FUDGE-5G

FUlly DisinteGrated private nEtworks for 5G verticals

Deliverable D2.3

FUDGE-5G Converged 5GLAN with TSN and Multicast Capabilities

Version 1.0

Work Package 2.2

Editor	Mika Skarp
Status	Final
Delivery date	June 30th 2022

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 957242

Disclaimer

Project details

Project title: FUlly DisinteGrated private nEtworks for 5G verticals
Acronym: FUDGE-5G
Start date: September 2020
Duration: 30 months
Call: ICT-42-2020 Innovation Action

For more information

Project Coordinator
Prof. David Gomez-Barquero
Universitat Politecnica de Valencia
iTEAM Research Institute
Camino de Vera s/n
46022 Valencia
Spain

http://fudge-5g.eu
info@fudge-5g.eu

Acknowledgement

FUDGE-5G has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement Nº 957242. The European Union has no responsibility for the content of this document.

Abstract

This document describes the outcome of Task 2.2, which is about providing LAN functionality on 5G network. The set of features that have been implemented are 5GLAN, 5GTSN and 5G-Multicast.

The first section describes the 5GLAN (5G Local Area Network) implementation and results. 5GLAN consists in providing LAN-like mechanisms for UEs, such as discovery of other UEs in the same 5G Private Virtual Network and routing of Ethernet frames, bringing more flexibility to 5G. This feature is implemented over Cumucore 5G Core.

The second feature is 5GTSN (5G Time Sensitive Networking), which allows interworking of 5G and TSN networks. This is essential for industrial use cases that need deterministic networks. In FUDGE-5G, it is used at the Industry 4.0 use case. It is also implemented over Cumucore 5G Core, and it leverages the 5GLAN functionalities.

Lastly, the document describes 5G-Multicast, developed by UPV and FOKUS. It consists of implementing multicast capabilities at the N3 (between the UPFs and gNBs), which allows saving bandwidth for use cases such as media delivery. This feature will be implemented over the Open5GCore.

List of Authors

Authors	Partner
Mika Skarp, Jose Costa-Requena	CMC
Borja Iñesta, Josep Ribes, Carlos Barjau, David Gómez-Barquero, Manuel Martínez	UPV
Dumitru Prutean, José Herrero, Manuel Fuentes	5CMM

List of Reviewers

Reviewers	Partner
Peter Sanders	O2M
Steve Appleby	BT
Pousali Chakraborty	FHG

Acronyms

5GC	5G Core
AF	Application Function
API	Application Programming Interface
AR	Augmented Reality
CB	Cell Broadcast
CBCF	Cell Broadcast Control Function
CCC	Command and Control Centre
CCTV	Closed-Circuit Television
CNC	Centralized Network Controller
DL	Download
DSTT	Device Side TSN Translator
E2E	End-to-End
GM	Grand Master
gPTP	Generic Precision Time Protocol
GTP	GPRS Tunnelling Protocol
HOT	Heat Orchestration Template
ICMP	Internet Control Message Protocol
IGMP	Internet Group Management Protocol
MB-SMF	Multicast and Broadcast - SMF
MB-UPF	Multicast and Broadcast - UPF
NC	Network Component
NEF	Network Exposure Function
LAN	Local Area Network
MBS	Multicast and Broadcast Service
MCC	Mobile County Code
MNC	Mobile Network Code
NF	Network Function
NPN	Non-Public Network
NRF	Network Resolution Function
NWTT	Network TSN Translator
PCF	Policy Control Function
PFCP	Packet Forwarding Control Protocol

PLMN	Public Land Mobile Network
PTP	Precision Time Protocol
PVN	Private Virtual Network
PWP	Public Warning Portal
REM	Real time Ethernet Multiprotocol
RTT	Round-Trip Time
SA	Stand-alone
SBA	Service-based Architecture
SCC	Service Chain Controller
SCP	Service Communication Proxy
SFEC	Service Function Endpoint Controller
SH	Service Host
SMF	Session Management Function
SFV	Service Function Virtualisation
SFVO	Service Function Virtualisation Orchestrator
SLA	Service Level Agreement
TEID	Tunnel Endpoint Identifier
TMGI	Temporary Mobile Group Identity
TSe	Time Stamping - egress
TSi	Time Stamping - ingress
TSN	Time Sensitive Network
UE	User Equipment
UL	Upload
UPF	User Plane Function
VAO	Vertical Application Orchestrator
VR	Virtual Reality
VTB	Virtual TSN Bridge
WP	Work Package

Executive Summary

Private 5G networks should leverage all the benefits of 5G while maintaining the current capabilities of existing corporate LAN solutions. 5GLAN is therefore an essential technology that will be embraced for offering private communication using IP and/or non-IP, and to seamlessly integrate 5G with fixed and wireless (Wi-Fi) LAN. Furthermore, 5GLAN can serve as basis for more innovative technologies. The existing 5GLAN functionality in Rel-16 is based on the emulation of LAN features in the 5G core network. FUDGE-5G will go one step further by extending 5GLAN support to unified access under one address “all-Ethernet” domain, opening up the capability to deliver content more efficiently to end users consolidating multiple existing networks into a single converged 5G network.

Using 5G technology we can create very large geographical Local Area Networks where end devices can be freely connected to each other even when they are moving around, and network is compatible with existing IT infrastructure.

In task T2.2 we have developed a working demonstrator using commercial 5G SA radio together with commercial 5G SA NC. 5G SA NC has specific 5GLAN capability to manage LAN connections. TSN network is using 5GLAN capability and UE are synchronized with TSN source. Multicast functionality was implemented to discover UEs easily from the network. Together these functions make mobile network to behave like Ethernet network without geographical limitations.

Table of content

1 Introduction

There is an increasing demand in industries to integrate fixed IT infrastructure with mobile infrastructure, where fix devices and mobile devices need to be part of the same Ethernet groups. Some industrial use cases show the need for controlling robots, orchestrating automated guided vehicles or having push to talk communications, but these use cases show examples where each new feature is used separately.

Use cases like audio production, because of its complexity, show the need of some of these features working together. An example could be an audio production setup where many microphones connected need to be grouped, synchronized and each microphone data stream needs to be broadcasted into ear monitors and loudspeakers. In such scenario, features like 5GLAN, 5G-TSN and 5G-MBS features are needed to fulfil the use case requirements.

Task 2.2 introduces the performance and features of 5GLAN closer to existing native LAN solutions. The main challenge is the inclusion of 5GLAN bearers, spanning across the existing 5G vertical protocols. The native approach is not feasible with the current cellular technology, since GPRS Tunnelling Protocol (GTP) encapsulates all traffic coursing through the cellular network. The main consequence is that packets coming from fixed networks to cellular networks will get encapsulated with GTP and decapsulated at UPF (User Plane Function), forbidding the use of a common address domain, adding more overhead and filtering broadcast Ethernet packets (e.g. ARP procedure).

Mobile networks currently use best-effort IP networking in the backhaul which delivers a flat network for best effort traffic management between the radio access network and the UPF. Task 2.2 incorporates 5G-TSN to its platform, via pre-provisioning resources for URLLC (Ultra-Reliable Low Latency Communications), both for IP and non-IP transport. In addition to URLLC to deliver low delay end-to-end communications between 5G mobiles and fixed LAN devices, multicast communications are required to provide fixed LAN native device and address discovery. In Task 2.2 we have specified and prototyped required NFs (Network Functions) for integrating 5G as part as of an end-to-end TSN network.

