
*Project funded by the Horizon 2020 Framework Program of the European Union,*

*Grant agreement Nº:* ***951947.***

Start date: **01/09/2020**

Duration: **48 months**

# D6.1. Test Cases Definition for the Small-Scale trials

| | |
|---|---|
| WP | WP6 |
| WP Leader | VEDECOM |
| Responsible Author | Julian Garbiso (VED), Chahrazed Ksouri (VED) |
| Contributors | All |
| Dissemination Level | PU |
| Nature | RE |

| Dissemination Level: | |
|---|---|
| PU | Public |
| PP | Restricted to other programme participants (Including the Commission Services) |
| RE | Restricted to a group specified by the consortium (Including the Commission Services) |

**PROPRIETARY RIGHTS STATEMENT**

This document contains information, which is proprietary to the 5GMED Consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to any third party, in whole or in parts, except with prior written consent of the 5GMED Consortium.

| | |
|---|---|
| CO | Confidential, only for members of the consortium (Including the Commission Services) |

| Nature | |
|---|---|
| PR | Prototype |
| RE | Report |
| SP | Specification |
| TO | Tool |
| OT | Other |

| | |
|---|---|
| Synopsis | Deliverable D6.1 provides a detailed definition of the test cases for the realization of the small-scale trials in all use cases. It also provides a detailed description of the service Key Performance Indicators (KPIs) measurement methodology, as well as the implementation of the common service KPI collection, storage, and visualization platform. The results of the small-scale trials will be reported in D6.2. |
| List of Keywords | Testing methodology, measurement methodology, data storage, data visualization, small-scale testbeds, test cases, service KPIs |

Funded by the Horizon 2020 Framework Programme of the European Union

## DOCUMENT HISTORY

| Version | Status[1] | Date | Comments | Author |
|---|---|---|---|---|
| 0.1 | Draft | 23/11/2022 | Creation of the ToC and initial contributions | Julian Garbiso |
| 0.9 | Draft | 19/07/2023 | Peer review version | Julian Garbiso |
| 1.0 | Draft | 15/09/2023 | Peer review version | Julian Garbiso, Chahrazed ksouri |
| 2 | Issued | 20/10/2023 | Ready for submission | Chahrazed ksouri |

## AUTHORS


| Authors in alphabetical order | | |
|---|---|---|
| Name | Organization | Email |
| Angelos Antonopoulos | NBC | aantonopoulos@nearbycomputing.com |
| Aniello Cammarano | AXBRYD | cammarano@axbryd.com |
| Antón Aguilar | CTTC | anton.aguilar@cttc.es |
| Chahrazed Ksouri | VEDECOM | Chahrazed.Ksouri@vedecom.fr |
| David Carrera | NBC | dcarrera@nearbycomputing.com |
| David Porcuna | AAE | david.porcuna@autopistas.com |


[1] Status (a status is associated to each step of the document life cycle)
Draft. This version is under development by one or several partner(s)
Under review. This version has been sent for review
Issued. This version of the document has been submitted to EC

| Authors in alphabetical order | | |
|---|---|---|
| Name | Organization | Email |
| Francisco Vázquez-Gallego | i2CAT | francisco.vazquez@i2cat. |
| George Triantafyllou | ATC | g.triantafyllou@atc.gr |
| Herve Marcasuzaa | VALEO | herve.marcasuzaa@valeo.com |
| Juan Agustí | COMSA | juan.agustimoreno@comsa.com |
| Judit Bastida Raja | CELLNEX | judit.bastida@cellnextelecom.com |
| Julian Garbiso | VEDECOM | julian.garbiso@vedecom.fr |
| Kévin Nguyen | VALEO | kevin.nguyen@valeo.com |
| Luigi Girletti | ATOS | luigi.girletti.external@atos.net |
| Luis Miguel Quintero | CELLNEX | luis.miguel.quintero@cellnextelecom.com |
| Mickael Messias | VALEO | mickael.messias@valeo.com |
| Montserrat Anglès | AAE | manglesc@autopistas.com |
| Nuria Trujillo | HSP | ntrujillo@hispasat.es |
| Oscar Trullols | NBC | otrullols@nearbycomputing.com |
| Philippe Seguret | VALEO | philippe.seguret@valeo.com |
| Philippe Veyssiere | IRT | philippe.veyssiere@irt-saintexupery.com |
| Ramsundar Kandasamy | VALEO | ramsundar.kandasamy@valeo.com |
| Raul Gonzalez Prats | CELLNEX | raul.gonzalez@cellnextelecom.com |
| Ricard Vilalta | CTTC | ricard.vilalta@cttc.es |
| Rodrigo Peces | ATOS | rodrigo.peces@atos.net |
| Spyros Papafragkos | ATC | s.papafragkos@atc.gr |
| Sylviane Roullier | SNCF | sylviane.roullier@sncf.fr |
| Thanasis Dalianis | ATC | t.dalianis@atc.gr |

# TABLE OF CONTENTS

# LIST OF ABBREVIATIONS

| | |
|---|---|
| **ACS-GW** | Adaptative Communication System-Gateway |
| **ADAS** | Advanced Driver Assistance Systems |
| **AI** | Artificial Intelligence |
| **AID** | Automatic Incident Detection |
| **CA** | Consortium Agreement |
| **CAV** | Connected and Autonomous Vehicle |
| **CCAM** | Cooperative, Connected and Automated Mobility |
| **CIM** | Container Infrastructure Manager |
| **CV** | Connected Vehicle |
| **C-V2X** | Cellular-V2X |
| **DDT** | Dynamic Driving Task |
| **DHCP** | Dynamic Host Configuration Protocol |
| **DoA** | Description of Action |
| **EC** | European Commission |
| **DENM** | Decentralized Environmental Notification Messages |
| **EMT** | Enjoy Media Together |
| **ETSI** | European Telecommunications Standards Institute |
| **FRMCS** | Future Railway Mobile Communication System |
| **FSTP** | Financial Support to Third Parties |
| **GA** | Grant Agreement |
| **GPS** | Global Positioning System |
| **H2020** | Horizon 2020 |
| **HD** | High Definition |
| **HMI** | Human Machine Interface |
| **HRR** | Home Routed Roaming |
| **I2V** | Infrastructure-to-Vehicle |
| **IOPS** | Input/output Operations Per Second |
| **IoT** | Internet of Things |
| **ITS** | Intelligent Transportation System |
| **KPI** | Key Performance Indicator |
| **LAN** | Local Area Network |
| **LFP** | Linea Figueras - Perpignan |
| **MANO** | Management and Network Orchestration |
| **MCM** | Maneuver Coordination Message |
| **MEC** | Mobile Edge Computing |
| **MNO** | Mobile Network Operator |
| **MRM** | Minimum Risk Maneuver |
| **NR** | New Radio |
| **ODD** | Operational Design Domain |
| **POI** | Point of Interest |
| **QoE** | Quality of Experience |
| **QoS** | Quality of Service |

| | |
|---|---|
| **R&D** | Research and Development |
| **RAN** | Radio Access Network |
| **REM** | Relay of emergency messages |
| **RSU** | Road-Side Unit |
| **RV/RS** | Remote Vehicle / Remote Station |
| **SIM** | Subscriber Identity Module |
| **SLA** | Service Level Agreement |
| **SME** | Small and medium-sized enterprise |
| **SNR** | Signal to Noise Ratio |
| **TC** | Test Case |
| **TCU** | Telematic Control Unit |
| **TMC** | Traffic Management Centre |
| **ToD** | Teleoperated Driving |
| **UC** | Use Case |
| **UE** | User Equipment |
| **VA** | Video Analytics |
| **V2I** | Vehicle-to-Infrastructure |
| **V2X** | Vehicle-to-Everything |
| **VDI** | Virtual Desktop infrastructure |
| **vRAN** | Virtual Radio Access Network |
| **VTC** | Valeo Teleoperation Cloud |
| **WLAN** | Wireless LAN |

Funded by the Horizon 2020 Framework Programme of the European Union

# LIST OF TABLES



# LIST OF FIGURES

Funded by the Horizon 2020 Framework Programme of the European Union

# EXECUTIVE SUMMARY

This document outlines the foundational framework for conducting small-scale trials within the 5GMED project. The primary objective of these trials is to validate the services and evaluate the performance of all four 5GMED use cases by ensuring they meet predefined service Key Performance Indicator (KPI) values. This deliverable establishes the groundwork by defining measurement methodologies for the service KPIs introduced in D2.1, necessary metrics, measurement procedures, and KPI calculation methods. The focus of these service KPIs lies in assessing cross-border quality of service, encompassing service-level metrics such as end-to-end latencies, throughputs, reliability, and interruption times, among others.

Once the service KPI measurement methodologies are established, the document offers comprehensive insights into the common service KPI collection, storage, and visualization platform developed in 5GMED. This platform, hosted on the 5GMED Cloud, utilizes connected MongoDB and Grafana instances. It is directly accessible by vehicles, road infrastructure, and cloud services for reporting metrics required for KPI calculation. Database structures for each use case are detailed, presenting all the fields and metrics essential for calculating KPIs. Additionally, synchronization methodology for both ground and mobile equipment is described, emphasizing the critical role of GPS-based Network Time Protocol (NTP) servers in ensuring the measurement validity of one-way latencies.

The document concludes by presenting detailed test cases for the small-scale trials of each 5GMED use case. These detailed test cases are an extension of the preliminary test cases defined in D2.2 for the performance evaluation of the use cases in the cross-border scenario. For each test case, the test case type, purpose, and pre-conditions are clearly defined; and a checklist is provided, listing the necessary equipment (including vehicles and network elements) and the personnel responsible for their operation, if applicable. The core of the test case definition comprises ordered and detailed scripts outlining all actions to be executed. Following the test steps, the measurements transmitted to the common database are specified, along with the test case's success criteria, which must be met to validate the test case's purpose. The realization of the small-scale trials is a crucial step in assessing the performance of the 5GMED cross-border network to meet the requirements of Connected and Automated Mobility and railway use cases operating in cross-border scenarios.

Funded by the Horizon 2020 Framework Programme of the European Union

# 1. Introduction

## 1.1 Objective of the Document

The present document is the first delivered version of D6.1 “Test cases definition for the small-scale trials”, which is part of WP6. The aim of this deliverable is to provide a detailed definition of the 5GMED small-scale trials. Furthermore, it offers an in-depth understanding of the methodology for measuring service Key Performance Indicators (KPIs), along with the establishment of a unified platform for collecting, storing, and visualizing service KPIs. The results of the execution of the test cases outlined in this document will be documented in D6.2.

## 1.2 Structure of the Document

This document is organized as follows. Section 2 presents the service KPIs that were defined in D2.1 for each one of the use cases (UC). Section 3 provides a detailed description of the service KPI platform implementation, including the KPI platform architecture, the database structures, and the ground and mobile equipment synchronization. From Section 4 to Section 7, it presents the test cases defined for use case 1 to 4, respectively, including one the map of the test site and detailing the following information for each test case: the type, pre-conditions, checklist, objective, required equipment, a step-by-step procedure, the measurements collected and transmitted to the database as well as the results and success criteria. Finally, Section 8 concludes the document.

# 2. Service KPI Measurement Methodology

This section presents the service KPI measurement methodologies for each use case, based on the list of service KPIs provided in D2.1 [1]. For each service KPI, the measurement methodology includes the following information:

- The measurement parameters and the target values (defined in D2.1) for the service KPI.
- The measurement procedure, describing the overall method across all the components of the use case deployed on the network and mobile equipment.
- The expression of the formula for the computation of the service KPI (if applies), as a function of the measurement parameters, and detailing the components of the use case where the measurement parameters are collected and where the computation of the service KPI takes place.

The high-level functional architecture of each use case and its main functional components are introduced in D2.1.

## 2.1 Use Case 1

The following table summarizes the measurement methodology of the service KPIs of UC1, including the target values, the measurement parameters, the measurement procedures, and the service KPI calculation methods.

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute? |
|---|---|---|---|
| **Command End-to-End Latency**<br>Target value:<br>20 ms | Timestamp | Each action on the steering wheel of the remote station has a timestamp which is sent with the command to measure the command end-to-end latency | The remote vehicle subtracts the attached timestamp to the current timestamp. |
| **Data End-to-End Latency**<br>Target value:<br>100 ms | Timestamp | Each sensors data packet sent from the remote vehicle to the remote station has a timestamp which is sent with the sensors data. | The remote vehicle subtracts the attached timestamp to the current timestamp. |
| **Command Reliability**<br>Target value:<br>99% | Number of command packets | Computed as the ratio between the number of commands packets received by the remote vehicle and the number of command packets transmitted by the remote station. | In the autonomous supervisor of the remote vehicle. |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute? |
|---|---|---|---|
| **Sensing Reliability** Target value: 95% | Number of sensors data packets | Computed as the ratio between the number of sensors data packets transmitted by the remote vehicle and the ones received by the remote station. | The remote vehicle retrieves the number of sensors data packets lost from the Teleoperation Cloud |
| **Uplink Service Data-Rate** Target value: 100 Mb/s | - | The TCU in the remote vehicle tests the current network access | The amount of data uploaded to the Valeo Teleoperation Cloud within a predefined time window will be compared to effective data transmitted and requested by the remote vehicle. |
| **Downlink Service Data-Rate** Target value: 1 Mb/s | - | The TCU in the remote vehicle test the current network access | The received data in the vehicle within a predefined time window will be compared to the expected transmitted data by the Valeo Teleoperation Cloud. |
| **Mobility interruption time** Target value: < 100 ms | - | The TCU in the remote vehicle tracks the throughput and when it drops to zero, the remote vehicle starts a timer. | The remote vehicle counts the time elapses with no throughput. |

*Table 1: Service KPIs, measurement methodology and tools for UC1*

## 2.2 Use Case 2

Before presenting the service KPI measurement methodologies for UC2, this section provides additional information about how both “hazard end-to-end latency” (Figure 1) and “traffic regulation end-to-end latency” (Figure 2), defined in D2.1, are measured.

