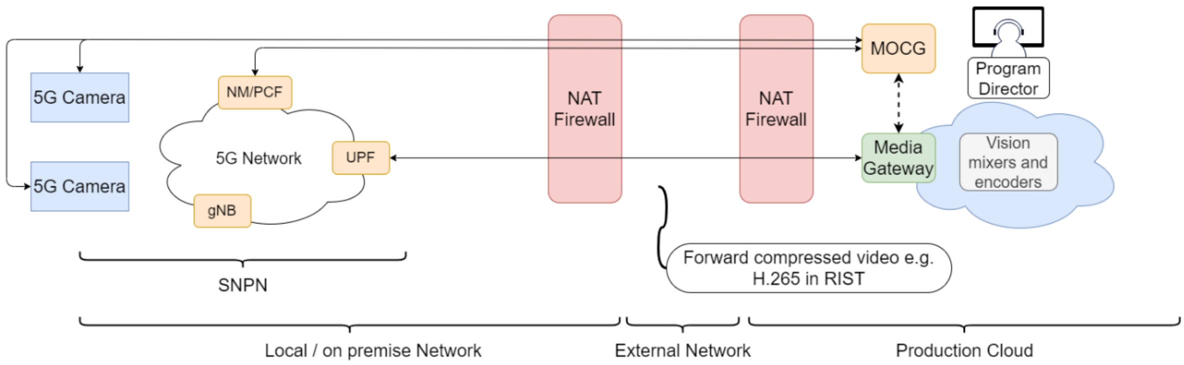


Figure 116: Scenario 2b: Remote Production

It should be noted that no media functionality is deployed within the local trust environment. Thus, the network management functions like PCF should be made accessible through the NAT / Firewall, e.g., by using port forwarding.

Accessing 5G cameras from the outside might be tricky, in particular when many devices are supported at the same time. A solution using 5G camera-initiated connection setup like e.g., using WebSocket or MQTT may be preferred.

Since the Media Gateway is deployed within the remote production environment, to avoid latency issues caused by transcoding the same video format and codecs should be used over the 5G radio interface and the external network.

The Key differences between Scenario 2a and 2b are the availability of a Media Gateway and the MOCG as on-premises functions. A key benefit of Scenario 2c is the reduced complexity of the on-premises network.

D.3 Scenario 3: Remote Production using PNI-PNPs

Scenario 3 deployments (Figure 117) are remote production deployments, which use 3rd party networks like PNI-NPNs as access networks. Note, that also a S-NPN may be shared (using concepts from PNI-NPNs) among multiple media producers.

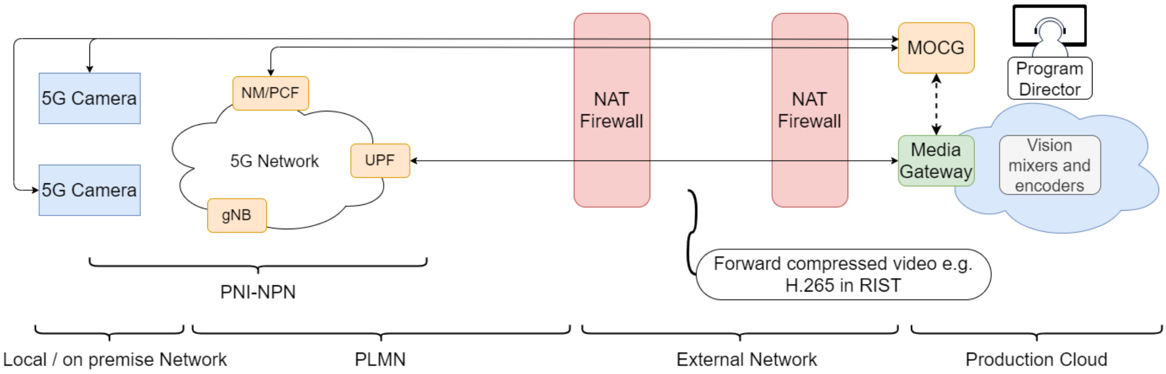


Figure 117: Scenario 3: PNI-NPN Remote Production

Since the 5G network is operated by a different provider, it is assumed that the local / on-premises production network is protected against missuse. In principle, scenario 3 looks very similar to scenario 2b and the same concepts (in particular for remote device control) can be applied.

D.4 Scenario 4: Integration of Multi-Production sites

Scenario 4 represents a multi-production deployment, whether multiple (local) 5G enabled event sites as operated simultaneously (Figure 118). Further, multiple media production sites are available to handle the received content.

We assume that a MOCG is deployed only within a primary production site and it can control devices in different 5G local event sites.

The Media Gateway function should be present in each remote production network since traffic via the untrusted / external network should be compressed.