Many use cases require delivering the same information from one source to multiple clients at the same time, such as media delivery, public warning or IoT. Having the source send the information once for every client is not scalable and uses many resources at the client itself and in the network. This deliverable explains the implementation of 5G-MBS over the Open5GCore. The feature consists of bringing IP multicast to the N3 interface (interface between the gNB and UPF). This helps reduce resources usage, since now the source will only need to send the information once, and the 5G Core will take care of making the information arrive to the clients interested in receiving it. Having IP multicast on N3 helps reducing bandwidth usage at the transport network between UPF and gNBs.

2 5GLAN

2.1 Feature description

5GLAN feature enables the integration of mobile networks as a part of an IT infrastructure. Connectivity based on 5GLAN reduces the use of Ethernet cables.

For traditional Ethernet communication, a device needs to find out the MAC address of its peer device. The device, according to the destination IP address XXX derived from the IP packet that needs to be delivered, would initiate an enquiry “who has the IP address XXX”, and this enquiry is broadcasted to all the devices belonging to the same LAN. The device who has this IP address will respond and provide its MAC address to the requesting device.

For 5GLAN case, it is essential to allow a UE to obtain the identifiers of other UEs in the same private communication of 5G LAN-type service for application communication use. In LAN networks, devices make use of discovery mechanism (e.g Bonjour, UPNP) to discover other devices online to be used and their characteristics. This discovery mechanism makes use of the multicast capabilities of the network. Therefore, it is important that 5G LAN supports discovery mechanisms.

The 5G network shall support the routing of non-IP packet (e.g. Ethernet packet) efficiently for private communication between UEs and Control Center (UE). A subset of UEs that are members of the 5G Private Virtual Network (PVN) can establish a multicast communication within their on-demand group, e.g. equipment A creates a multicast on demand and B and C join this multicast to receive A’s multicast messages.

The 5GLAN Group that creates the PVN may be dynamically created by an operator or possibly requested by Application Function via service exposure (Figure 1).

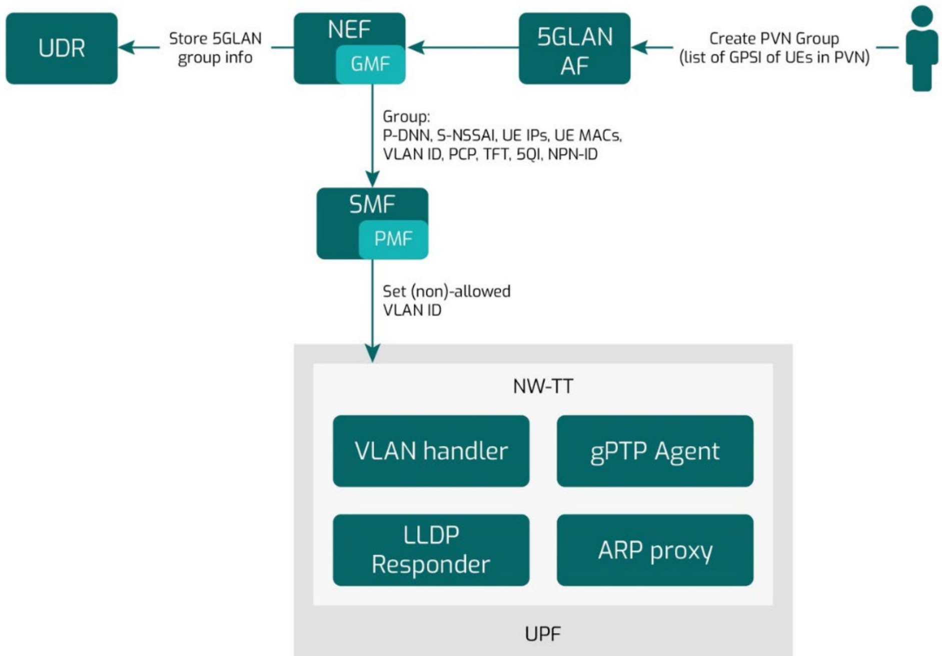

Figure 1: 5GLAN group management process

Identities, a Non-Public Network ID (NPN-ID) identifies a non-public network. The NPN-ID supports two assignment models:

- Locally managed NPN-IDs are assumed to be chosen randomly at deployment time to avoid collisions (and may therefore not be unique in all scenarios).
- Universally managed NPN-IDs are managed by a central entity are therefore assumed to be unique.

Identities, a Closed Access Group (CAG) ID uniquely identifies a closed access group (CAG) in a PLMN. PLMN ID consists of MCC 999 assigned by ITU for private networks and an MNC defined by 3GPP to identify the network as part of a non-public network. The configuration of the UE is performed from a logical Application Function (AF) that configures the UE via the PCF directly or indirectly via the NEF first and then via the PCF.

2.2 FUDGE-5G use case requirements

2.2.1 Requirements from WP1

End-to-end latency: Latency is measured as the time delay from message generated at source until its arrival at the end node. The 5G NPN delay is considered as part of the E2E latency, with values within 1-2 ms range.

Throughput: In typical 5G consumer use cases, DL throughput is of utmost importance. However, in industrial use cases UL throughput is equally important. Control related traffic does not require high throughput in both directions, but condition monitoring, optimization, VR, AR, and CCTV applications require significantly higher throughputs. Expectation is a throughput up to 200 Mbps in the UL and up to 400 Mbps in the DL.

2.3 5GLAN Implementation

2.3.1 5GLAN Virtual Group Manager

5G Core has a user management service where allowed users of the network are configured. User definition has also information about services users are allowed to use. For example, access to the network slices is preconfigured.

During the attachment process 5G Core will ensure that the user has been granted an access right to use the network. In the end of the attachment process active data flow between User through Network to Internet is established using Interfaces N3 and N9.

Interface N2 is used for signalling between UE, Radio Access Network and Core Network during the active data flow. User Equipment delivers for example measurement results over N2 interface to AMF and receives for example radio channel power settings over this interface as shown in Figure 2.

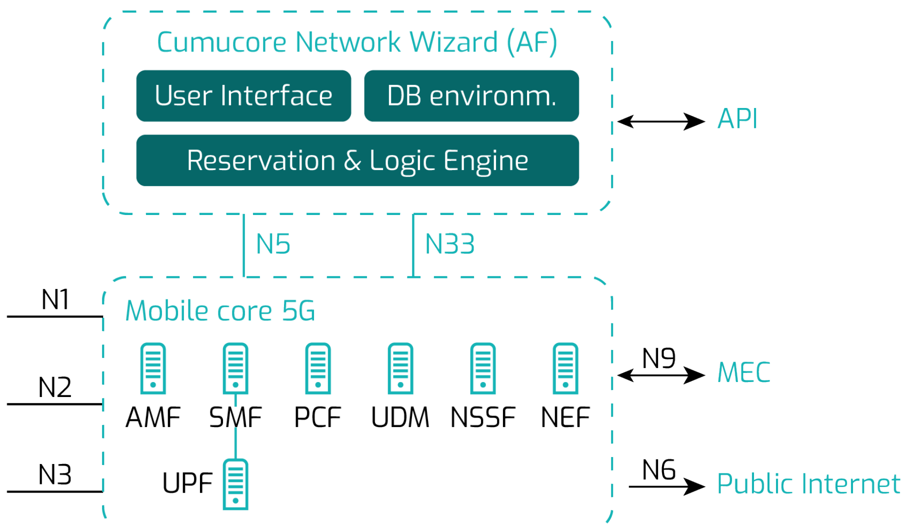

Figure 2: 5GLAN virtual group manager architecture

Mobile core networks are organized into slices; in other words separate virtual networks. Mobile Network core slices have different capacity and latency capabilities. Network Slice can also have 5GLAN and TSN functionalities. Network slice has its own UPF and can also have other own Network Functions if needed. Inside the network slice there are user profiles that can be network slice specific. Network slices and user capabilities are managed through Cumucore Network Wizard (CNW). User profiles are attached to data flows to differentiate Quality of Service levels. User profile includes:

- Connection type: Guaranteed bitrate / Non-guaranteed bitrate.
- Max capacity downlink / uplink.
- Quality of Service class.
- Traffic Priority classifier.
- 5GLAN definitions.