**Hazard end-to-end latency**

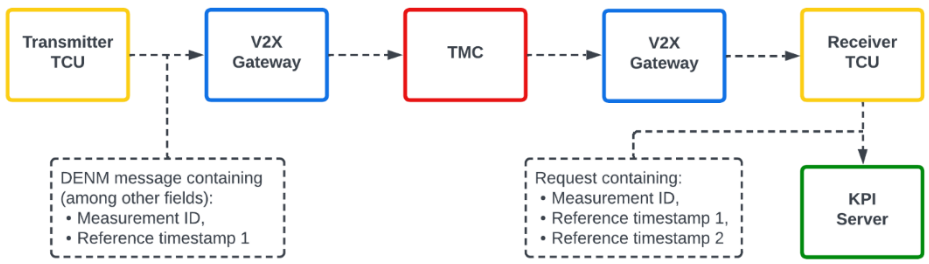


*Figure 1: Scheme of the methodology to measure the hazard end-to-end latency of UC2*

The hazard end-to-end latency measures the time elapsed between the detection of an obstacle in the road and the corresponding hazard notification received by a vehicle. Thus, a single measurement needs the following parameters: a unique identifier and 2 timestamps (one for the time of detection and another for the time when the hazard notification arrives). The latency can be calculated as the difference between the 2 timestamps. In addition, the same approach will be used to split the end-to-end latency between components. Each TCU receiver and the V2X Gateways will send to the MongoDB: stationID of the TCU or V2X Gateway, stationID of the TCU that transmits the message, delay, Reception Timestamp, and V2X Gateway_ID where the TCU is connected.

**Traffic Regulation End-To-End Latency**

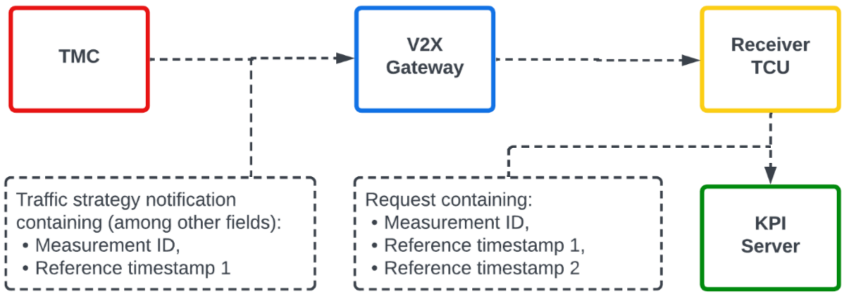


*Figure 2: Scheme of the methodology to measure the Traffic Regulation End-To-End Latency of UC2*

The Traffic Regulation End-To-End Latency measures the time elapsed between the transmission of a traffic strategy notification by the Traffic Management Center (TMC) and the reception of the traffic strategy notification by a vehicle. Thus, a single measurement needs a unique identifier and 2 timestamps (one for the time of transmission and another for the time when the notification arrives at the TCU). The latency can be calculated as the difference between the 2 timestamps.

The following table summarizes the measurement methodology of the service KPIs of UC2, including the target values, the measurement parameters, the measurement procedures, and the service KPI calculation methods.

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute? |
|---|---|---|---|
| **Hazard end-to-end latency**<br>Target value: Max. 200 ms | **Reference Timestamp 1:** It represents the time when an obstacle is detected on the road. The format of this parameter corresponds to the ETSI time format. The DENM transmission message includes this timestamp.<br><br>**Reference Timestamp 2:** It represents the time when the hazard notification (generated due to the prior obstacle detection) arrives to a vehicle. The format of this parameter corresponds to the ETSI time format.<br><br>**Measurement ID:** It identifies a unique measurement taken from an obstacle detection to the corresponding final hazard notification. Its value is defined as the Station ID (TCU Identifier) + Sequence Number. | When an obstacle is detected, the **transmitter application in the TCU** encapsulates the **Reference timestamp 1** and the **Measurement ID** in the DENM message transmitted. These 2 parameters will be carried all the way, until the receiver application in the TCU receives the corresponding hazard notification (through a DENM message). Then, the TCU will generate the **Reference Timestamp 2** and send all 3 parameters to the KPI server. | The latency calculation is made in the receiver TCU as Reference Timestamp 2 – Reference Timestamp 1 |
| **Traffic Regulation End-To-End Latency**<br>Target value: Max. 200 ms | **Reference Timestamp 1:** It represents the time when traffic strategy notification is transmitted by the TMC. The format of this parameter corresponds to the ETSI time format.<br><br>**Reference Timestamp 2:** It represents the time when the traffic strategy notification is received by a vehicle. The format of this parameter | The traffic strategy notification message encapsulates the **Reference timestamp 1** and the **Measurement ID**. These 2 parameters will be carried all the way, until the receiver application in the TCU receives the traffic strategy notification. Then, the TCU will generate the **Reference Timestamp 2** and send all 3 parameters to the KPI server. | The latency calculation will be made in the receiver TCU as Reference Time 2 – Reference Time 1 |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute? |
|---|---|---|---|
| | corresponds to the ETSI time format.<br><br>**Measurement ID:**<br>It identifies a unique measurement taken from a traffic strategy notification. Its value corresponds to a unique Message ID that ensures that no 2 measurements will have the same identifier | | |
| **Hazard Notification Reliability**<br>Target value: 100% | **StationID:** The hazard notification message (DENM) message includes the ITS Station identifier | Number of DENM notifications with the same StationID received and transmitted counter. It must insert in the value to the MongoDB.<br>The V2X Gateway counts the number of each DENM message it sends and receives. | Calculated on the KPI server as the ratio between the number of hazard notification messages received by a TCU over the number of hazard notification messages transmitted by another TCU. |
| **Traffic Regulation Reliability**<br>Target value: 100% | **StationID** included in the traffic strategy notification message (MCM)<br>**MessageID:** Unique identifier of the message<br>**TargetStationID:** StationId for the destination of the message | Both V2X Gateway and TCU maintain a count of the sent and received messages (MCM) with their unique identifiers.<br>• V2X Gateway uploads in the KPI server the ID of send messages for each TCU Station IDs<br>• TCU upload in the KPI service the ID of received message from V2X Gateway, identified by a unique StationID | Calculated on the KPI server as the ratio between the number of received messages over the number of sent messages for messages sent by V2X Gateway and received by TCU. |
| **Mobility Interruption Time**<br>Target value: < 100 ms | **A**: Time when the first lost packet was acknowledged as lost.<br><br>**B**: Time when the first received packet arrives after a packet was lost. | The TCU sends one-packet pings periodically to the V2X Gateway and, if no response is received, then mobility interruption time starts counting. | **B – A**<br>Calculated by the TCU |

*Table 2: Service KPIs, measurement methodology and tools for UC2*

## 2.3 Use Case 3

The following table summarizes the measurement methodology of the service KPIs of UC3 for service B1, service B2, service P1, and service P2 defined in D2.1. The measurement methodology including the target values, the measurement parameters, the measurement procedures, and the service KPI calculation methods.

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| **B1 uplink data rate**<br>Target value:<br>> 100 Mb/s<br>< 750 Mb/s | Throughput measurement results | Execute iPerf3 in reverse TCP mode with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as an iperf3 server. | Measured amount of data sent uplink per second, as implemented in https://github.com/esnet/iperf. Calculated by the **iperf3 client**. |
| **B1 downlink data rate**<br>Target value:<br>> 200 Mb/s<br>< 750 Mb/s | Throughput measurement results | Execute iPerf3 in TCP mode with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as an iperf3 server. | Measured amount of data sent downlink per second, As implemented in https://github.com/esnet/iperf. Calculated by the **iperf3 client**. |
| **B1 uplink end-to-end latency**<br>Target value:<br>< 100 ms | One-way latency measurement in the uplink | Execute OWPING with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as a server while **both PC**'s time is synchronized by an **NTP server**. | Measured time it takes for a packet to go from client to server, as implemented in https://github.com/perfsonar/owamp. Calculated by the **OWPING application**. |
| **B1 downlink end-to-end latency**<br>Target value:<br>< 100 ms | One-way latency measurement in the downlink | Execute OWPING with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as a server while **both PC**'s time is synchronized by an **NTP server**. | Measured time it takes for a packet to go from server to client, as implemented in https://github.com/perfsonar/owamp. Calculated by the **OWPING application**. |
| **B1 uplink reliability**<br>Target value:<br>≥ 97% | **A:** number of packets received by a server<br>**B:** number of packets sent by a client | Execute iPerf3 in reverse UDP mode with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as an iperf3 server. | Ratio between the number of packets received by the server over packets sent by the client (A/B), as implemented in https://github.com/esnet/iperf. Calculated by the **iPerf3 application**. |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| **B1 downlink reliability**<br>Target value:<br>≥ 98% | **A**: number of packets received by a server<br>**B:** number of packets sent by a client | Execute iPerf3 in reverse UDP mode with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as an iperf3 server. | **A**: number of packets received by the client over packets sent by the server, as implemented in https://github.com/esnet/iperf.<br>Calculated by the **iPerf3 application**. |
| **B1 uplink jitter**<br>Target value:<br>< 5 ms | Jitter measurement results | Execute iPerf3 in reverse UDP mode with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as an iperf3 server. | Measured as average time difference between each packet sequence sent by the client, as implemented in https://github.com/esnet/iperf.<br>Calculated by the **receiver application**. |
| **B1 downlink jitter**<br>Target value:<br>< 5 ms | Jitter measurement results | Execute iPerf3 in reverse UDP mode with the **PC-NUC in the train** as a client and the **PC-NUC on ground** as an iperf3 server. | Measured as average time difference between each packet sequence sent by the server, as implemented in https://github.com/esnet/iperf.<br>Calculated by the **receiver application**. |
| **B1 Mobility Interruption Time**<br>Target value:<br>< 10 s | **A**: Time when the first lost packet was acknowledged as lost.<br><br>**B**: Time when the first received packet arrives after a packet was lost. | Execute a script whereby the **PC-NUC in the train** sends one-packet pings every half second to the **PC-NUC on ground** and, if no response is received, then mobility interruption time starts counting. | **B – A**<br>Calculated by the **receiver application** |
| **KPI** | **Measurement parameters** | **Measurement procedure** | **How and where to compute the KPI** |
| **P1 uplink data rate**<br>Target value:<br>> 5 Mb/s<br>< 6 Mb/s | Number of received TCP datagrams | Count the number of received TCP datagrams over one second | At the receiver side, multiply the number of received TCP datagrams over one second by the size of the TCP datagram |
| **P1 downlink data rate**<br>Target value:<br>5 Mb/s | Number of received TCP datagrams | Count the number of received TCP datagrams over one second | At the receiver side (the train presentation tool), multiply the number of received TCP datagrams over one second by the size of the TCP datagram<br>Note: the train presentation tool is set on board to “exercise” the |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| | | | downlink connection, but will not in reality be onboard |
| **P1 uplink Cloud End-to-End Latency**<br>Target value: < 1 s | **A**: time at which a TCP datagram is transmitted from the client<br>**B**: time when a TCP datagram is received in the server | When an TCP datagram is sent, the **transmitter application** encapsulates the **timestamp A**<br>When the server receives the TCP datagram, it generates the **timestamp B** | **B**-**A** calculated by the receiver at the server |
| **P1 downlink Cloud End-to-End Latency**<br>Target value: < 1 s | **A**: time at which a TCP datagram is transmitted from the server<br>**B:** time when a TCP datagram is received by the client on-board | When an TCP datagram is sent, the **transmitter application** encapsulates the **timestamp A**<br>When the server receives the TCP datagram, it generates the **timestamp B** | **B**-**A** calculated by the receiver at the client |
| **P1 uplink reliability**<br>Target value: ≥99.99% | **A**: number of TCP datagrams that are received by the server<br>**B**: number of TCP datagrams that have been transmitted by the client | The client and server count the number of TCP datagrams B and A, respectively | **A**/**B***100 |
| **P1 downlink reliability**<br>Target value: ≥99.99% | **A**: number of TCP datagrams that are received by the client<br>**B**: number of TCP datagrams that have been transmitted by the server | The client and server count the number of TCP datagrams A and B, respectively | **A**/**B***100 |
| **P1 Mobility Interruption Time**<br>Target value: < 1 s | **A**: Time when the first lost packet was acknowledged as lost.<br>**B**: Time when the first received packet | A timer is started when a TCP datagram is received | For each received UDP datagram on the Cloud, we store the timestamp of its reception. Then, if the time difference between the reception of 2 UDP datagrams is greater than 400ms, we consider that a Mobility Interruption event |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| | arrives after a packet was lost. | | occurred. The calculated Mobility Interruption Time is the time difference between the reception of the last UDP datagram before the Mobility Interruption event and the first one after the event. |
| **KPI** | **Measurement parameters** | **Measurement procedure** | **How and where to compute the KPI** |
| **P2 Uplink data rate**<br>Target value:<br>> 5 Mb/s<br>< 20 Mb/s | Number of data bits transmitted by the LiDAR | Execute Psutil Python library. | Compute the amount of data received on the network interface of the LiDAR per second |
| **P2 Uplink End-to-End Edge Latency**<br>Target value:<br>< 200 ms | Round-trip time measurement results | Execute a ping command | Get the latency value from the ping command response |
| **P2 Downlink End-to-End Edge Latency**<br>Target value:<br>< 200 ms | | | |
| **P2 Uplink Reliability**<br>Target value:<br>≥99.99% | **A**: number of packets that are received by the P2 Edge Module<br>**B**: number of packets that have been transmitted by the P2 train module | The P2 Edge Module and the P2 train module count the number of packets A and B, respectively | **A**/**B***100 |
| **P2 Downlink Reliability**<br>Target value:<br>≥99.99% | **A**: number of packets that are received by the P2 train Module<br>**B**: number of packets that have been transmitted by the P2 Edge module | The P2 Edge Module and the P2 train module count the number of packets A and B, respectively | **A**/**B***100 |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| **P2 Mobility Interruption time**<br>Target value: < 1s | **A**: Time when the first lost packet was acknowledged as lost.<br>**B**: Time when the first received packet arrives after a packet was lost. | A timer is started when a TCP datagram is received | For each received UDP datagram on the Cloud, we store the timestamp of its reception. Then, if the time difference between the reception of 2 UDP datagrams is greater than 500ms, we consider that a Mobility Interruption event occurred. The calculated Mobility Interruption Time is the time difference between the reception of the last UDP datagram before the Mobility Interruption event and the first one after the event. |
| **KPI** | **Measurement parameters** | **Measurement procedure** | **How and where to compute the KPI** |
| **B2 uplink data rate**<br>Target value:<br>> 0.6 Mb/s<br>< 3.8 MB/s | Amount of received data on the upfgtp interface on the UPF of the 5G Core of the B2 Service. | Compute the amount of data received on the upfgtp interface on the UPF every one second. | Execute Python code that implements the measurement procedure on the UPF machine. |
| **B2 downlink data rate**<br>Target value:<br>> 0.6 Mb/s<br>< 3 MB/s | Amount of transmitted data on the upfgtp interface on the UPF of the 5G Core of the B2 Service. | Compute the amount of data transmitted on the upfgtp interface on the UPF every one second. | Execute Python code that implements the measurement procedure on the UPF machine |
| **B2 uplink reliability**<br>Target value:<br>> 97% | Number of GTP datagrams sent.<br>Number of GTP datagrams received. | Subtraction of sent and received packets. Count the percentage of packets correctly received by the UPF on ground. | The python code is executed on the small cell gNB (Amarisoft) and the UPF machine. The code uses Scapy library to sniff the packets going in and out on the GTP protocol of both the small cell and the UPF machine. From Scapy, we get the packet id number that is sent from the gNB. We then look if this packet is received in the UPF. |
| **B2 downlink reliability**<br>Target value:<br>> 97% | Number of GTP datagrams received. | Subtraction of sent and received packets. Count the percentage of packets correctly | The python code is executed on the small cell gNB (Amarisoft) and the UPF machine. The code uses Scapy library to sniff the |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| | Number of GTP datagrams sent. | received by the small cell gNB in train. | packets going in and out on the GTP protocol of both the small cell and the UPF machine. From Scapy, we get the packet id number that is sent from the UPF. We then look if this packet is received in the gNB. |
| **B2 uplink jitter**<br>Target value: < 40 s | Latency measurements | Difference between highest and lowest uplink latency calculated every time we send/receive GTP packets. | Execute Python code that implements the measurement procedure on the UPF machine |
| **B2 downlink jitter**<br>Target value: < 40 s | Latency measurements | Difference between highest and lowest downlink latency calculated every time we sent/receive GTP packets. | Execute Python code that implements the measurement procedure on the UPF machine |
| **B2 mobility interruption time**<br>Target value: < 1s | **A**: Time when the first lost packet was acknowledged as lost.<br>**B**: Time when the first received packet arrives after a packet was lost. | Execute a script whereby the small cell gNB (Amarisoft) in the train sends one-packet pings every half second to the UPF machine and, if no response is received, then mobility interruption time starts counting. | **B – A**<br>Calculated by a Python code in the small cell |
| **B2 End-to-End latency between UE's using the 5G small-cell on-board**<br>Target value: < 200 ms | **A/B**: time at which UDP packets on the GTP protocol are sent and received on the small cell gNB/UPF machine. | Execute Python code on the small cell gNB (Amarisoft) and the UPF machine. The code uses Scapy library to sniff the packets going in and out on the GTP protocol of both the small cell and the UPF machine. | **A/B**: Calculate the difference in time when the packet is sent and received on either the small cell or the UPF machine.<br>We differentiate between uplink and downlink latency from the source and destination ip addresses. |