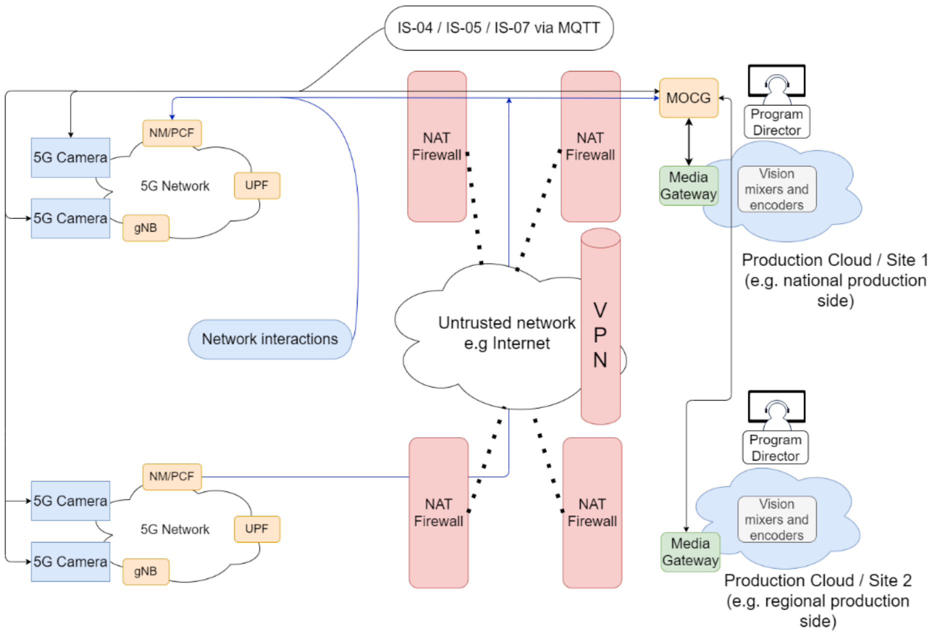


Figure 118: Scenario 4 Integration of Multi-Production sites (Main/Secondary PI)

Annex E: Overview of the AMWA Networked Media Open Specifications (NMOS)

Before going into detail of the operational control functions, it is helpful to introduce the AMWA Networked Media Open Specifications (NMOS). This is a family of specifications produced by the Advanced Media Workflow Association (AMWA), related to networked media for professional applications. They were created to help enable automation in live IP-based architectures through control plane APIs that are built on typical patterns used for web services (REST, publish-subscribe). NMOS specs are increasingly being adopted for applications using SMPTE ST 2110, and are part of the EBU's Technology Pyramid for Media Nodes (Tech 3371) [8]. The most relevant NMOS specs to this task are as follows:

- AMWA IS-04 allows media nodes (i.e., networked media devices) to register themselves, along with what they are (or are capable of) sending or receiving and allows control applications to query this information.
- AMWA IS-05 allows control applications to set up and remove connections between media nodes.
- AMWA IS-07 provides a publish-and-subscribe channel for sending time-based events such as tally information.
- AMWA IS-08 specifies how to handle audio channels in NMOS APIs.
- AMWA BCP-002-01 provides grouping of related resources, e.g., video, audio and data senders.
- The AMWA BCP-003 suite of specifications (including IS-10) covers secure communication and authorisation of NMOS APIs.

- AMWA BCP-004-01 lets a receiver describe any constraints on the types or parameters of streams it can receive.

Figure 119 shows the NMOS specifications in the context of the Network Media Systems Template.