CNW is responsible for managing network slice life cycle and creating data flows per request. CNW ensures that network slices can meet the promised KPI for all active and reserved upcoming data flows. Specific data flows can be requested outside of the network. There is an open API to receive data flow requests. Data flow request includes flowing parameters:

- Source and destination IP address.
- Requested profile.
- Downlink and Uplink bitrate.
- 5GLAN parameters.
- Time that data flow is requested (start & end time).

CNW has information on all active data flows, how much resources they are consuming and the cell they are in. Network Slicing manager has information about users' real time radio environment and location through NEF function. Based on this information Network Slicing manager will decide if requested data flow can be served by the network slice without causing any KPI violation to existing or upcoming reserved data flows.

New data flow requests will be sent over N5 interface to PCF that will set up a data flow and send requests to the radio network over N2 interface. After all settings in the Radio and Core networks are done, the user is informed over the N1 interface to start using new data flow for a specific application.

CNW has a user interface where the operator can see every application's resource utilization level. This information can be used to improve overall efficiency of the network.

2.4 Used testing methods and equipment

2.4.1 Test cases

5G system should interwork with industrial applications and existing network management solutions, with the aim of hiding implementation details (Figure 3).

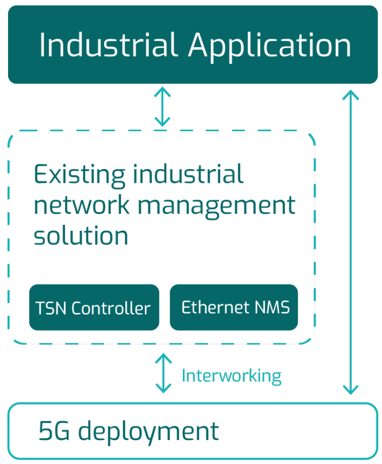

Figure 3: 5GLAN Test environment

2.5 Test results

Testing was carried out in Telenor laboratory in Fornebu, Norway May-June 2022. The actual test set-up is described in the Figure 4: Telenor test set-up.

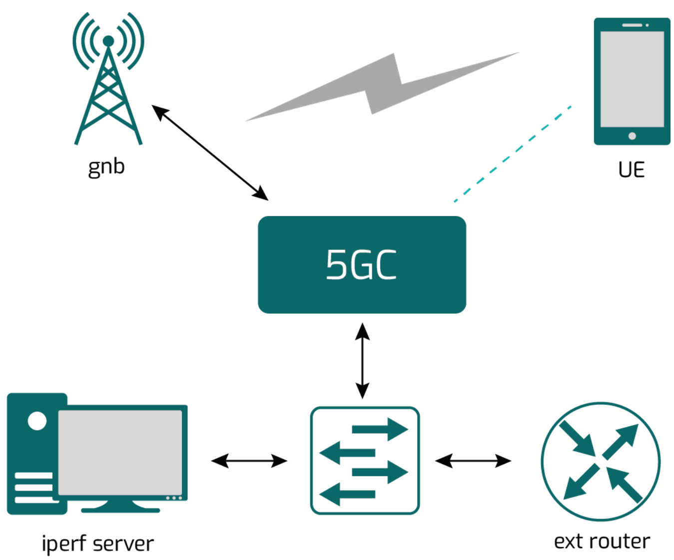

Figure 4: Telenor test set-up

The test procedure is the following:

1. connect 1 UE
2. ping GW at 192.168.19.1
3. ping UE from iperf laptop
4. perform internet speed test
5. perform local iperf test
6. disconnect and repeat for other UE
7. Repeat with 2 UEs

Results Huawei:

Case 1

- UE has normal IP.
- Laptop host iperf3 in LAN, run iperf3 from UE
- Traffic goes via GW to laptop

Metric	Result
Downlink speed	166Mbps
Uplink speed	100Mbps
ping from UE to laptop	1358/1343 1% loss ~270 seconds min/avg/max/mdev 4.6/9.9/66/3 ms

Case 2

- UE has LAN IP
- Laptop host iperf3 in LAN, run iperf3 from UE
- traffic goes directly to laptop

Metric	**Result**
Downlink speed	166Mbps
Uplink speed	115Mbps
ping from UE to laptop	1608/1593 0% loss ~322 seconds min/avg/max/mdev 5.0/10.7/81.7/3.7 ms

Case 3

- UE has LAN IP
- UE1 host iperf3 in LAN, run client iperf3 from UE2

Metric	Result
ping from UE .131 to other LAN UE .132:	468/468 0% loss ~93 seconds min/avg/max/mdev 11.0/18.9/27.8/2.6 ms

Case 4

- Changed slice parameters for "default-slice" used for UEs for DL and UL values.
- In this list there is some AMBR value testing where first is DL followed by UL, DL/UL.

Iperf test from UE1 to laptop:

Metric	Result
Downlink speed	155Mbps
Uplink speed	5Mbps
Ping from UE1 to laptop	468/468 0% loss ~93 seconds min/avg/max/mdev 11.0/18.9/27.8/2.6 ms

UE1 same iperf server:

Metric	Result
Downlink speed	198Mbps
Uplink speed	45Mbps

UE2 same iperf server:

Metric	Result
Downlink speed	145Mbps

Uplink speed	45Mbps

UE1+UE2, two iperf servers:

Metric	Result
Aggregated downlink speed	146 + 160 = 306Mbps

UE1+UE2+CPE, three iperf servers:

Metric	Result
Aggregated downlink speed	119 + 156 + 132 = 407 Mbps

UE1 to UE2 iperf, seems to be limited by uplink on single device:

Metric	Result
Downlink speed	113Mbps

Testing shows uplink ~100Mbps, so this is consistent.

3 Time Sensitive Network TSN

3.1 Feature description

Time Sensitive Networks (TSN) functionality requires time synchronization information which is delivered by a PTP server connected to the Core. TSN feature also requires 5GLAN functionality to be in use.

The industrial LAN may also consist of TSN-enabled Ethernet bridges. The latest release of 5G specification supports the fully centralized TSN configuration model, where a central controller should be able to configure both Ethernet and 5GS bridges as a unified network.

The IEEE 802.1Q provides the definition of TSN: as a standard through which deterministic services can be provided on Ethernet (802) networks. Through deterministic delivery of packets, the packets should be delivered within a given window while eliminating delays notable for causing congestion and errors in networks. Even though, TSN is a layer 2 technology, it does not use the IP protocol but rather the Ethernet protocol. This means that the forwarding decisions are then meant to be handled by the TSN bridges.