*Table 3: Service KPIs, measurement methodology and tools for UC3*

## 2.4 Use Case 4

The following table summarizes the measurement methodology of the service KPIs of UC4, including the target values, the measurement parameters, the measurement procedures, and the service KPI calculation methods.

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| **End-to-end latency**<br>Target value:<br>1) Video Streaming: <4s<br>2) Video Conferencing: < 20–100 ms | **A:** App latency<br>**B**: UL latency<br>**C**: DL latency | End-to-end latency value is calculated by adding **A**, **B** and **C** values.<br>**A**: time that the application takes to process a video packet.<br>**B**: Uplink latency. Time that a request takes to arrive from the UE to the Edge.<br>**C**: Downlink latency. Time that a response takes to arrive from the Edge to the UE. | Custom Implementation in the EMT Web Client. Implemented out of band mechanism to collect uplink and down link latency. This is added to the application latency collected from the Player-hls.js library.<br>**A+B+C** |
| **Data rate**<br>Target value:<br>1) Video Streaming: >20Mbps<br>2) Video Conferencing: >1Mbps | **A:** Bandwidth estimate | Measure effective application bandwidth estimate in bps. | EMT Web Client Video Player. Open source, Player-hls.js library built-in analytics that provides an estimate based on the download of video chunks. |
| **Jitter**<br>Target value:<br>1) Video Streaming: <0.4 s<br>2) Video Conferencing: <2ms | **A:** App latency<br>**B**: UL latency<br>**C**: DL latency | Absolute value (abs) of the difference between sample values (deriv) in a range of 2 seconds, ([2s]).<br>Metric is collected every second so there will be 2 measurements in 2 seconds. | Calculated on Open Source, Grafana dashboard.<br>Uplink latency jitter: $abs(deriv(B[2s]))$<br>Downlink latency jitter: Formula **a)** below |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| | **a)** $sumby(user)\left(abs(deriv(A[2s])) + abs(deriv(C[2s]))\right)$ | | |
| **Framerate**<br>Target value:<br>1) Video Streaming: >30 fps<br>2) Video Conferencing: >30 fps | **A**: Number of frames rendered.<br>**B**: Seconds passed. | Frames per second. Calculated by taking the number of frames rendered and dividing it by the seconds passed. | EMT Web Client Video Player. Open source, Player-hls.js library built-in analytics.<br>**A / B** |
| **Mobility interruption time**<br>Target value:<br>1) Video Streaming: <1s<br>2) Video Conferencing: <100ms | **A**: Timeframe that UE browser detects no network connection. | Amount of time the service goes offline. | EMT Web Client. Custom implementation within the EMT Web Client counting the number of seconds that the service connection is down. |
| **Reliability**<br>Target value:<br>99.9% | **A:** Frames successfully delivered.<br>**B**: Total of corrupted and dropped frames sent to the player.<br>**C**: Total of frames sent to the player. | Percentage of frames successfully delivered to the player divided by the total number of frames sent.<br>Frames successfully delivered: number of total frames sent minus the total of corrupted and dropped frames. | EMT Web Client, Open source, Player-hls.js library built-in analytics.<br>**A = C – B**<br>**A / C** * 100 |
| **Service migration time**<br>Target value:<br>20 – 35 s | **A**: Timestamp generated by the Decision Engine on migration flag.<br>**B**: Timestamp generated by EMT Edge when the service is ready on the new Edge.<br>**C**: Time elapsed from new migration is | Time required by the Orchestrator to migrate the EMT Edge service components from one edge node to another. | Execution of the migration from the moment the Decision Engine flags for new migration until the containers are ready on the target Edge server.<br>**C = B - A** |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI? |
|---|---|---|---|
| | triggered, to the EMT Edge server is up and running. | | |

*Table 4: Service KPIs, measurement methodology and tools for EMT service of UC4*

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI |
|---|---|---|---|
| **End-to-end latency**<br>Target value:<br>1) High resolution media: <1 s<br>2) 360-degree high resolution media: < 20-100ms<br>3) Immersive media: 20-80 ms | **A**: application-layer RTT measurements across recently active connections. This value is based on recently observed | The NetworkInformation.rtt is a Web API that returns the estimated effective round-trip time of the current connection, rounded to the nearest multiple of 25 milliseconds. | To get the end-to-end latency value we get the value (a number) of round-trip time property from the [NetworkInformation.rtt - Web APIs]. |
| **Uplink latency** | $t_0$: Timestamp of the current time the request is being send from the mobile device to the edge node<br>$t_1$: Timestamp of the time the request is received by the edge node | A ping edge service was implemented for this purpose. We get the uplink latency by subtracting these two reference timestamps | $t_1 - t_0$ |
| **Downlink latency** | $t_0$: Timestamp of the current time the request is being send from the edge node to the mobile device<br>$t_1$: Timestamp of the time the request is received | A ping edge service was implemented for this purpose. We get the downlink latency by subtracting these two reference timestamps | $t_1 - t_0$ |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI |
|---|---|---|---|
| | by the mobile device | | |
| **Data rate**<br>Target value:<br>1) High resolution media: > 20 Mbps<br>2) 360-degree high resolution media: > 100 Mbps<br>3) Immersive media: > 100 Mbps | **A**: Application layer throughput across recently active connections, excluding connections made to a private address space. | The **NetworkInformation.downlink** Web API returns the effective bandwidth estimate in megabits per second, rounded to the nearest multiple of 25 kilobits per seconds. | We get the value of downlink property from the NetworkInformation.downlink - Web APIs. |
| **Jitter**<br>Target value:<br>1) High resolution media: < 10-50 ms<br>2) 360-degree high resolution media: < 10 ms<br>3) Immersive media: < 10 ms | **A**: Current round-trip time<br>**B**: Previous round-trip time | To get the jitter value we calculate the average subtraction values of consequent rtt measurements. | $\frac{abs(A-B)[0] + abs(A-B)[1] + \cdots + abs(A-B)[N]}{N}$ |
| **Framerate**<br>Target value:<br>> 30 fps | **A**: Total video frames in $t_1$<br>**B**: Total video frames in $t_0$ | The VideoPlaybackQuality.totalVideoFrames returns the total number of video frames that have been displayed or dropped since the media was loaded. | To get the framerate we divide the total frames in a time period with this period of time:<br>$\frac{A-B}{t_1-t_0}$ |
| **Mobility interruption time** | $t_0$: Timestamp at the time the **mobile** lost its connection with the network<br>$t_1$: Timestamp at the time the network | To get the interruption time we subtract the timestamp of the time that the network connection was reinstated with the timestamp of the time | $t_1 - t_0$ |

| Service KPI | Measurement parameters | Measurement procedure | How and where to compute the KPI |
|---|---|---|---|
| | connection was reinstated | that the interruption has occurred | |
| **Interruption Flag** | Depends on interruption metric | A flag which is true (‘1’ as value) when the mobile network connection is offline and false (‘0’ as a value) when the mobile has an active network connection | Get the mobile network state by adding an event listener on ‘online’ and ‘offline’ events |
| **Reliability**<br>Target value: 99.9% | **A:** the total interruption time.<br>**B:** the elapsed time of the network connection. | To get the reliability we aggregate past interruption times and subtract this time by the elapsed network connection time. We then convert this to a percentage format. | $100 - \left(\frac{A}{B}\right) * 100$ |
| **Service migration time**<br>Target value: 20 – 35 s | **A**: Timestamp generated by the Decision Engine on migration flag.<br>**B**: Timestamp generated by TP Edge when the service is ready on the new Edge.<br>**C**: Time elapsed from new migration is triggered, to the TP Edge server is up and running. | Time required by the Orchestrator to migrate the TP Edge service components from one edge node to another. | Execution of the migration from the moment the Decision Engine flags for new migration until the containers are ready on the target Edge server.<br>**C = B - A** |

*Table 5: Service KPIs, measurement methodology and tools for TP service of UC4*

Funded by the Horizon 2020 Framework Programme of the European Union

# 3. Service KPIs Platform Implementation

## 3.1 KPI Platform Architecture

The general architecture of the service KPI platform has been introduced in the deliverable D2.2 [2]. Figure 3 below illustrates this architecture, where:

- A common platform composed of a MongoDb database and Grafana visualization is deployed in the 5GMED Cloud.
- Every vehicle under tests and service running for supporting the 5GMED use cases are in charge of taking the measurements needed for calculating service KPIs. These KPIs are calculated by the UC entities (vehicle, infrastructure, cloud service) and uploaded on the common KPI platform.

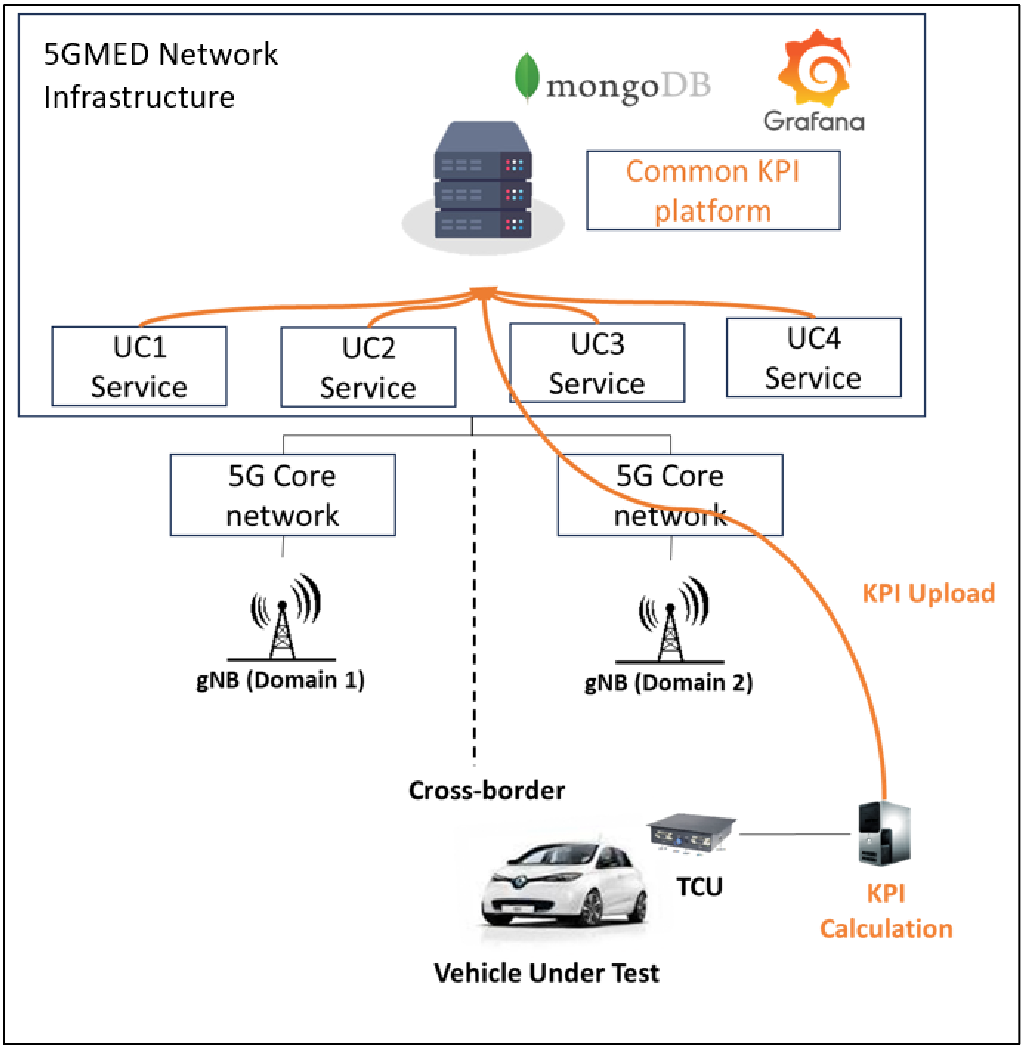


*Figure 3: Overall architecture of the common KPI platform*

The list of KPIs to be uploaded as well as the procedure to compute them being specific for every use case and their corresponding test cases, UC owners are responsible to specify the KPIs and define a structure for storing them as time series into the database.