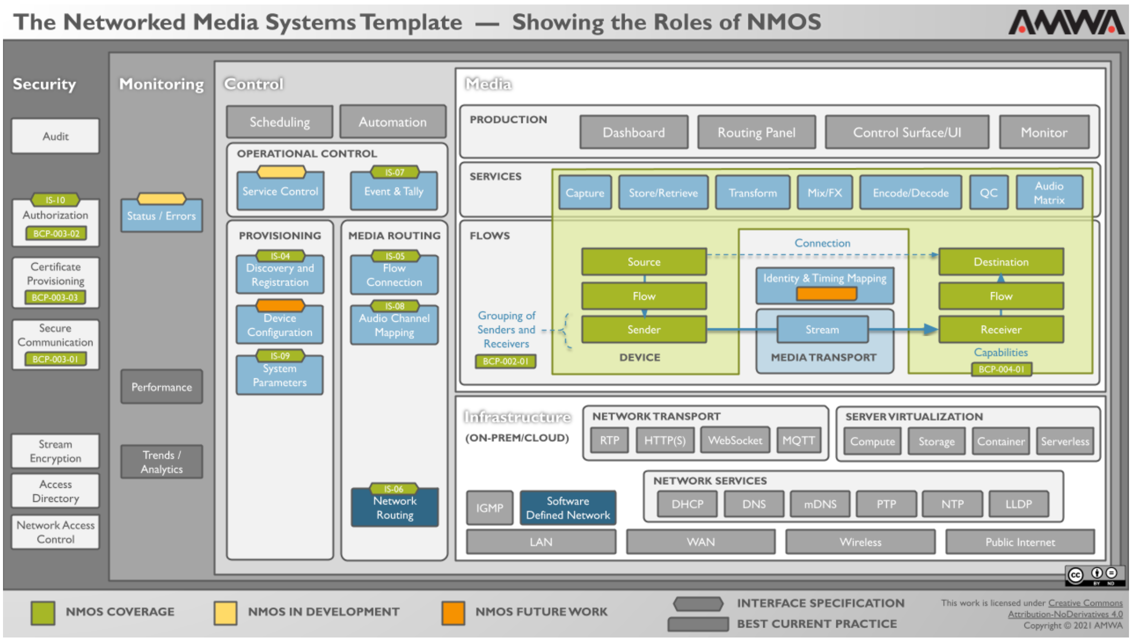


Figure 119: AMWA Networked Media Systems Template

Further details of NMOS and the specifications can be found at AMWA's website (https://www.amwa.tv/nmos-overview, https://specs.amwa.tv/nmos) [23] [24]. To date NMOS has mostly been used with ST 2110 uncompressed multicast video and audio streams within wired facilities. However, NMOS can be used with other types of streams, including unicast. There is growing interest in other areas, such as professional audio-visual applications using compressed video, and where media is streamed between facilities over WAN connections (VSF WAN group) [25]. The project is investigating the possible relevance of NMOS within 5G media applications, and whether any extensions to the NMOS specs would be required.

7 References

[1] 5G-RECORDS project, “D4.1 - Integration of 5G components (phase 1)”.

[2] 5G-RECORDS Project, “Deliverable D2.1 - Use-cases, requirements and KPIs”.

[3] 3GPP group, “Technical specification (TS) 23.501 - System architecture for the 5G System (5GS)”.

[4] IEEE, “1588-2019 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems”.

[5] “LU800,” [Online]. Available: https://www.liveu.tv/products/field-units/lu800.

[6] “LU2000-SMPTE,” [Online]. Available: https://www.liveu.tv/products/decoders/lu2000-smpte.

[7] (JT-NM), Joint Task Force on Networked Media, “JT-NM Website,” 4th September 2015. [Online]. Available: https://www.jt-nm.org/reference-architecture.

[8] EBU, “Tech 3371 - Minimum User Requirements To Build And Manage An IP-Based Media Facility,” July 2020. [Online]. Available: https://tech.ebu.ch/publications/tech3371.

[9] AMWA, “Networked Media Systems. The big Picture.,” March 2021. [Online]. Available: https://static.amwa.tv/networked-media-systems-big-picture-2021-03-05.pdf.

[10] Github/Sony, “An NMOS C++ Implementation,” [Online]. Available: https://github.com/sony/nmos-cpp.

[11] Github/BBC , “NMOS Registry Registration API Service,” [Online]. Available: https://github.com/bbc/nmos-registration.

[12] Github/Sony, “An NMOS Client in JavaScript,” [Online]. Available: https://github.com/sony/nmos-js.

[13] AMWA, “AMWA IS-05 NMOS Device Connection Management Specification: Overview,” [Online]. Available: https://specs.amwa.tv/is-05/v1.1/docs/1.0._Overview.html.

[14] Confluent.io, “https://www.confluent.io/,” [Online]. Available: https://www.confluent.io/blog/kafka-fastest-messaging-system/#latency-test.

[15] Muetsch.io, “https://muetsch.io/,” [Online]. Available: https://muetsch.io/basic-benchmarks-of-5-different-mqtt-brokers.html.

[16] Flespi.com, “https://flespi.com/,” [Online]. Available: https://flespi.com/blog/benchmarking-popular-mqtt-json-implementations.

[17] Scalagent, “www.scalegent.org,” [Online]. Available: http://www.scalagent.com/IMG/pdf/Benchmark_MQTT_servers-v1-1.pdf.

[18] Xilinx Inc., “Zynq UltraScale+ MPSoC ZCU106 Evaluation Kit,” [Online]. Available: https://www.xilinx.com/products/boards-and-kits/zcu106.html.

[19] AMWA, “NMOS IS-07: Message types,” [Online]. Available: https://specs.amwa.tv/is-07/v1.0.1/docs/2.0._Message_types.html.

[20] “OAuth 2.0 Website,” [Online]. Available: https://oauth.net/2/.

[21] 3GPP, “TS 23.501 - System architecture for the 5G System (5GS)”.

[22] 3GPP, “TS 23.502 - Procedures for the 5G System (5GS)”.

[23] AMWA, “NMOS. A family of open, free of charge specifications,” [Online]. Available: https://www.amwa.tv/nmos-overview.

[24] AMWA, “Networked Media Open Specifications: Introduction,” [Online]. Available: https://specs.amwa.tv/nmos/.

[25] VSF - Video Services Forum, “SMPTE ST 2110 over WAN,” [Online]. Available: https://vsf.tv/SMPTE_ST_2110_over_WAN.shtml.

[26] LiveU , “Tally Light,” [Online]. Available: https://www.liveu.tv/ip-solutions/tally-light.

[27] “LU800,” [Online]. Available: https://www.liveu.tv/products/field-units/lu800.