The Centralized Network Controller (CNC) is the entity in TSN network that has complete knowledge of the network topology and is responsible for configuring the bridges to enable transmission of TSN streams from source to destination [1]. The 5G control plane is interacting with CNC via the TSN Application Function (AF) which maps between the TSN parameters and the 5GS parameters. The TSN AF reports the 5GS bridge capabilities such as minimum and maximum delays between every port pair and per traffic class, including the residence time within the UE and DS-TT via TSN-AF to CNC (Figure 5**).**

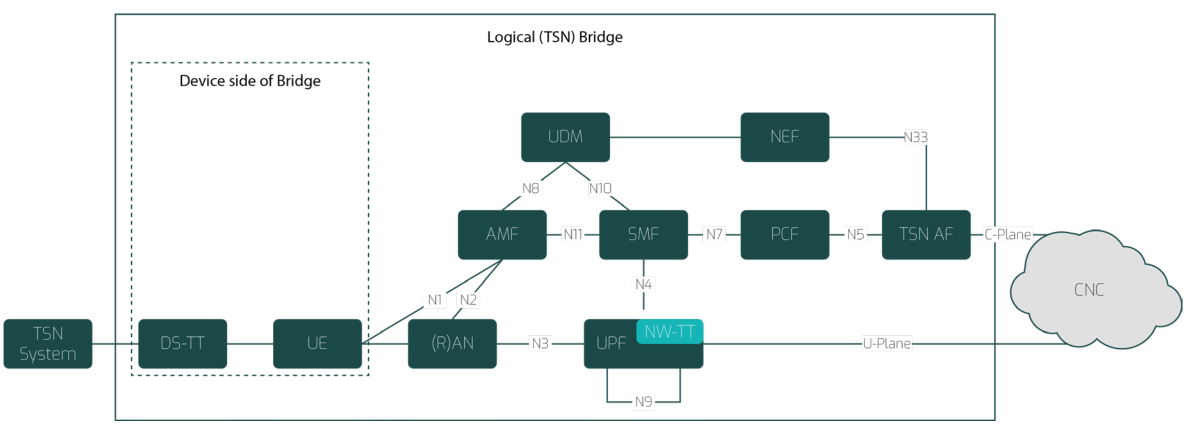

Figure 5: Logical TSN bridge

This section describes the current network functions defined in 3GPP as part of 5G architecture. The 5GS by implementing or incorporating the TSN translators, i.e., Device-

Side TSN Translator (DSTT) and Network-Side TSN Translator (NWTT) then becomes a virtual TSN bridge (VTB) that is the capable of interfacing with TSN devices to ensure an E2E transmission/communication between the different end-stations (talkers or receivers). The DSTT is integrated with the UE while the NWTT is integrated with the UPF. The gNB then serves the wireless connections to the UEs. Only the UP is considered since that is where the gPTP messages are signalled. With the enhancements to incorporate the TSN translators, the VTB then appears as any other standard TSN bridge in the network when it is connected to other TSN bridges. To hide the complexity of the 5GS, the DSTT and NWTT act as logical ports of the VTB. As a result, DSTT and NWTT towards the network both adopt TSN bridge ingress and egress port operations.

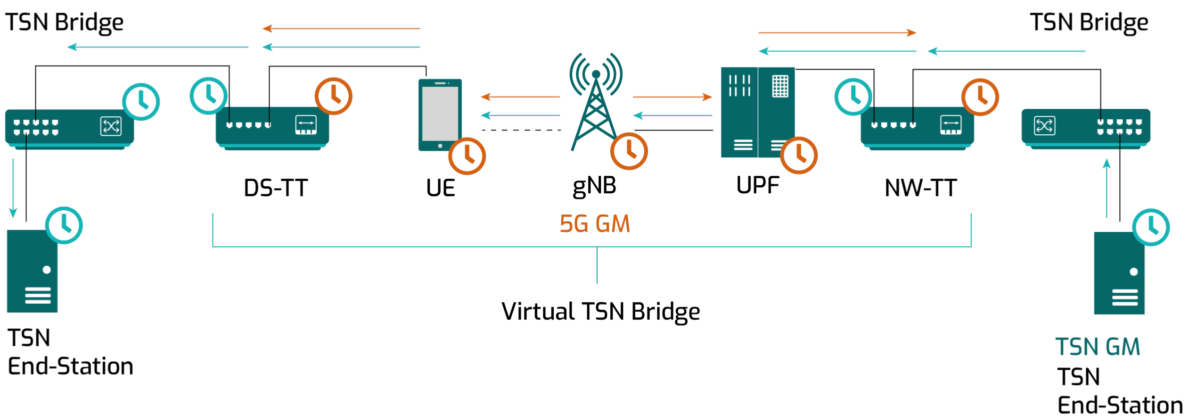

Figure 6: Physical TSN network

The 5GS to support the whole industrial network, both Medium Access Control (MAC) learning and flooding-based forwarding as well as the static forwarding configured by the CNC controller need to be supported. 3GPP has defined that a 5GS can be modelled as one or more virtual TSN bridges. The 5GS defines the TSN application function (TSN-AF) that interacts with TSN CNC to manage the 5GS as a native TSN bridge.

3.2 FUDGE-5G use case requirements

Time accuracy is a critical aspect for TSN. If there are different TSN controllers or 5G TSN systems, it is necessary to keep time synchronization among all of them. End-devices need to be always synchronized to avoid any type of accident in industrial scenarios. For the industrial use case, these devices are required to have a synchronicity time from 10 µs to 50 ms [2].

3.3 TSN Implementation

TSN is not a single component but rather a combination/collection of different standards and amendments thereof, together with multiple projects published or being developed by the IEEE 802.1 TG. It defines provisions on how to achieve deterministic services which are operated on top of IEEE 802 networks [3], [4]. From the TSN toolbox, choices can be made depending on different application needs from the listed items shown:

1. Time Synchronization
2. Bounded low latency
3. Reliability
4. Resource Management

The implementation of TSN support in 5G network starts with the development of time synchronization functionality, which is critical for TSN integration. Therefore, the first components to be implemented as part of the TSN are the network functions that will handle the delivery of time synchronization between fixed TSN network and 5G connected devices. The remaining items such as latency, reliability and resource management will be configured using existing mechanisms already standardised in 3GPP as part of 5G network management. The latency and resource management will be done using internal 5G network functions and will be available to external TSN controller through the TSN-AF.