## 3.2 Database structures

### 3.2.1 Use Case 1

The remote vehicle recovers the information from the Valeo Teleoperation Cloud (VTC) and the TCU to log the following data:

- Timestamp: Internal time at which the measurement is made.
- UTC Timestamp: GPS time at which the measurement is made.
- Latitude: Latitude of the geographical position of the vehicle at which the measurement is made.
- Longitude: Longitude of the geographical position of the vehicle at which the measurement is made .
- Heading: Heading of the vehicle.
- Command End-to-End Latency.
- Data End-to-End Latency.
- Command reliability.
- Sensing reliability.
- Uplink Service Data-Rate.
- Downlink Service Data-Rate.
- Disconnection time: Time when a disconnection occurs.
- Connection time: Time when a connection occurs.
- Mobility interruption time: time value when the calculation of interruption time is performed.
- Handover success event: to indicate if a handover succeeded or failed.
- Velocity: Speed of the vehicle.

### 3.2.2 Use Case 2

The connected vehicle retrieves information from the V2X messages processed by the TCU. This is handled by a JSON object containing the following fields:

- MessageType: CAM, DENM or MCM.
- Delay: delta time difference between detectionTime and receptionTime (ms).
- DetectionTime: also known as transmissionTime, it refers to the time at which the originating station has detected an event.
- Latitude: Location of detected event.
- Longitude: Location of detected event.
- OriginatingStationId: stationId of the original station that detected the event. Together with sequenceNumber it constitutes the MessageId, also known as ActionId.
- ReceivingStationId: receiving stationID.

- ReceptionTime: Reception timestamp.
- SendingStationId: stationId of the message's sender.
- SequenceNumber: Message sequence number to identify it. Together with originatingStationId it constitutes the MessageId, also known as ActionId.
- StationIp: IP address of the current station, that received the message.
- V2xgatewayIp: IP address of the V2X Gateway to which the TCU is connected.
- Mobility interruption time.

### 3.2.3 Use Case 3

For B1 service, the database structure consists of the following fields:

- Timestamp: time at which the measurement is made.
- Latitude: Latitude of the geographical position of the train at which the measurement is made.
- Longitude: Longitude of the geographical position of the train at which the measurement is made.
- Speed: Speed of the train at the time the measurement is made.
- Uplink data rate.
- Downlink data rate.
- Uplink edge end-to-end latency.
- Downlink edge end-to-end latency.
- Uplink reliability.
- Downlink reliability.
- Mobility interruption time.

For P1 service, the database structure consists of the following fields:

- Timestamp: time at which the measurement is made.
- Uplink data rate.
- Downlink data rate.
- Uplink Cloud End-to-End latency.
- Downlink Cloud End-to-End latency.
- Uplink reliability.
- Downlink reliability.
- Mobility interruption time.

For P2 service, the database structure consists of the following fields:

- Timestamp: time at which the measurement is made.
- Uplink data rate: throughput of lidar data.
- Uplink latency.
- Downlink latency.
- Uplink reliability.
- Mobility interruption time.

For B2 service, the database structure consists of the following fields:

- Timestamp: time at which the measurement is made.
- Uplink data rate.
- Downlink data rate.
- Uplink reliability.
- Downlink reliability.
- Mobility interruption time.
- End-to-End latency between UE's using the 5G small-cell on-board.

### 3.2.4 Use Case 4

In the EMT Service, the EMT Client, running in the UE, generates the following metrics and uploads them to the EMT Cloud Server every second. The database structure is composed of the following fields:

- Service: Service name. Value type: string.
- User: Username connected to the EMT Service. Value type: string.
- Coordinates: Latitude and longitude values are taken from the device's GPS. Value type: float.
- Speed: Speed value in m/s taken from the UE's GPS. Value type: float.
- Mobilityinterruption: Time that the service has lost network connection in milliseconds. Value type: integer.
- Interruption: Interruption time, in milliseconds, experienced by the user. Value type: integer.
- Interruptionflag: Set to 0 if the video plays correctly and 1 when there is a "buffering pause". Value type: integer.
- Bandwidth: Bandwidth estimate in bps. Value type: float.
- Fps: Frames per second value. Value type: integer.
- Reliability: Service performance value. Value type: float.
- Applicationlatency: Processing time, in milliseconds, of a video packet. Value type: float.
- UL latency: Uplink latency value in milliseconds. Value type: float.
- DL latency: Downlink latency value in milliseconds. Value type: float.
- Roundtriptime: Time that takes to get a response of a request in milliseconds. Value type: float.
- Timestamp: Value in milliseconds. Taken when the metrics are sent from the UE to the EMT Cloud Server. Value type: integer.
- Mec: Edge server IP where the user is connected. Value type: string.

In order to measure the Service Migration time, Decision engine will write the first timestamp to the MongoDB indicating that the orchestrator has been flagged for deployment, then the EMT Edge server writes the following data into the MongoDB upon start.

- Servicename: Name of the service that is logging the start-up time. Value type: string.
- Servicestartuptime: Timestamp value in milliseconds. Taken when the service is up and running on the new edge server. Value type: integer.
- Ip: Edge server ip where the new service instance is deployed. Value type: string.

In the TP Service, the TP Client, running in the UE, generates the following metrics and uploads them to the TP Cloud Server every second. The database structure is composed of the following fields:

- Service: Service name. Value type: string.
- User: Username connected to the TP Service. Value type: string.
- Coordinates: Latitude and longitude values are taken from the device's GPS. Value type: float.
- Speed: Speed value in m/s taken from the UE's GPS. Value type: float.
- Interruption: Interruption time, in milliseconds, experienced by the user. Value type: integer.
- Interruptionflag: Set to 0 if the video plays correctly and 1 when there is a "buffering pause". Value type: integer.
- Bandwidth: Bandwidth estimate in bps. Value type: float.
- Fps: Frames per second value. Value type: integer.
- Reliability: Service performance value. Value type: float.
- Applicationlatency: Processing time, in milliseconds, of a video packet. Value type: float.
- UL latency: Uplink latency value in milliseconds. Value type: float.
- DL latency: Downlink latency value in milliseconds. Value type: float.
- Timestamp: Value in milliseconds. Taken when the metrics are sent from the UE to the TP Cloud Server. Value type: integer.

To measure the Service Migration time, Decision engine will write the first timestamp to the MongoDB indicating that the orchestrator has been flagged for deployment, then the TP Edge server writes the following data into the MongoDB upon start.

- Servicename: Name of the service that is logging the start-up time. Value type: string.
- Servicestartuptime: Timestamp value in milliseconds. Taken when the service is up and running on the new edge server. Value type: integer.
- Ip: Edge server ip where the new service instance is deployed. Value type: string.

## 3.3 Synchronization

### 3.3.1 Ground equipment

The 5GMED Network has an NTP server running in one of the Virtual Machines hosted in the private Cloud in Castelloli, and it is accessible from all other machines. This NTP server is synchronized with the public master server ROA (https://www2.roa.es/hora/). The 5GMED virtual machines use a Linux daemon to synchronize their operating system with the NTP clock source. In turn, all services running on these machines use the time provided by the OS. The Druid 5G Core (Raemis platform) is one of the services getting synchronized in that way.

The ground equipment of UC1 receives a GNSS clock, processed by a local clock and passed on to the car and the remote station so that their clocks are also synchronized. The UC1 middleware timestamps correctly every data and discards outdated commands/data.

The ground equipment to be synchronized in UC2 is the TMC Edge server. It is done by using the 5GMED Network NTP server.

The ground equipment related to UC3 consists of various virtual machines and computers, as well as the ACS gateway. These are synchronized using the 5GMED Network NTP server.

The ground equipment of UC4 consists of the TP & EMT servers. Synchronization will be achieved through the utilization of the NTP server on the 5GMED Network.

### 3.3.2 Mobile equipment

As described in the previous section, the Druid 5G Cores get their time sync from the container OS. UEs connected to the 5G network receive their time sync from the 5G Core once they attach to the network.

The mobile devices of UC2 that will have to be synchronized are the TCUs. The policy will be that for all TCUs, ideally, they will have their own GPS-based time synchronization.

- VEDECOM TCU: This development was done before the trial preparation phase. GPS-based local synchronization is available at the time of writing.
- CTTC TCU: This development was done before the trial preparation phase. GPS-based local synchronization is available at the time of writing.

The mobile devices of UC3 consists of various computers on board the train, as well as the ACS gateway. These will be synchronized to an NTP server located on board the train.

UC4 employs Nord One Plus mobile devices (UEs), which will be synchronized by the 5G Core upon their connection to the 5GMED Network.

# 4. Use Case 1 Test Cases

In this document, we focus on performance evaluation of the use cases, but focusing on communications and cross-border issues. For this reason, those preliminary test cases defined in D2.2 only for functional validation of the use cases are out of the scope of this document, instead, they will be considered in WP4 and WP5 to generate the functional validation results in D4.3 and D5.3.

Therefore, since the "TC01 - System Failure detection", “TC02 - Request for Remote Assistance”, “TC04 - Teleoperation at High-speed”, “TC06: Teleoperation Failure and Minimum Risk Maneuver” and “TC07: Prevention from Dangerous Teleoperation” defined in D2.2 are related to autonomous driving functionalities, they have not been included in this deliverable. The remaining test cases of UC1 are: “TC03: Teleoperation in Highway Insertion” and “TC05: Teleoperation at High-speed in Cross-border”.

## 4.1 UC1-TC03 : Teleoperation in Highway Insertion

### 4.1.1 Test case type

Verification and performance evaluation of service 3 (ToD) in UC1.

### 4.1.2 Test case pre-conditions

UC1-TC02 successful and Remote Vehicle stopped.

### 4.1.3 Checklist

- ToD connection in RV: Verification that the teleoperator has taken control of the vehicle by status indicator and acting on the brake.
- ToD connection in RS: Verification that the vehicle has in control by status indicator on the teleoperator interface.

### 4.1.4 Test case purpose

The purpose of this test case is to evaluate the teleoperation of the connected and autonomous vehicle to take it safely back onto the highway traffic, on the rightmost lane, at the minimum legal speed. The test starts with the vehicle stopped on the emergency lane and when all technical conditions are met to initiate the teleoperation maneuver. The remote driver (teleoperator) has agreed to intervene for assistance on that vehicle. One of the most challenging driving moments on a highway environment may happen when a stopped vehicle must get to speed, leave the emergency lane and re-enter the traffic flow safely. With a teleoperated driving mode, this maneuver is obviously expected to be even more challenging, hence an essential test case is mandatory to assess its performance. The remote vehicle is stopped (0km/h) on the emergency lane and all technical conditions are supposedly met to initiate a ToD operation. The remote driver (Teleoperator) has already agreed to intervene for assistance on that remote vehicle. The test case is about the remote driver diagnosing whether ToD is manageable given the remote vehicle state and then taking the remote vehicle safely back onto the highway traffic, on the rightmost lane, at the minimum legal road speed.

### 4.1.5 Equipment involved

Autonomous vehicle is capable of ToD and equipped with a 5G TCU. The teleoperator site is ready behind the remote station anywhere else.

#### 4.1.5.1 Vehicles

The remote vehicle, the VTC and the remote station are involved to operate the ToD.

#### 4.1.5.2 Networks

The 5G network is involved and one responsible of the following component must be active: RV, RS, VTC and TCU. As an example, in Castelloli small-scale test site the most relevant network elements for this test Case are (see Figure 4):

- Two 5G SA Networks
- Valeo Teleoperation Station
- Valeo self-driving car

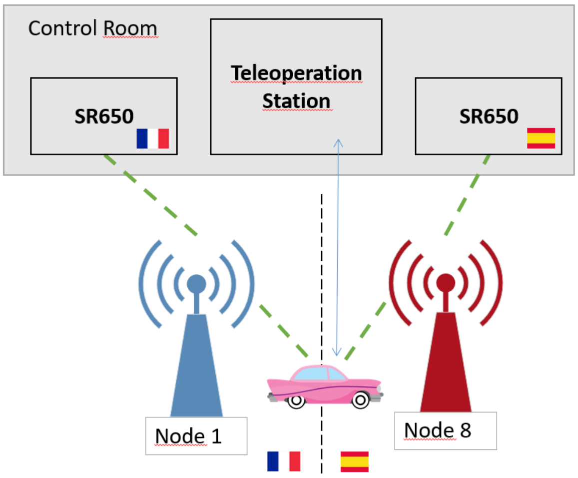


*Figure 4: 5G Network elements used in UC1*

The network elements are specifically described in D3.3. check that deliverable for specific information about used equipment and network configuration. For this test cases in the small-scale test site the roaming is configured as HRR (Home Routed Roaming) using the equivalent PLMN functionality as well as the N14 interface with radio handover.

### 4.1.6 Test steps

To carry out the test case, the remote driver must take the following actions as illustrated by the following Figure 5:

- Ensure a proper technical status (vehicle and connectivity) of the remote vehicle.
- Ensure a proper perception of the environment (ex: non obstructed view of the rear, good view at the front).
- Make a good assessment of the safety of an insertion manoeuvre (incoming traffic in particular).
- Proceed with adequate timing and reinsertion speed relatively to the incoming traffic.
- Operate in a way that feels safe and comfortable for passengers.
- Stabilize the remote vehicle at minimum legal speed on the highway rightmost lane.

This test case reproduces what a regular driver would do when leaving an emergency lane. First the remote driver diagnoses that the vehicle is operational (ex: proper view of the rear), then looks at the highway traffic until he assesses the time is right. The remote vehicle occupants are notified, and the remote vehicle will speed up. When traffic allows, the remote driver will insert into the rightmost lane of the highway and keep on accelerating until reaching the minimum legal speed.

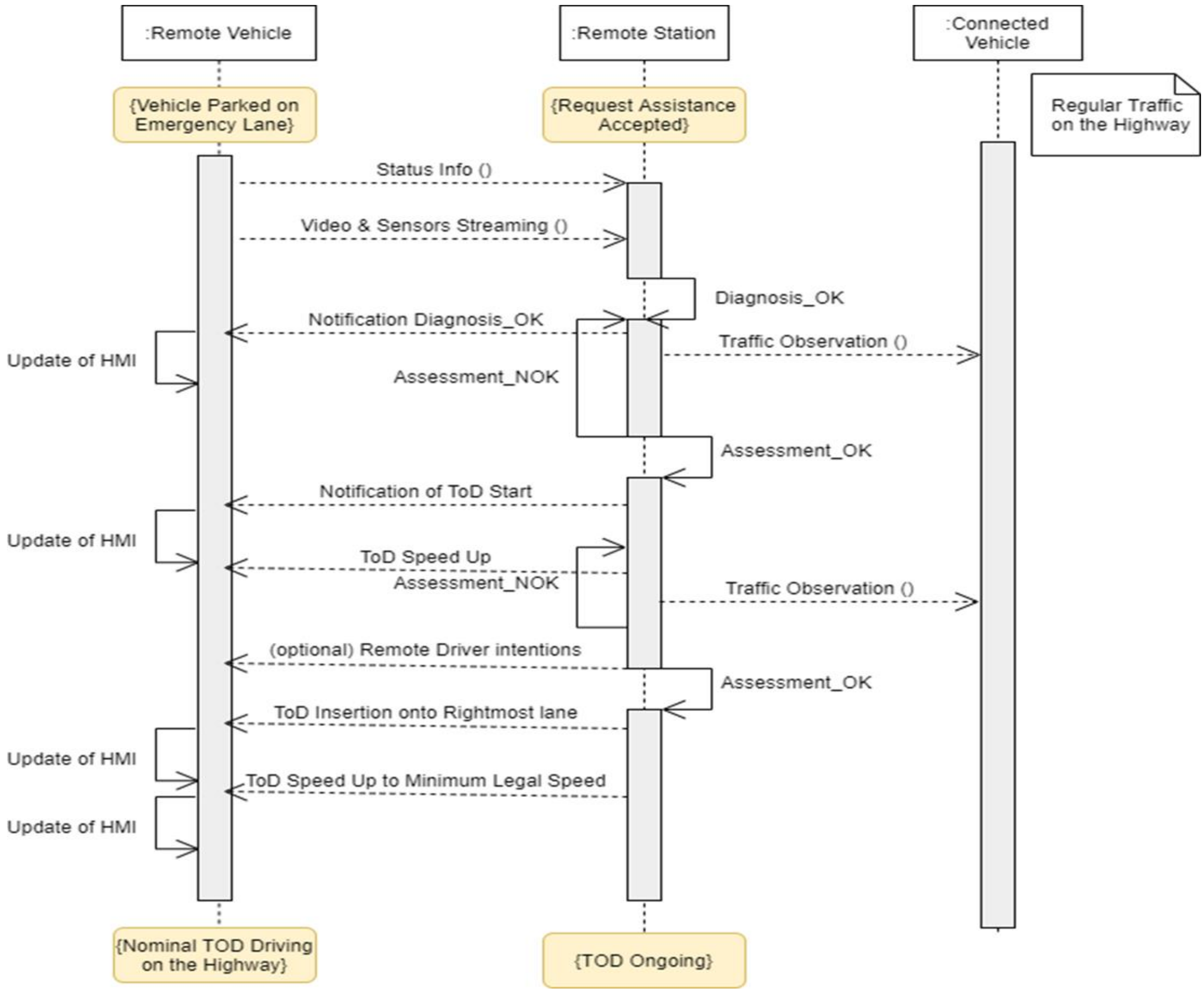


*Figure 5: UC1-TC 03 data flows*

### 4.1.7 Measurements transmitted to the database

- GPS data
- Command End-to-End Latency
- Data End-to-End Latency
- Command Reliability
- Sensing Reliability
- Uplink Service Data-Rate
- Downlink Service Data-Rate
- Mobility interruption time
- Handover success rate
- Velocity

Funded by the Horizon 2020 Framework Programme of the European Union

### 4.1.8 Test results and success criteria

All the steps of the teleoperation maneuver in the highway insertion must be successfully executed in safety conditions. To this end, it is of great importance that the remote driver is fully aware of the vehicle's situation and can evaluate the incoming traffic, including distances to other vehicles, relative speeds, and times-to-reach. Thus, video images and sensors data must be perceived by the remote driver with sufficient quality and short delay. Furthermore, the vehicle's actuators must execute commands reliably and with low latency to facilitate full control by the remote driver.

The following service KPIs will be measured, analyzed, and monitored during the realization of this test case.

- Command End-to-End Latency
- Data End-to-End Latency
- Command Reliability
- Sensing Data Reliability
- Uplink Service Data-Rate
- Downlink Service Data-Rate

For the successful completion of the test case, the corresponding target values (detailed in Table 1) must be accomplished for each service KPI.

## 4.2 UC1-TC05 : Teleoperation at High-speed in Cross-border

### 4.2.1 Test case type

Verification and performance evaluation of service 3 (ToD) in UC1.

### 4.2.2 Test case pre-conditions

UC1-TC03 successful.

### 4.2.3 Checklist

Same as UC1-TC03

### 4.2.4 Test case purpose

The purpose of this test case is to evaluate the teleoperation of the connected and autonomous vehicle at high speed, i.e., 80 km/h in France and 60km/h in Spain. The test starts with the vehicle on the right lane of the highway. The remote driver must be able to operate the vehicle as efficiently, safely, and comfortably as a driver seated inside, adapting the vehicle position inside the ego lane (laterally), maintaining the speed, anticipating the traffic flow, and changing lanes.

### 4.2.5 Equipment involved

Autonomous vehicle is capable of ToD and equipped with a 5G TCU. The teleoperator site is ready behind the remote station anywhere else.

#### 4.2.5.1 Vehicles

The remote vehicle, the VTC and the remote station are involved to operate the ToD.

#### 4.2.5.2 Networks

This test case is using the same network elements as the ones described in the previous Section 4.1.5.2. See details above.

### 4.2.6 Test steps

As shown is Figure 6, the remote driver executes a sequence of teleoperated maneuvers that are common in the situation of stabilized highway driving: speed variations, lane changes, and activation and deactivation of ADAS functions.

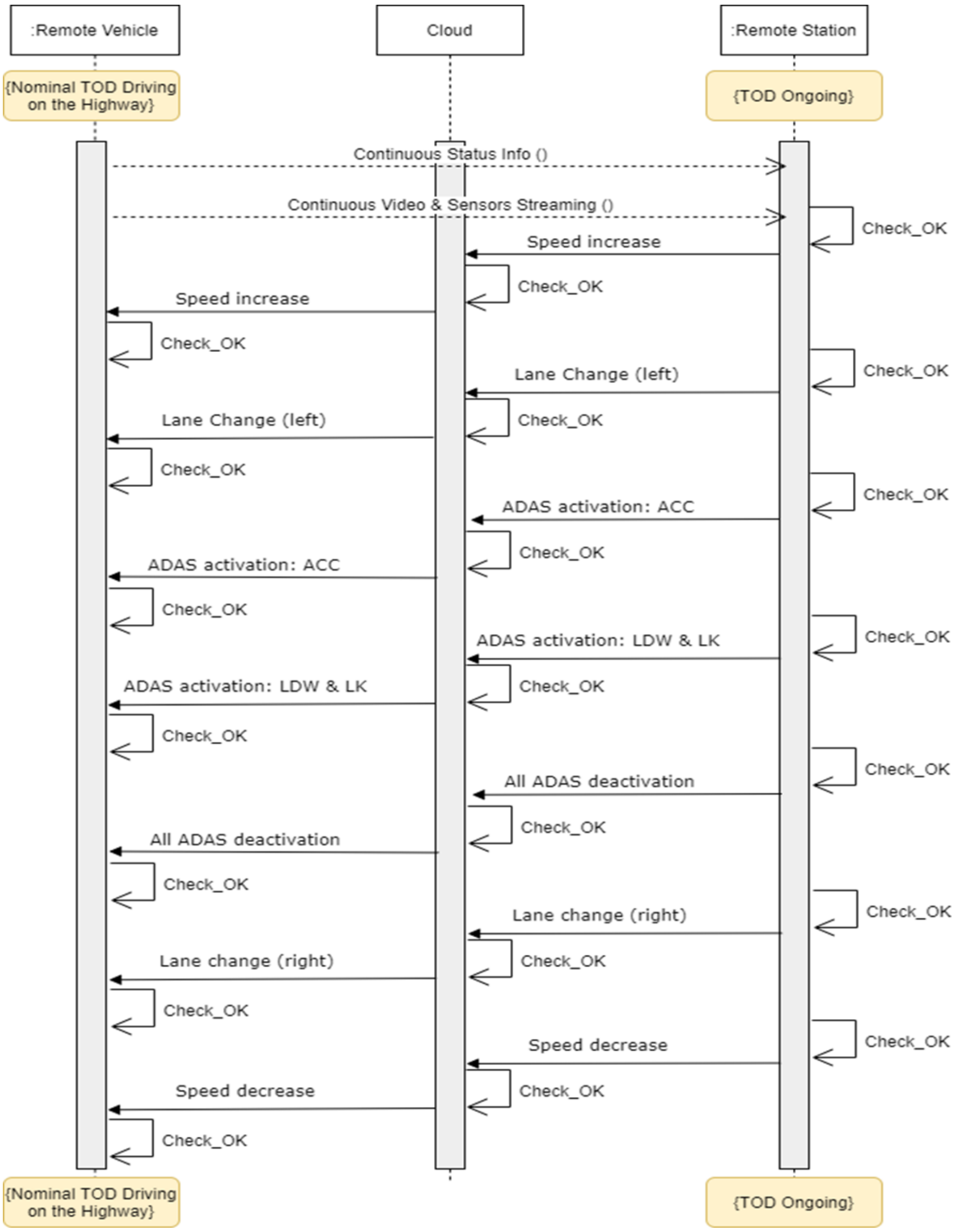


*Figure 6: UC1-TC 05 data flows*

### 4.2.7 Measurements transmitted to the database

- GPS data
- Command End-to-End Latency
- Data End-to-End Latency
- Command Reliability
- Sensing Reliability
- Uplink Service Data-Rate

- Downlink Service Data-Rate
- Mobility interruption time
- Handover success rate
- Velocity

### 4.2.8 Test results and success criteria

All the steps of the teleoperation maneuver at high speed must be successfully executed in safety conditions. As in test case 3, video images and sensors data must be perceived by the remote driver with sufficient quality and short delay, the vehicle's actuators must execute commands reliably and with low latency to facilitate full control to the remote driver.

The following service KPIs will be measured, analyzed, and monitored during the realization of this test case.

- Command End-to-End Latency
- Sensing data End-to-End Latency
- Command Reliability
- Sensing Data Reliability
- Uplink Service Data-Rate
- Downlink Service Data-Rate

For the successful completion of the test case, the corresponding target values (detailed in Table 1) must be accomplished for each service KPI.

# 5. Use Case 2 Test Cases

This section describes test cases that will be carried out during the small-scale trials of UC2. From the preliminary list of the UC2 test cases presented in the D2.2 [2], and according to the project objectives, in this deliverable we will just describe the Test Cases related with performance evaluation of service 1 (REM) and service 2 (AID).

- **UC2-TC02:** Relay of Emergency Messages (REM) in Cross-Border Scenario
- **UC2-TC04:** Automatic Incident Detection (AID) in Cross-Border Scenario

The rest of test cases will be carried out and explained in D4.3, since they are related to the functional validation.

- **UC2-TC01:** Relay of Emergency Messages (REM) with one MEC
- **UC2-TC03:** Automatic Incident Detection (AID) with one MEC

## 5.1 Map overview

All the small-scale test cases of UC2 will take place in the closed circuit Parcmotor in Castelloli. There will be two scenarios corresponding to the two services that will be tested: REM (Figure 7) and AID (Figure 8).