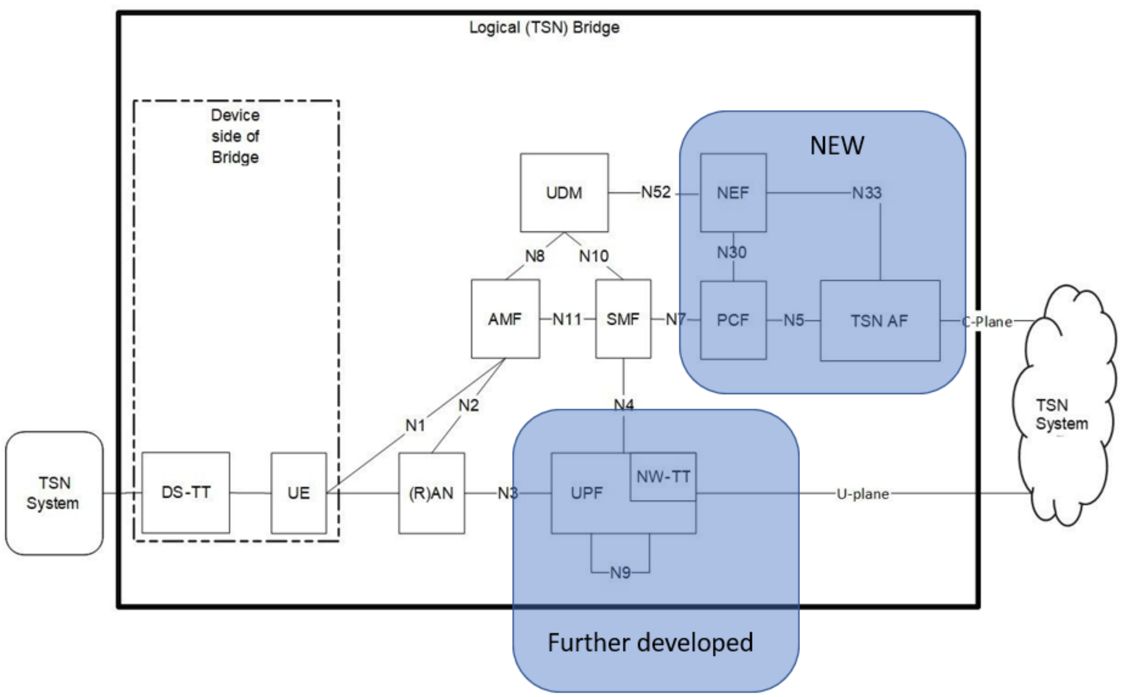

Figure 7: TSN implementation

The first functionality to be designed and implemented consist of the distributing of time synchronization between a fixed TSN network and the mobile 5G devices. This is performed thanks to the NWTT and DSTT, which are implemented at the network and device side, respectively. These two components will take care of distributing time information between fixed and 5G devices.

The different mechanism involved in the distribution of TSN clock and time sampling are covered in the Release 16 of the IEEE 802.1AS. When a downlink (DL) gPTP message is received by the NWTT, an ingress time-stamping (TSi) is made for each and every message. Additionally, the cumulative rate ratio that is obtained from the gPTP message payload is used in the calculation of the link delay of the upstream TSN node which is expressed using the GM time. A new cumulative rate ratio is calculated by the NWTT which then modifies the gPTP message payload by replacing the previous rate ratio (from upstream) with the new one and in the gPTP packet a TSi suffix field is added. Additionally, the NWTT adds the link delay received from the upstream TSN node to the GM time and put into the correction field. The resulting gPTP messages are then forwarded to the UP through the use of the UPF and transmitted on a QoS Flow in accordance with the upper bound residence time which is a requirement specified in the IEEE 802.1AS.

Upon receiving new gPTP messages, the UE forwards them to the DSTT. Egress time stamping (TSe) is created for the gPTP event messages for domains that are external. Calculations are then made for the residence time taken by the gPTP in the 5GS, by taking the difference between the TSi and TSe. The residence time spent in the 5GS is then converted by the DSTT in GM time using the rate ratio found in the gPTP messages. The residence time is then added and TSi information removed from them payload suffix field. The resulting gPTP message is then sent to the downstream node.

3.4 Topology

In building a LAN network TSN prototype, one of the tools that can be used is the TSN Evaluation Kit. The kit provides TSN gateway functionality, which enables a quick access to the different TSN features and helps in understanding better how TSN works. The REM Switch chip (Innovasic's fido5000) can be utilized to provide a TSN solution to different applications. The TSN gateway supports different IEEE specifications such as 801.1AS, AS-REV (Time synchronization), 802.1Qbv (Scheduled Traffic), 802.1Qci (Ingress Policing), 802.1CB (Seamless Redundancy), 802.1Qcc (Stream Reservation Protocol) and 802.1Qbu/802.3br (Preemption). Out-of-the box TSN gateway is re-installed with software that offers support for 802.1AS, 802.1Qbv, 802.1Qcc and stream translation. Currently support for 801.1Qcc is offered through a web-server but future implementations will offer support for gateway configuration through a Central Network Controller (CNC).

Figure 7 shows the topology that will be used for LAN networks when testing and measuring E2E synchronization together with the other components involved such as talkers and listeners. The 5GS is implemented as a logical TSN bridge and consists of the 5G network components together with the TSN translators. The figure shows a prototype that will be used to test and measure the integration between 5GS and TSN in relation to time synchronization.

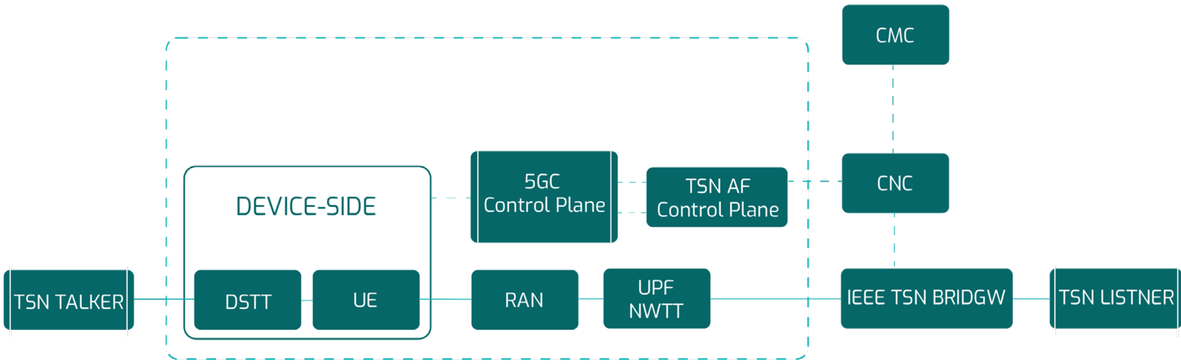

Figure 8: TSN Control and User planes

The prototype consists of the NWTT and DSTT that handle the time synchronization messages. In the prototype the upstream traffic that comes from different TSN bridges (such as TSN evaluation kits used in this setting), on arrival at the NWTT, need to be modified. These modifications are needed in order, to firstly, insert ingress time-stamping into the PTPv2 packets. These timestamps are either inserted into the SYNC messages (one-step) or the exact time at which the SYNC messages were sent, inserted into the FOLLOW_UP messages (two-step). Secondly, the upstream delay, which might be stored in either the correctionField of SYNC and/or FOLLOW_UP packets need to be updated too and a new value, depending on the rateRatio inserted/updated in the correctionField of SYNC and/or FOLLOW_UP packets. The devices in theses settings tend be in small, and detached networks, which is not connected to the Internet. For these reasons, having the rateRatio modified and inserted to either the SYNC or FOLLOW_UP does not have a huge impact. The closeness of the devices ensures that the rateRatio rarely changes and in this project though, assumed at first to be zero and does not change much. For these packet capture and modifications to be made possible, there needs to be an implementation regarding what packets to capture and in which way they are to be captured. The best solution in this regard is the use of Linux raw sockets. The reason for the choice of Linux raw sockets is because they provide all the headers related to a particular protocol, which then make for easier modifications.

3.5 Preliminary test results

The important synchronization results when both TSN modules have been started are offered by the DSTT, which receives the message from the NWTT, computes the residence time and the necessary corrections and reports on the delay. For this reason, from here on, when discussing the synchronization results achieved, these will be reported from the context of the DSTT. The results will be reported in number to display what is really happening at the DSTT over the duration of synchronization and report on the expected versus achieved behavior. The general principle on which the testing results are to be based on, is the fact that the GM has the most accurate time. Thereafter, when the synchronization messages are exchanged between the GM, and firstly, the NWTT, the NWTT clock after synchronization should be as close as possible (nano or microseconds accuracy). Messages that have been exchanged between the GM and NWTT are used to calculate the time offset between them and the GM then tries to correct the NWTT time. To this end, the NWTT time is now considered as close as possible to that of the GM even though it is a slave. The NWTT, then captures the same packets from which it was synchronized, and inserts the exact time (TSi), which is now accurate and based on hardware timestamping, into the packets. The packets are then masqueraded to make them appear as originating from the GM.

To demonstrate that the end result is achieved, which was in this scenario, to synchronize accurately with devices far from the GM, the TSN switch GUI offers the needed synchronization information used to verification purposes. From the results, it can be seen that the TSN switch has detected the GM based on the messages it received from the NWTT, and recorded the Best Master Priority 1 and 2, which are the GM configurations. The TSN switch too reports that the GM is one step away from it, which was the purpose intended by capturing the gPTP messages from the NWTT. The GM time and the local time seems to match with a time offset of 304372 nanoseconds reported (see Figure 8), which translates to around 304 microseconds. The reporting in some cases comes to 86 microsecond which is a good indication. The time synchronization approximation gets better when the code is left to run for some periods of time. The accuracy levels reached happen to average somewhere in tens of microseconds. In some cases, there are outliers, which occur once in a while.

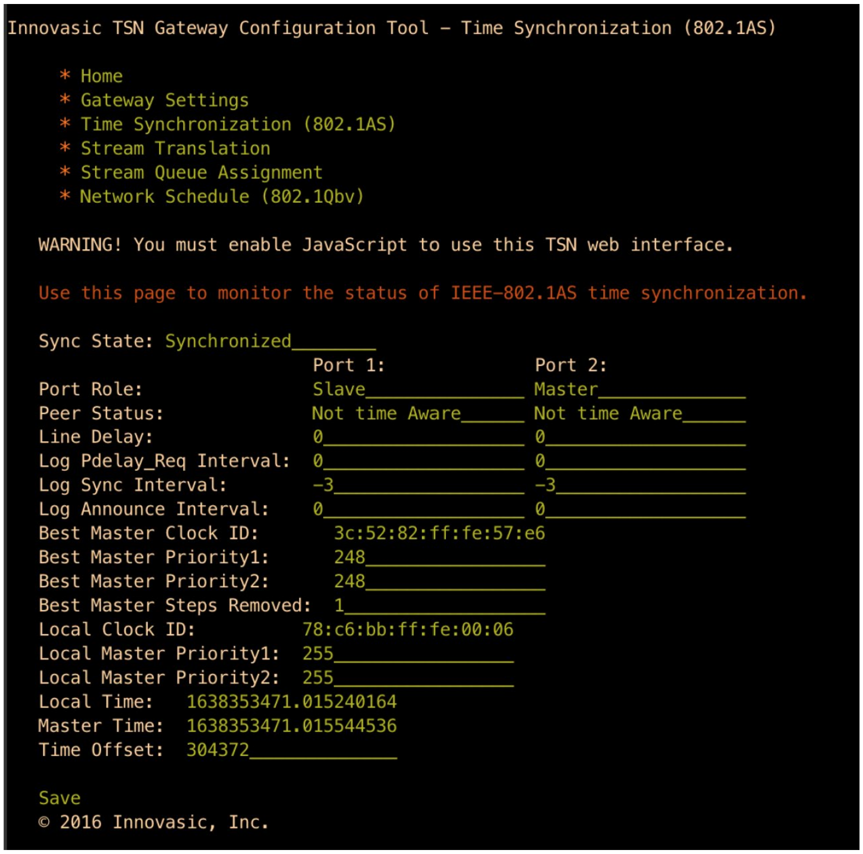

Figure 9: Preliminary TSN test results

3.6 5G-TSN device integration

Once these preliminary results permitted us to validate the overall TSN architecture and topology, the next step is the full integration of the 5G modem developed by Fivecomm for the project as part of the end-to-end architecture. It incorporates a 5G module and a control system based on Raspberry Pi 4 development board with ARM64 CPU, running a custom OS based on OpenWRT. The connection between these two components is realized via USB port and the external interface for TSN devices is Ethernet. Figure 8 shows the 5G modem overall architecture.

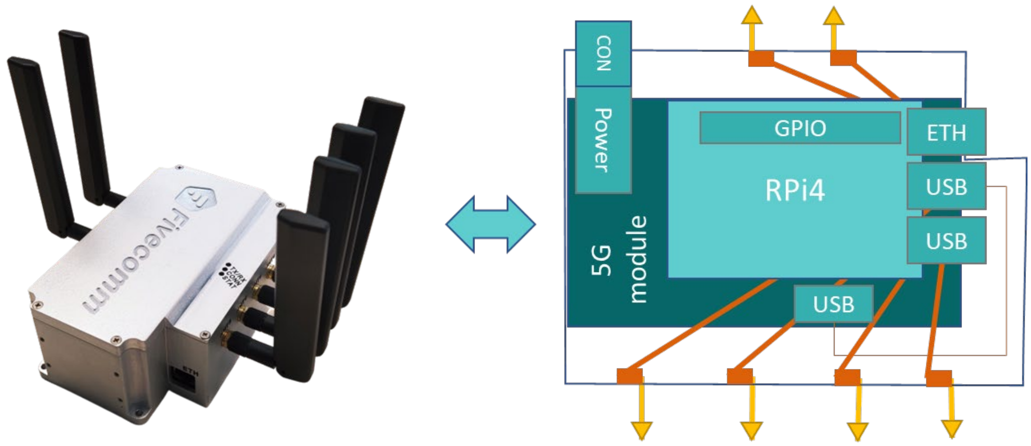

Figure 10: 5G modem to be used in FUDGE-5G for TSN demonstration.

The CPU from the Raspberry Pi is actually running DSTT software in order to ensure the TSN functionality workflow that was described in the subsection above. Two network interfaces are used for TSN implementation. The first interface is the WWAN to receive the gPTP packets from UPF through NWTT using the 5G radio link. Those packets are forwarded to the DSTT software that inserts egress time stamp (TSe) into the gPTP messages, after that the residence time is calculated and converted in GM time. In the end, this time is added to the gPTP packet and send to the TSN end node via LAN network interface. There are two possible options of timestamping: software timestamping and hardware timestamping.

- *Software timestamping* is ensured by Linux kernel from custom OS of Raspberry Pi using socket option of SO_TIMESTAMPING which requests RX timestamps when data enters the kernel. These timestamps are generated just after a device driver hands a packet to the kernel receive stack. This option of timestamping is currently implemented and will be tested in this project.
- *Hardware timestamping*, which requires additional hardware (timecard with precise hardware clock) for being available, but also provides a more precise time stamping. This option is still analysed because of the lack of suitable timecards that will correspond to the requirements needed and compatibility.

The planning in the coming months is to test the TSN functionality with the software timestamping and analyse the results. Meanwhile, partners will look for a solution for hardware timestamping, more exactly finding suitable additional hardware with very precise clock, as the software part of the hardware timestamp option is already implemented. Then test the hardware timestamping results and compare them to the one from software implementation. These functionalities will be tested in Cumucore lab.

4 Multicast Capabilities

4.1 Feature description

Multicast and broadcast delivery optimizes the network resources needed for transmitting the same content to a group of users/all the users in the same network [5].