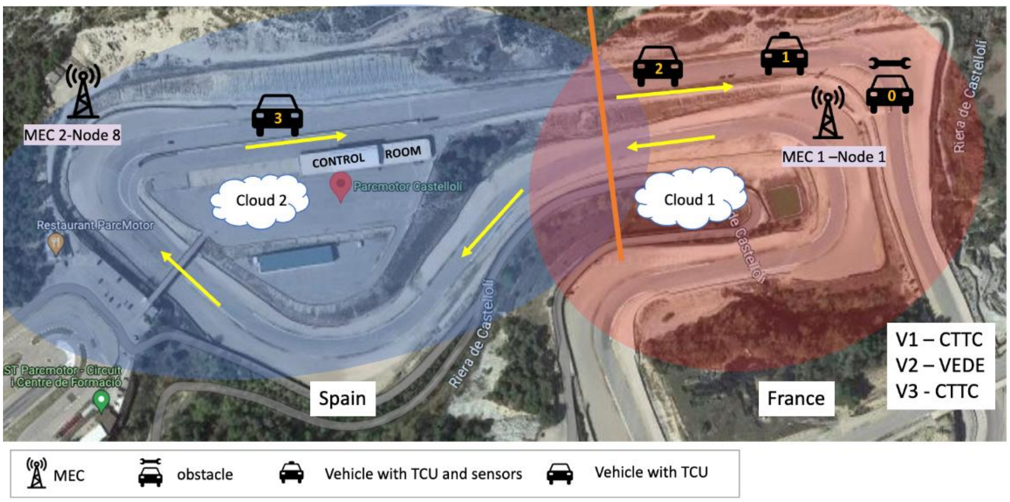


*Figure 7: Map overview of UC2 - Scenario 1: Service 1 – REM*

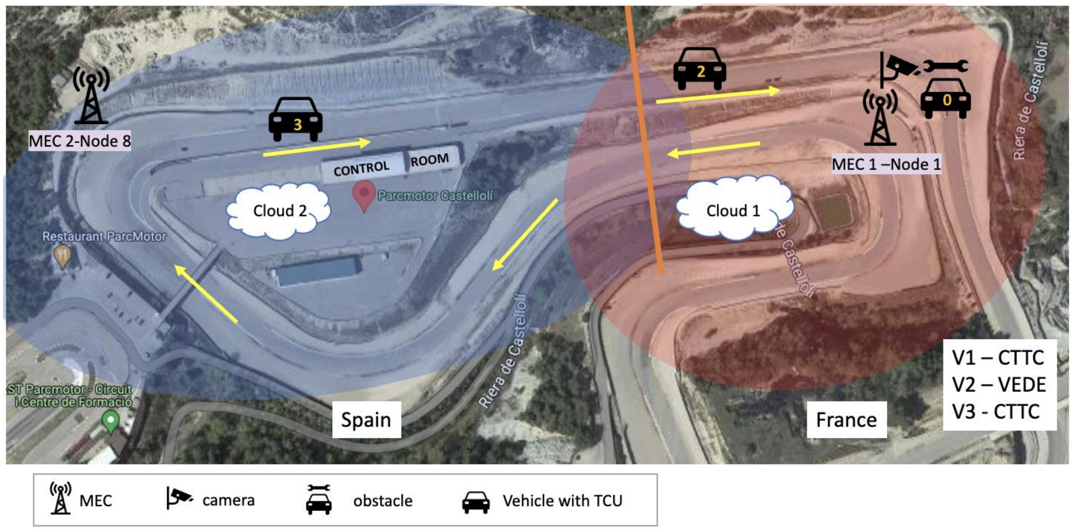


*Figure 8: Map overview of UC2 - Scenario 2: Service 2 – AID*

The network equipment is located in Node 1 and Node 8, as reported in the following Figure 9. Both nodes are physically deployed in the Control Room in the virtualized SR650 Lenovo servers.

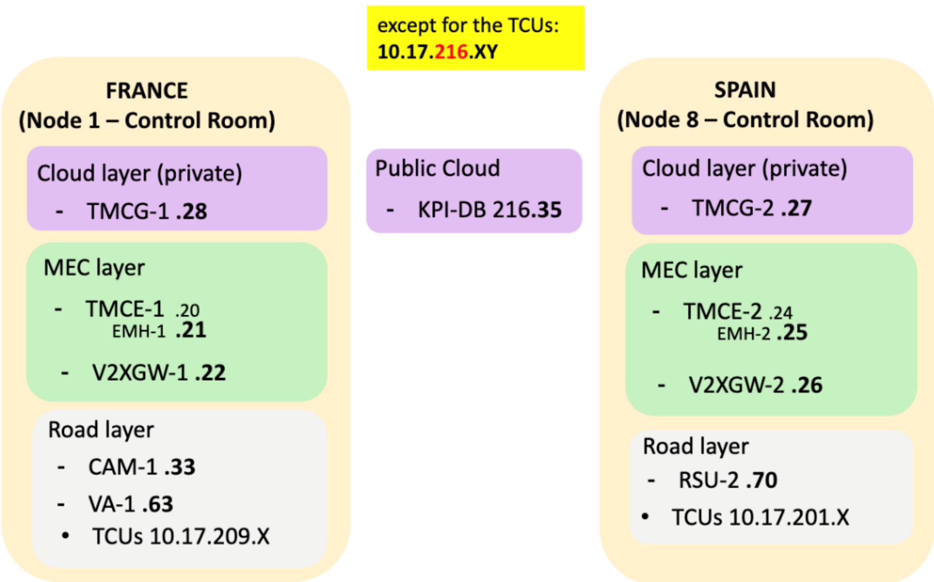


*Figure 9: IP list of servers used in UC2*

The TCUs IP addresses are dynamically assigned by the 5G SA networks.

## 5.2 UC2-TC02: Relay of Emergency Messages (REM) in Cross-Border Scenario

### 5.2.1 Test case type

Performance evaluation of service 1 (REM) in UC2.

### 5.2.2 Test case pre-conditions

- UC2-TC01 successful.
- UC2-TC02 functional verification done.
- At least one MEC server in each country with the TMC Edge and V2X Gateway up and running.
- One TMC Global at each of the Clouds up and running.
- At least one connected vehicle to trigger the periodical transmission of emergency messages with “fake” hazard information.
- At least two connected vehicles, to receive warning messages and traffic strategies and display on HMI or dashboard.

### 5.2.3 Checklist

**TCU**

- TCU (laptop) is turned ON.
- NTP server on-board is ON and GPS antenna connected.
- 5G Modem connected to TCU Laptop.
- TCU Laptop is connected to NTP server.
- GPS receiver connected to TCU Laptop.
- GPS server (gpsd) is running (gps.sh).
- 5G Modem is connected to 5G network.
- Interruption_time.py script is running.
- Qmi-cli network monitoring is running.
- Pings to 8.8.8.8 using wan interface is running.
- TCU application is running.

**TMC**

- TMC Edge servers have network access to their respective TMC Global servers, and vice versa.
- TMC Edge France, TMC Global France, TMC Edge France and TMC Global Spain are up and running.
- TMC Edge servers have network access to their respective V2X Gateway servers.
- TMC Edge France is able to subscribe to the EMH MQTT broker and receive messages published by the VA server.

### 5.2.4 Test case purpose

The purpose of this test case is to evaluate the performance of service 1 and stress the system in the cross-border scenario. The service KPIs will be validated by replicating several messages from the TCU of a vehicle that will simulate the detection of several hazardous situations along the cross-border scenario triggering the corresponding warning messages. Additionally, traffic strategies will be generated and transmitted from both the TMC Edge and the TMC Cloud, to the vehicles driving behind that may be affected by the simulated hazards.

### 5.2.5 Equipment involved

#### 5.2.5.1 Vehicles

| Vehicle | Purpose | Hardware | Software | People |
|---|---|---|---|---|
| CTTC Vehicle 1 | -Hazard detection.<br>-Processing V2X messages from UC service | - Jetson Nano<br>-Intel Realsense Depth Camera<br>-Commsignia OM4 OBU<br>-GPS receiver<br>- Car PC laptop<br>-Electric connection cables | - Smart sensor application<br>- TCU application(s)<br>- Ubuntu OS and necessary libraries. | - Driver<br>- TCU operator |
| CTTC Vehicle 2 | -Processing V2X messages from UC service | -Commsignia OM4 OBU<br>-GPS receiver<br>- Car PC laptop<br>-Electric connection cables | - TCU application(s)<br>- Dashboard application.<br>- Ubuntu OS and necessary libraries. | - Driver<br>- TCU Operator |
| VED vehicle | - Processing V2X messages from UC service | - Laptop<br>- Electric connection tables<br>- GPS receiver<br>- Raspberry PI for NTP time-synchronization<br>- Quectel modem for 5G connectivity | -Ubuntu OS and necessary libraries<br>- TCU application<br>-HMI on tablet | -Driver<br>- TCU Operator |

*Table 6: Vehicle details for UC2-TC02*

### 5.2.5.2 Networks

The network elements in the small-scale test sites and the corridors are specifically described in D3.3, but this section describes the specific parts used by this UC and specifically this Test case.

For this specific test case, that takes place in Castelloli we can remark the elements pictured in Figure 10:

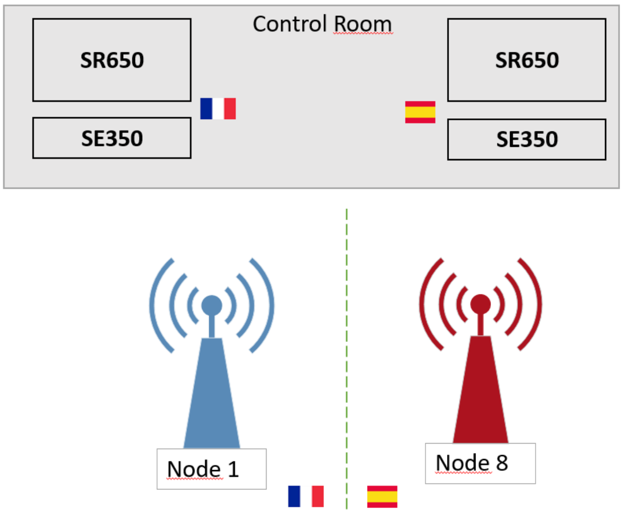


*Figure 10: 5G Network infrastructure used in UC2*

- The Lenovo SR650 (one server per country) are servers that through VMWare virtualization host the different VMs for each country and the Kubernetes clusters that host the 5G SA Cores.
- The Lenovo SE350 (one per country) are edge servers that can host different virtual machines that can be placed closer to the nodes, so latencies are reduced, and services are optimized.
- The different Nodes, also replicated in each country, are Sunwave cells that are connected to the control room through Fiber optic links in the case of Node 1 and through radio links in the case of Node 8.

The French TMC Global and the Spanish one are hosted in respective country SR650server, so each VM is physically segregated in different hardware. In the same way, the TMC edge and the V2X Gateway are also VMs located in the different machines depending on the country they refer to, so the Edge Layer is also differentiated.

The TCUs that are attaching to the 5G SA network, connect to the elements in the edge or in the cloud depending on the service mission.

For this test cases in the small-scale test site the roaming is configured as HRR (Home Routed Roaming) using the equivalent PLMN functionality as well as the N14 interface with radio handover. The scenarios defined for the test case focus on cross-border roaming with vehicles moving across the border.

In this test case, the most important points to have present before starting, to assure the network performance are:

1. Connect into the different Core GUIs, to assure all nodes in France and Spain are up and running.
2. Assure that the VSphere platform and all the networking elements are not sending any alarms and the systems are stabilized.
3. Provide the SIM cards for each country to the users.

### 5.2.6 Test steps

The steps of this test case are described below and will be repeated at least 10 times in order to have statistical significance:

1. One connected vehicle (CTTC V1) will transmit emergency messages periodically while moving in the circuit. The emergency messages will contain information of simulated hazards separated at a certain distance that will depend on the speed of the vehicle and the time period between consecutive emergency messages.
2. All other connected vehicles (VED V2 and CTTC V3) will drive behind the vehicle that transmits emergency messages (V1), at a safe distance, but close enough so that their driving could be affected by the simulated hazards. These vehicles will cross the border and they will receive warning messages and recommendations through the corresponding V2X Gateways using HRR.
3. The TMC Globals will establish the best strategy throughout the corridor when vehicles drive between two different TMC Edges.
4. By driving the entire circuit, the vehicles will cross the border where they will be able to test the effectiveness of 5G roaming to the service (HRR).

It is considered that the necessary samples will be taken for a correct evaluation of the measurements needed for the KPIs. In addition, in order to ensure road safety for all road users, this test case later will be carried out in a real traffic scenario in the cross-border scenario and therefore the warning strategies will be recommendations to be displayed on the HMIs or on the dashboards.

### 5.2.7 Measurements transmitted to the database

The fields sent from the TCUs to the database are the following:

- Station ID Origin
- Message ID
- Station ID Destination
- Message Type: (CAM, DENM, MCM)
- Transmission time
- Reception time
- Time delay
- Latitude
- Longitude
- Number messages transmitted (by message type)
- Number messages received (by message type)
- Hazard End-to-End Latency
- Mobility Interruption Time

### 5.2.8 Test results and success criteria

The following service KPIs shall be measured, analyzed, and monitored during the execution of this test case:

- Hazard End-to-End Latency.
- Hazard Notification Reliability.
- Mobility interruption time.

The criterion for success is that these KPIs reach the theoretically established targeted values (see Table 2).

## 5.3 UC2-TC04: AID in Cross-Border Scenario

### 5.3.1 Test case type

Performance evaluation of service 2 (AID) in UC2.

### 5.3.2 Test case pre-conditions

- UC2-TC03 successful.
- Functional verification of UC2-TC04 successful.
- At least one MEC server in each country with the TMC Edge and V2X Gateway up and running.
- Two TMC Globals at the Cloud (one for each country) up and running in the server.
- Roadside video camera connected via 5G to the corresponding TMC Edge.
- At least two connected vehicles to receive warning messages and traffic strategies and display them on HMI or dashboard.

### 5.3.3 Checklist

**<u>TCU</u>**

- TCU (laptop) is turned ON.
- NTP server on-board is ON and GPS antenna connected.
- 5G Modem connected to TCU Laptop.
- TCU Laptop is connected to NTP server.
- GPS receiver connected to TCU Laptop.
- GPS server (gpsd) is running (gps.sh).
- 5G Modem is connected to 5G network.
- Interruption_time.py script is running.
- Qmi-cli network monitoring is running.
- Pings to 8.8.8.8 using wan interface is running.
- TCU application is running.

**<u>TMC</u>**

- Video Analytics (VA) server has access to the video stream from the camera installed on the test site (looking at the French section).

- VA server has network access to the EMH France server.
- VA server is up and running.
- EMH France has network access to the VA server and TMC Edge France.
- EMH France MQTT broker is up and running.
- VA server is able to publish MQTT messages to the EMH MQTT broker.
- TMC Edge France server has network access to the EMH France server.
- TMC Edge servers have network access to their respective TMC Global servers, and vice versa.
- TMC Edge France, TMC Global France, TMC Edge France and TMC Global Spain are up and running.
- TMC Edge servers have network access to their respective V2X Gateway servers.
- TMC Edge France is able to subscribe to the EMH MQTT broker and receive messages published by the VA server.

### 5.3.4 Test case purpose

The purpose of this test case is to evaluate the performance of service 2 and stress the system in the cross-border scenario. As the purpose is to validate the KPIs of the service based on the detection of hazard events from the video analysis executed in the TMC Edge in one of the MECs, it will be necessary to reproduce several events in MECs simulating that a hazard situation occurs in front of supposed cameras that were deployed in different points of the cross-corridor. These hazard events will be replicated at such a distance that the CVs and CAV driving along the corridor will be affected and should be warned as well as facilitating the maneuver or strategy by the TMC Edges through the corresponding V2X Gateway.

### 5.3.5 Equipment involved

#### 5.3.5.1 Vehicles

| Vehicle | Purpose | Hardware | Software | People |
|---|---|---|---|---|
| CTTC Vehicle 1 | -Processing V2X messages from UC service | - Jetson Nano<br>-Intel Realsense Depth Camera<br>-Commsignia OM4 OBU*<br>-GPS receiver*<br>- Car PC laptop<br>-Electric connection cables | - Smart sensor application<br>- TCU application(s)<br>- Ubuntu OS and necessary libraries. | -Driver<br>-TCU operator |
| CTTC Vehicle 2 | -Processing V2X messages from UC service | -Commsignia OM4 OBU*<br>-GPS receiver*<br>- Car PC laptop | - TCU application(s)<br>- Dashboard application. | -Driver<br>- TCU Operator |

| | | | | |
|---|---|---|---|---|
| | | -Electric connection cables | - Ubuntu OS and necessary libraries. | |
| VED vehicle | - Processing V2X messages from UC service | - Laptop<br>- Electric connection tables<br>- GPS receiver<br>- Raspberry PI for NTP time-synchronization<br>- Quectel modem for 5G connectivity | -Ubuntu OS and necessary libraries<br>- TCU application<br>-HMI on tablet | -Driver<br>- TCU Operator |

*Table 7: Vehicle details for UC2-TC04*

#### 5.3.5.2 Networks

This test case is using the same network elements as the ones described in the previous Section 5.2.5.2. See details above.

### 5.3.6 Test steps

The steps of this test case are described below and will be repeated 10 times in order to have statistical significance:

1. The camera installed in the French node will detect obstacles continuously.
2. The TMC will process and transmit emergency messages to the V2X Gateway.
3. The connected vehicles in the zone (VED V2 and CTTC V3) will be drive behind the obstacle that), since they could be affected by the simulated hazards. These vehicles will cross the corridor and they will receive warning messages and recommendations through the corresponding V2X Gateways using HRR.
5. The TMC Globals will establish the best strategy throughout the corridor when vehicles drive between two different TMC Edges.
6. By driving the entire cross-border corridor, the vehicles will cross the border where they will be able to test the effectiveness of 5G roaming to the service (HRR).

In this test case, traffic status information monitored by cameras is necessary. As only the French side is equipped with roadside cameras, the test will start at a short distance from the border, as the first camera is located right at the border and controlled by the TMC Edge at the French MEC. The first “fake” hazard event will be generated as if it was taking place in that camera and synchronized with the arrival of the vehicles from France, which will be roaming at that moment. The hazard detection will be simulated at the output of the video analysis as if the camera will detect different hazard situations just before the passage of the different connected vehicles.

The next step will be that the connected vehicles will receive via the corresponding V2X Gateway the warning messages and the necessary strategies to avoid collision with this simulated hazard. The TMC

Global will be responsible for defining the best strategy to be applied by the different TMC Edges to the vehicles circulating in the segments affected.

### 5.3.7 Measurements transmitted to the database

The database keeps track of the messages sent and received by all the agents in the network. They send to the database the necessary information to further process and compute the KPIs. The fields sent from the TCUs to the database are the following:

- Station ID Origin: ID of station that created the message.
- Message ID: In the CASE of DENMS, it is composed by station ID and a consecutive number
- Station ID Destination: Station ID of the receiver agent.
- Message Type: (CAM, DENM, MCM)
- Transmission time: It is obtained from the timestamp include in the message.
- Reception time: The current time of the day's timestamp at the moment of reception.
- Time delay: It is the subtraction between transmission and reception time.
- Latitude : Location of the receiving station.
- Longitude: Location of the receiving station.
- Number messages transmitted (by message type).
- Number messages received (by message type).
- Mobility Interruption Time

### 5.3.8 Test results and success criteria

The following service KPIs shall be measured and monitored during the execution of this test case:

- Hazard End-to-End Latency.
- Hazard Notification Reliability.
- Mobility interruption time.

The criterion for success is that these KPIs reach the theoretically established targeted values (see Table 2).

# 6. Use Case 3 Test Cases

## 6.1 Map overview

All tests here described will be performed along the rail track going from the LFP site in Llers to the LFP site in Le Perthus as shown in Figure 11. The network equipment is in both Llers and Le Perthus sites, and additional cloud equipment is located in Castelloli.


*Figure 11: Map overview of UC3*

## 6.2 Rail track preparation for trials

Permission and operation times are agreed with LFP when needed for testing. There is no fixed schedule for all tests but rather an approach based on the needs of every test case and the status of the track when the need arises.

## 6.3 Statistical significance

In order to achieve a statistical significance on results of $p<0.05$ it will be ensured that any conclusion will be supported by at least 30 measurements of the KPI which supports said conclusion. The amount of measurements will depend on the available time.

## 6.4 UC3-TC 01: Service Continuity during Inter-RAT Handovers

### 6.4.1 Test case type

Performance evaluation of the ACS-GW handover.

### 6.4.2 Test case pre-conditions

- Train Access Network (TAN) on ground up and running
- Radio units installed on top of the train up and running

### 6.4.3 Checklist

- Train and Ground network running and properly configured
- Both ACS-GW VMs (on train and ground) up and running
- At least two TANs available
- Check the reachability between the ACS-GW VMs
- Check the capability of enabling/disabling the TANs

### 6.4.4 Test case purpose

The purpose of this test case is to validate the performance of the ACS-GW when it makes the inter-RAT handover process due to the movement of the train across different segments of the cross-border corridor. The ACS-GW is responsible for aggregating/disaggregating the traffic delivered from/to the different application components to/from the different Radio Access Networks that compose the TAN that communicates the train and the ground.
It is required that the ACS-GW selects the radio access technology according to the application and the train position. To do this, the ACS-GW must use a table of priorities assigned to the different radio access networks for each service.

### 6.4.5 Equipment involved

The equipment needed to perform UC3 Test cases are:

- ACS-GW VMs train and ground
- TANs
- Hardware for P1, P2, P3 and B1, B2 Service

#### 6.4.5.1 Network

The elements deployed in the corridor for this UC can be observed in the network infrastructure pictured in Figure 12. The network infrastructure will only be described in this section, but it is used in all test cases and services of UC3.

Although all infrastructure elements are described in D3.3 [1] and D5.2 [1], it is worth noting the following ones:

- The 70GHz network deployed in the Spanish part of the railway corridor in LFP premises.
- The satellite backhaul and the VSAT links, installed in Castelloli for the first round of trials and in the corridor in BTS10 for the second round of trials. They are all interconnected with Hispasat teleport in Arganda and Cellnex network.
- The ground ACS-GW, in LFP premises.
- 5G sites, covering the railway infrastructure, providing another access network to the users and the train itself, for the different train services and test cases.
- 5G Distributed Antenna System, covering Le Perthus Tunnel.

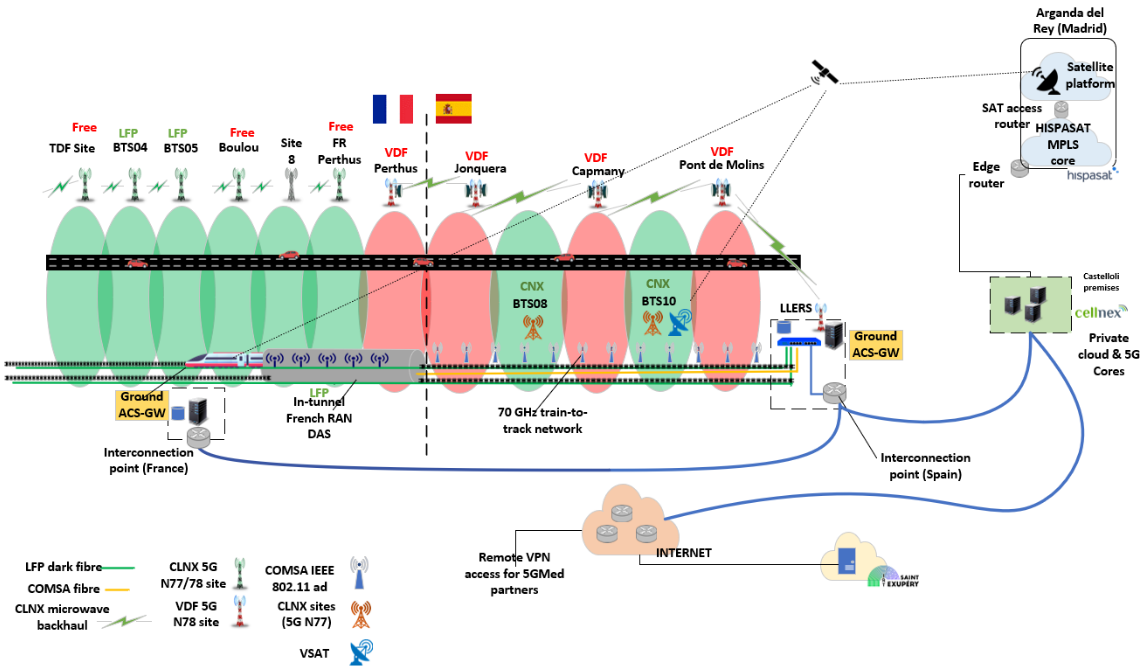


*Figure 12: Network infrastructure used in UC3*

### 6.4.6 Test steps

- Check the Train Network Access links status. When at least two TANs are available, then check that the ACS-GW recognizes the TANs as active
- Start the traffic flows for all the services simultaneously
- Move the train along the entire cross-border corridor
- Check the radio access network selected by the ACS-GW for each service along the different track segments
- Disable and enable some TANs. The ACS-GW must automatically react to these changes as follows:

  - When a preferred link is disabled, all the services' traffic flows that are using that link will hand over to the next-preferred link (i.e., with lower priority), without the need to be re-initiated.
  - When a link is enabled, all the services' traffic flows that declared this link with higher priority with respect to the current one in use, will start using that link without the need to be re-initiated. Measurements transmitted to the database.
- The ACS-GW sends the metrics on the TAN used by each application to the database.

### 6.4.7 Test results and success criteria

Mobility interruption time should remain below target values to ensure session continuity to the services.

## 6.5 UC3-TC 02: Service Continuity during Cross-border Roaming

### 6.5.1 Test case type

Functional verification of the ACS-GW in UC3.

### 6.5.2 Test pre-conditions

- Train Access Network (TAN) on ground and radio units on top of the train up and running.
- Test case 1 successful.

### 6.5.3 Test case purpose

The purpose of this test case is to validate the functional operation of the ACS-GW when it executes a roaming process due to the movement of the train across the border between Spain and France, changing from one Mobile Network Operator to another. Le Perthus tunnel contains the border between both countries. It is assumed that the 5G roaming process will be executed at the exit of each direction of the tunnel (as Le Perthus has separate tubes for each train direction).

The TAN will route the traffic from the train to the ground according to the train location. When the train is in Spain, the traffic flows will be delivered to the Spanish side of the network; and when the train is in France, the traffic flows will be delivered to the French side of the network. The service components on-board the train will communicate with the MEC server in the Spanish or French side, respectively, when the train is moving along Spain or France. Thus, they will have to handover between one MEC to another when the train crosses the border.

### 6.5.4 Test steps

The steps of this test case are described below:

1. Move the train along the entire corridor. Verify the correct operation of the different services of UC3 during the roaming process.
2. Verify that the TAN switches the traffic to each country, according to the train position.
3. Verify that the MEC servers are used according to the train position.

### 6.5.5 Test results and success criteria

All the steps described above must be successfully executed for all the services of UC3.

## 6.6 UC3-TC 03: FRMCS P1 service

### 6.6.1 Test case type

Functional network verification and performance evaluation of the FRMCS P1 service of UC3.

### 6.6.2 Test pre-conditions

- Test case 1 successful.
- Test case 2 successful.
- Train Control Centre up and running in the cloud.

### 6.6.3 Checklist

Before starting, a simple ping test from the PC bearing the IoT simulator and the PC bearing the Train presentation tool to the TrainControl Center in the cloud shall be successful to make sure the connection is enabled. Then a real round trip while being static can be run: from IoT simulator to Train Control Center and then from Train Control Center to the Train Presentation tool.

### 6.6.4 Test case purpose

The purpose of this test case is to validate the correct operation of the FRMCS P1 service and evaluate the performance of the service along the cross-border corridor. Machine-type data traffic will be generated from a simulator of sensors on-board the train and the data will be transmitted by the ACS-GW through the Train Access Network to the Train Control Centre deployed in the cloud.

### 6.6.5 Equipment involved

#### 6.6.5.1 Vehicles

- LFP train
- Hardware needed :
  - All equipments building the TCN
  - All equipments involved in train to ground communication
  - 2 PCs
- Software needed:
  - IoT simulation tool
  - Train Presentation Tool
  - ACS Gateway

#### 6.6.5.2 Networks

In addition to the common test cases network infrastructure presented in sub-section 6.4.5.1, for this specific test case 03 for UC3 it was used:

- TAN: For HW/SW description of the TAN, please refer to D5.2.
- In the cloud: a virtual machine hosting the Train Control Center software is installed.

### 6.6.6 Test steps

The steps of this test case are described below.

1. Start the Train Presentation tool
2. Activate IoT sensor data regular emission from the IoT simulator tool
3. Move the train along the entire cross-border corridor
4. Repeat the test in the other train direction.

### 6.6.7 Measurements transmitted to the database

- Uplink Data Rate
- Downlink Data Rate
- Uplink Cloud End-to-End Latency
- Downlink Cloud End-to-End Latency
- Uplink Reliability
- Downlink Reliability
- Mobility interruption time

### 6.6.8 Test results and success criteria

All the functionalities of the FRMCS P1 service must work properly along the entire cross-border corridor. The service KPIs will be measured, analyzed, and monitored during the realization of this test case. The results of the service KPIs will be presented to show the performance obtained along the different track segments shown in Table 8.