With the increasing usage of multimedia applications, mobile networks would benefit from including this kind of transport into their networks. These transport mechanisms will save important network resources, enabling the possibility of handling a bigger number of users in downlink reception.

5G Multicast Broadcast Services (5G-MBS or 5MBS) is 3GPP's specifications for 5G regarding multicast and broadcast communications. These services provide a very important role enabling use cases involving hundreds or thousands of users, such as media streaming, IoT or massive software updates. But it could be used in a smaller LAN context too as an alternative for WiFi multicast, a technology not suitable for high-bandwidth applications, enabling the use of multicast for use cases with bigger throughput constraints.

The implementation shown in this deliverable follows 3GPP's technical specifications and it's compliant with it at the Core side. For the gNB and UE part, the implementation is not fully compliant with the standard, since we are working with an emulated gNB and UE.

The goal of these services is to perform multicast or broadcast delivery in mobile networks. To do this it is necessary to set up a multicast transport network between the UPFs and the gNBs on the N3 user plane interface and to perform point-to-multipoint (PTM) or point-to-point (PTP) delivery on the Uu interface at the gNBs, to reach the UEs on the radio network.

Although the standards present special cases and optional network functions, cases like designing the interoperability between multicast/broadcast capable network components and non-multicast/broadcast capable network components, the development within the FUDGE-5G project is focused only on working with the multicast/broadcast-capable components.

The selection of network functions from the architecture shown in the 3GPP technical specification 23.247 [6] is the following, the MB-SMF will be the network function in charge of exposing the services through the SBI, the MB-UPF will support the multicast delivery on the N3 interface and from the gNB interface to the UE comes the non-standard part. The NEF will expose the services provided at the MB-SMF to perform non-trusted application function (AF) access.

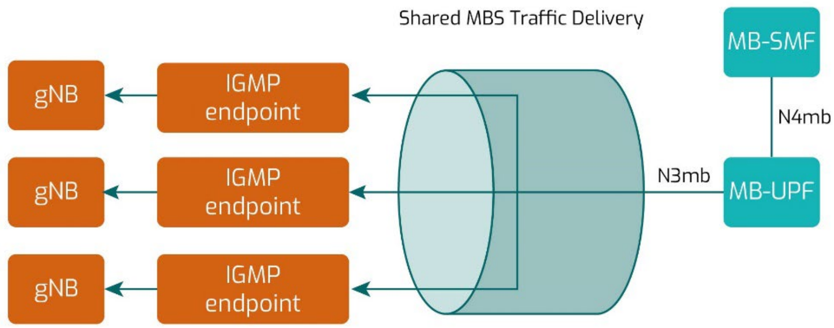

Figure 11: 5G-MBS implemented architecture

4.2 FUDGE-5G use case requirements

One of the biggest use cases regarding 5G-MBS is media streaming [7]. Throughput, latency and jitter are important parameters in use cases like video reception. The network should not only to be able to achieve the required throughput but to do it while maintaining a reasonable latency and jitter. Because of the nature of multicast/broadcast communications being a unidirectional transmission, only downlink parameters will be measured for the multicast implementation. We expect to be able to transmit a 1080p video, which uses a downlink bandwidth of around 5Mbps.

Another important use case is public warning use. In this case, the requirements are more about reliability, availability and scalability of the system.

4.3 Multicast implementation

The 5G-MBS implementation is developed on top of Fraunhofer FOKUS' Open5GCore. The existing release 6 network functions SMF, UPF and NEF have been modified to support the SBI services needed. The rest of the network functions present in Open5GCore are used without modifications, gNB and UE emulators present in Open5GCore have been modified to be able to perform end-to-end tests between the UPFs sending the multicast traffic and the UEs receiving it.

The APIs exposed by the network functions MB-SMF and NEF are the ones shown in 3GPP TS 29.532 [8] and in 3GPP TS 29.522 [9] respectively. These technical specifications define the TMGI management and MBS Session management services, needed to start the 5G-MBS procedures.

Table 1: MB-SMF services implemented

Service	Description	Operation
Nmbsmf_TMGI	Management of the TMGI identifier used in MBS processes.	Allocate
		Deallocate
Nmbsmf_MBSSession	Management of MBS Sessions for multicast and broadcast sessions.	Create
		Delete

Table 2: NEF MBS services implemented

Service	Description	Operation
Nnef_MBSTMGI	Exposure of the management of the TMGI identifier used in MBS processes.	Allocate
		Deallocate
Nnef_MBSSession	Exposure of the management of MBS Sessions for multicast and broadcast sessions.	Create
		Delete

By communicating with the MB-SMF (acting as a trusted AF) or with the NEF (acting as a non-trusted AF) APIs an MBS Session can be created, allocating the needed TMGI on the MB-SMF and activating the chain of communications needed to establish the N3 multicast tunnel between the MB-UPF and the gNBs.

The transmission method implemented between the MB-UPF and the gNBs corresponds to the Shared MBS Traffic Delivery specified in TS 23.247 [6] , avoiding the implementation of Individual MBS Traffic Delivery.

The MBS Session requires three essential parameters, a C-TEID for the GTP tunnel and two multicast addresses. The lower layer multicast address is used for the multicast group between the MB-UPF and the gNBs. The source multicast address is the destination address used by the server acting as source of multicast traffic [7].

The source multicast address is needed for the MB-UPF to perform its packet detection mechanism using the PDRs (Packet Detection Rules). This is configured by the MB-SMF using

PFCP in the N4 interface and lets the MB-UPF route the packets coming from the multicast source and send them to the gNBs through the N3 interface using multicast.

The gNB and UE emulators present at Open5GCore are not standard compliant. The only standard procedures are the ones between the gNBs and the 5G core, the messages sent and received by the gNB from the AMF and the UPF. The gNB and UE emulators work by using standard UDP communication, avoiding the need to implement all the protocol stack used in the radio network. Because of this, there is no particular interest in modifying the gNB source code to be able to receive multicast traffic just as a commercial multicast-compatible gNB would do.

To overcome this limitation in the current scenario, a tool capable to join multicast groups and forward the traffic to a unicast address has been developed. This tool receives the IP multicast group address to join the group (using IGMP) and the IP unicast address to forward the traffic to. With this tool, even commercial non-multicast gNBs could receive the traffic using unicast. For this to work, each gNB on the network wanting to join the multicast group needs to use an instance of this tool.

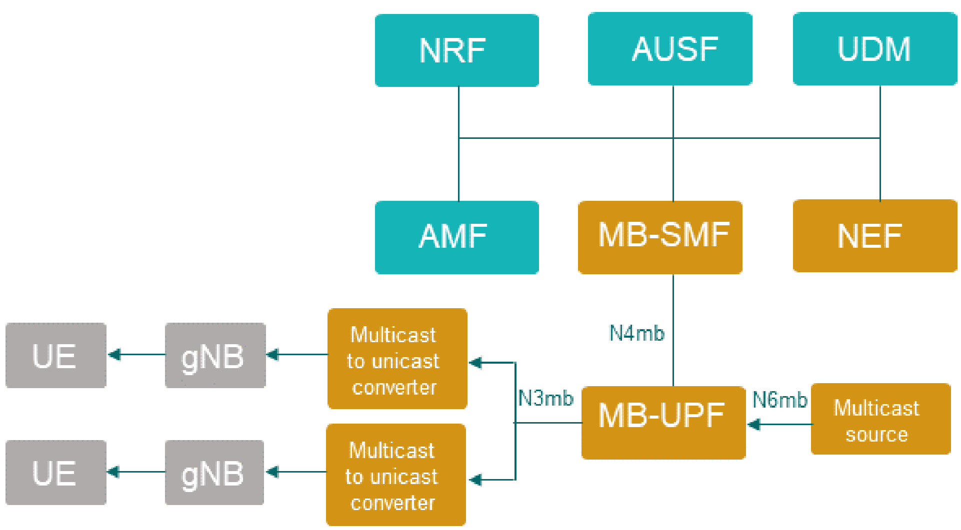

Figure 12: Multicast 5G Core architecture

This scenario could only be tested terminating the reception at the gNB if commercial gNBs are being used. To perform more interesting tests, a workaround has been developed to enable the UEs behind the gNBs to receive the multicast traffic. This will let us perform more interesting end-to-end tests.

A diagram of the TMGI allocation and MBS session creation can be found in Figure 12: TMGI allocation and MBS session creation call flow.

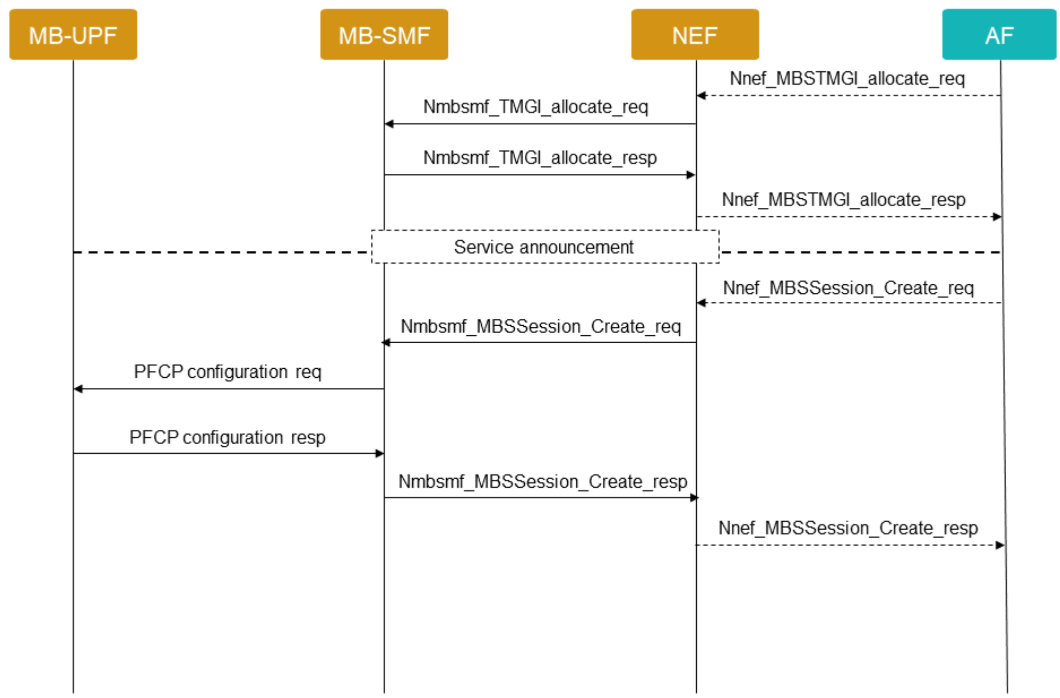

Figure 13: TMGI allocation and MBS session creation call flow

4.4 Used testing methods and equipment

4.4.1 Test cases

Two different testing scenarios will be prepared to evaluate the results of the implementation. These tests will be based on the tools iperf and ffmpeg, tools that have been crucial to validate the intermediate steps of this 5G-MBS implementation.

First test will consist of delivering multicast video using ffmpeg and receiving it on the UE side. This test will serve just as end-to-end system validation, proving that the system is capable to deliver a typical video.

For the second test, iperf will be used to perform multicast transmissions, extracting the KPIs from the tool itself, achieving the maximum throughput that the implementation can handle. This way the implementation will be benchmarked and after this, the first video transmission test will be performed again with a video with characteristics closer to the maximum supported by the implementation.

Test setup

The parameters that we need to decide for this test are:

- A TMGI, the C-TEID at the gNBs side of the GTP tunnel.

- The lower layer multicast address used for the data transmission from the UPF to the gNBs.
- The source specific multicast address used for the video transmission itself.

The test setup is the following:

1. Connect two gNBs to the UPF. Then attach one UE to each gNB. Once attached, we register the UEs to the 5G Core and we create a PDU session for each UE. For the gNBs to be able to receive the multicast traffic, they need to know which is the GTP tunnel that is going to be used to receive the multicast traffic. The C-TEID value (used to identify the GTP tunnel) is passed to the gNBs at startup configuration.
2. At the gNBs side, execute the multicast to unicast converter for each gNB. The multicast to unicast converter will join the multicast group identified by the lower layer multicast address using IGMP. Then, it will act as a reverse proxy that receives multicast traffic from the UPF and sends it to its gNB.
3. To be able to receive the multicast traffic at the UEs, the already established PDU session is used. Therefore, the only thing left to do at the UE is to join the source specific multicast group.
4. Last thing is to prepare the 5G Core for the multicast delivery. An MBS session is created at the SMF, passing the TMGI and the two multicast addresses as parameters. The SMF will then automatically create the needed PFCP rules in the UPF, thus configuring the data plane for the multicast delivery.

With everything configured, the next steps to be taken are test specific.

4.5 Test results

Test results will be published in D2.5 Technology Components and Platform – Final Release.

5 Conclusion

5GLAN testing was carried out successfully in the Telenor laboratory. TSN and multicast testing was done successfully in the separate laboratory sessions. Next step is to integrate 5G SA network with an ABB application in the ABB laboratory and have 5G LAN with TSN and Multicast features integrated into the system. This work will be reported in D2.5 Technology Components and Platform – Final Release

6 References

[1] 5G-ACIA, "Integration of 5G with Time-Sensitive Networking for Industrial Communications," 5G-ACIA, February 2021.

[2] FUDGE-5G, "D1.1 v2 Technical Blueprint for Vertical Use Cases and Validation Framework".

[3] L. Lo Bello and W. Steiner, "A perspective on IEEE time-sensitive networking for industrial communication and automation systems," *Proceedings of the IEEE,* vol. 107, no. 6, p. 1094–1120, 2019.

[4] Y. Seol, D. Hyeon, J. Min, M. Kim and J. Paek, "Timely survey of time sensitive networking: Past and future directions," *IEEE Access,* vol. 9, p. 142 506–142 527, 2021.

[5] FUDGE-5G, "D1.2 FUDGE-5G Platform Architecture: Components and Interfaces".

[6] 3GPP, "TS 23.247: Architectural enhancements for 5G multicast-broadcast services," [Online].

[7] A. Rico-Alvariño, I. Bouazizi, M. Griot, P. Kadiri, L. Liu and T. Stockhammer, "3GPP Rel-17 Extensions for 5G Media Delivery," *IEEE Transactions on Broadcasting,* vol. 68, pp. 422-438, June 2022.

[8] 3GPP, "TS 29.532: 5G System; 5G Multicast-Broadcast Session Management Services; Stage 3".

[9] 3GPP, "TS 29.522: 5G System; Network Exposure Function Northbound APIs; Stage 3".

[10] ITU-T, "Rec. E.212 Amendment 1," *07,* 2018.