For the successful completion of the test case, the corresponding target values (detailed in Table 3) must be accomplished for each KPI of the FRMCS P1 service.

## 6.7 UC3-TC 04: FRMCS P2 service

### 6.7.1 Test case type

Performance evaluation of the FRMCS P2 service of UC3.

### 6.7.2 Test case pre-conditions

- Test case 1 successful.
- Test case 2 successful.
- Train Control Centre operative.
- AI module for obstacle detection fully operative allocated in the Edge nodes.
- LiDAR sensor installed and fully operative in the front cabin of the train.

### 6.7.3 Checklist

- Both edge servers are running.
- Both edge servers can be pinged.
- The FRMCS P2 AI Module on the edge server is running and configured
- The FRMCS P2 Train Module on the train is running and configured
- The LiDAR is running and configured

### 6.7.4 Test case purpose

The purpose of this test case is to validate the correct operation of the FRMCS P2 service and evaluate the performance of the service along the cross-border corridor. The LiDAR sensor will send data to an AI module allocated at an Edge node on ground for processing and obstacle detection on the adjacent rail track. When the obstacle is detected, the AI module sends warning message to the Train Control Centre, which finally sends a warning to trains in the other track.

### 6.7.5 Equipment involved

- LiDAR
- Edge server
- Train computer

### 6.7.6 Test steps

The steps of this test case are described below:

1. Introduce the obstacle near the track.
2. Move the train along the entire corridor.
3. When the train arrives at the end of the corridor, move the obstacle to a different segment in the other track.
4. Repeat the test moving the train in the opposite direction.

### 6.7.7 Measurements transmitted to the database

- Uplink Data Rate
- Uplink Edge End-to-End Latency
- Downlink Edge End-to-End Latency
- Uplink Reliability
- Downlink Reliability
- Time for service warning
- Mobility interruption time

### 6.7.8 Test results and success criteria

All the functionalities of the FRMCS P2 service must work properly along the entire cross-border corridor. The SNCF Alarm Control Tool has correctly received the warning.
In addition, the following data will be recorded during the realization of the test case:

- Train direction
- Position where the obstacle is located.

## 6.8 UC3-TC 06: B1 Service

### 6.8.1 Test case type

Performance evaluation of B1 service.

### 6.8.2 Test case pre-conditions

- Test case 1 successful.
- Test case 2 successful.

### 6.8.3 Checklist

- Both ground servers are running.
- The ground server on the country where the train currently is can be pinged from the client on the train.
- The ground server in the other country can be pinged.
- The client on the train is connected to the Wi-Fi network and has the right IP address.

### 6.8.4 Test case purpose

This test case's objectives are to verify the B1 service's effectiveness along the cross-border corridor. iPerf3 will be used to imitate a train full of passengers using Wi-Fi against ground servers. Continuity of service is anticipated, albeit with varying performance outcomes depending on the position of the train.

### 6.8.5 Equipment involved

- PC NUC on ground
- WiFi AP
- ACS-GW
- PC-NUC on train

#### 6.8.5.1 Vehicles

- Train
    - Purpose: Move the cameras along the tracks and through the border
    - Hardware: CCTV cameras, PoE switch, ethernet cables.

#### 6.8.5.2 Networks

In addition to the common test cases network infrastructure presented in sub-section 6.4.5.1, for this specific test case 06 for UC3 it was used:

- Train Network
- Ground Network

### 6.8.6 Test steps

1. Run the command “sudo ./B1-kpis.sh” on the terminal of the PC-NUC on the train.
2. Stop the process once sufficient data has been recorded.

### 6.8.7 Measurements transmitted to the database

The following information will be transmitted:

- Timestamp: time at which the measurement is made.
- Latitude: Latitude at which the measurement is made.
- Longitude: Longitude at which the measurement is made.
- Speed: Speed at the time the measurement is made.
- Uplink data rate: uplink data rate as defined in table 15 of D2.1 [1].
- Downlink data rate: downlink data rate as defined in table 15 of D2.1 [1].
- Uplink edge end-to-end latency: uplink edge end-to-end latency as defined in table 15 of D2.1 [1].
- Downlink edge end-to-end latency: downlink edge end-to-end latency as defined in table 15 of D2.1 [1].
- Uplink reliability: uplink reliability as defined in table 15 of D2.1 [1].
- Downlink reliability: downlink reliability as defined in table 15 of D2.1 [1].
- Mobility interruption time: mobility interruption time as defined in table 15 of D2.1 [1].

### 6.8.8 Test results and success criteria

The results will be evaluated and, should they be within the values specified in Table 3, the test will be considered successful.

## 6.9 UC3-TC 07: B2 service

### 6.9.1 Test case type

Functional verification and performance evaluation of the B2 service of UC3.

### 6.9.2 Test pre-conditions

- Test case 1 successful.
- Test case 2 successful.
- Neutral 5G-Core fully operative and connected to both the 5G networks in both countries.

### 6.9.3 Test case purpose

The purpose of this test case is to validate the correct operation of the B2 service and evaluate the performance of the service along the cross-border corridor.

### 6.9.4 Test steps

The steps of this test case are described below:

1. Move the train along the entire corridor.
2. Verify the B2 small cell is properly identified by the neutral 5G core while the train is in the first country.
3. Verify the B2 small cell radiates the appropriate frequencies in the first country.
4. Verify the B2 test users can communicate between them properly while the train is in the first country:
    a. Users belonging to the same MNO (in national and international calls).
    b. Users belonging to different MNO's (national and international calls).
5. Verify the B2 small cell is properly identified by the neutral 5G core while the train is in the second country.
6. Verify the B2 small cell radiate the appropriate frequencies in the second country.
7. Verify the B2 test users can communicate between them properly:
    a. Users belonging to the same MNO (in national and international calls).
    b. Users belonging to different MNO's (national and international calls).
8. Repeat the test in the other train direction.

### 6.9.5 Test results and success criteria

The following service KPIs will be measured, analyzed, and monitored during the realization of this test case. The results of the service KPIs will be presented to show the performance obtained along the different track segments shown in Table 8.

- Uplink Data Rate
- Downlink Data Rate
- Uplink Reliability
- Downlink Reliability
- Mobility interruption time
- End-to-End latency between UE's using the 5G small-cell on-board

For the successful completion of the test case, the corresponding target values (detailed in Table 3) must be accomplished for each KPI of the B1 service.

# 7. Use Case 4 Test Cases

## 7.1 Map overview

Figure 13 shows an overview of the racetrack where we can differentiate the trajectory that the vehicle will follow, the different UC4 Service layers (client, edge and cloud) and areas of Spanish (Pole8) and French (Pole1) coverage.

During the static tests, the vehicle will remain stationary under the Pole8 coverage area. In the case of the dynamic tests, the trajectory of the vehicle will be a loop, starting from Pole8 to Pole1 and back, so the vehicle will move between countries over time.

As for the different layers (client, edge and cloud), they are distributed as follows:

- Cloud Layer: Common and accessible from both countries. UC4 services running in other layers can access it from any coverage area.
- Edge Layer: UC4 Edge Layer provides functionalities to the Client Layer. The in-vehicle UE is only able to access the Edge Layer provided by the coverage area it is under.
- Client Layer: In-vehicle UE, where the Client Layer of the UC4 services run.

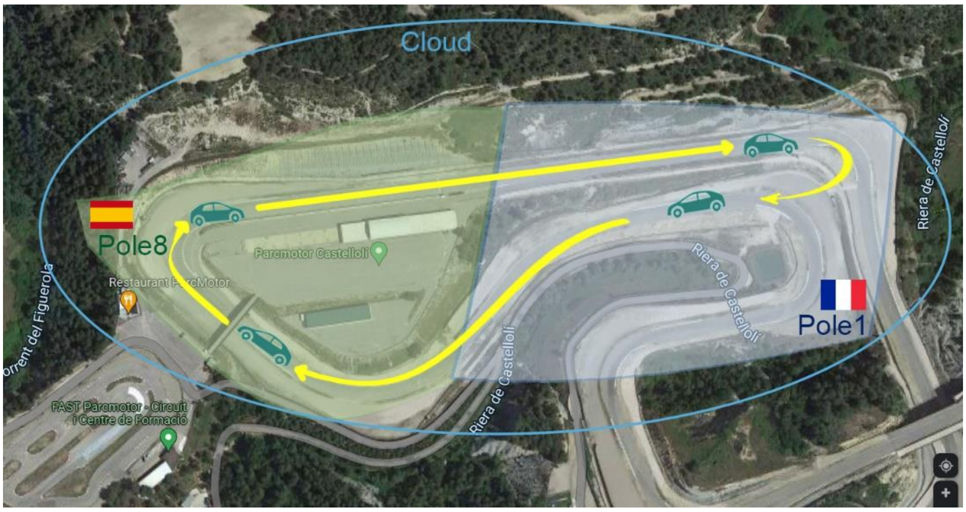


*Figure 13: Map overview of UC4*

Funded by the Horizon 2020 Framework Programme of the European Union

## 7.2 UC4-TC 03

Two static users are located under pole8 coverage area while consuming synchronized video content and using the chat capabilities.

### 7.2.1 Test case type

UC4 EMT service performance evaluation.

### 7.2.2 Test case pre-conditions

Testing infrastructure ready and application deployed.

### 7.2.3 Checklist

Performing the test case will require:

- 2 Users.
- 2 Mobile Devices.
- 2 5G Sim cards.

### 7.2.4 Test case purpose

The purpose of this test case is to display the features of consuming synchronized media content chat capabilities among static users connected to the same 5G network cell through a specific base station. Due to the lack of user movement, some KPIs related to mobility are not generated, therefore edge migration does not occur. On the other hand, the rest of the KPIs are generated as a result of the interaction with the service.

### 7.2.5 Equipment involved

Mobile devices are able to connect to the 5G network and allow the user to interact with the EMT service though the internet browser.

#### 7.2.5.1 Vehicles

This test case involves static users, there is no need for vehicles.

#### 7.2.5.2 Networks

The networks involved in the small-scale test sites for UC4, is described in Figure 14.

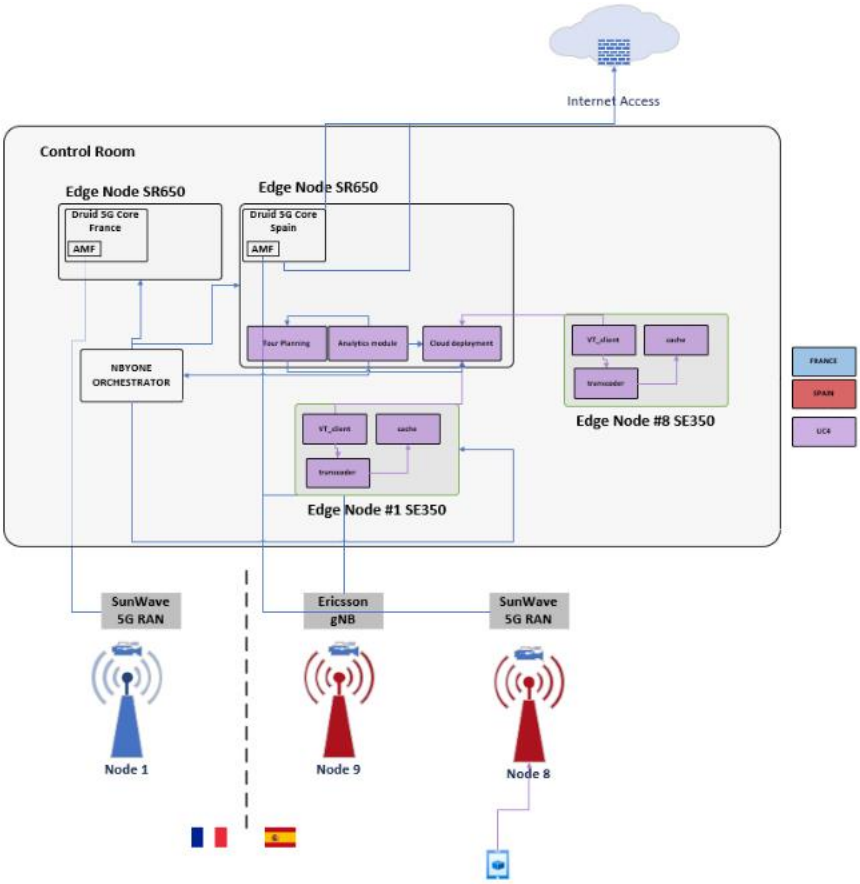


*Figure 14: 5G network infrastructure used in UC4*

The 5G network available through the testing area were in first place three gNodeBs two Sunwave nodes and one Ericsson gNodeB, but in the second round of tests, the Ericsson gNodeB was redeployed in the corridor. Each Sunwave gNodeB is connected to one 5GSA Core showcasing the two countries in the small-scale test site.

For the different test cases there were used two edge nodes SE350 Lenovo servers, connected to the NearbyOne orchestrator. The aim of this connection was to deploy the services in the edge along the edge servers when needed: depending on the mobility of the user. There are also elements involved in the 5GMED cloud, hosted in SR650 Lenovo servers.

This is the common test cases network infrastructure. For this specific test case 03 for UC4 it was used:

- Hardware needed : One edge server managed by one operator.
- Software needed : NearbyOne Orchestrator in order to deploy the services.
- People involved – not required

### 7.2.6 Test steps

1. Users 1 and 2 are located at base station A.

2. User 1 opens Chrome Browser and goes to the EMT application URL.
3. Chrome browser will load the welcome page, User 1 taps the Create a Room button. (Chrome browser will load and User 1 will join the new created room)
4. Now that User 1 is in a room, now is time to invite User 2. User 1 shares the link with User 2 through a messaging application.
5. User 2 taps on the link sent by User 1.
6. Chrome opens up and User 2 joins the room User 1 created. User 2 taps on the Sync Me! Button.
7. User 1 chooses content from the list provided and tap on WATCH button. (Video player opens up for both users and plays the content)
8. While video content is playing. [Chat] User 1 locates the Tab Menu on the right of the screen and taps on the Chat tab. Tab changes to Chat view and on the bottom of the screen User 1 can use the input function to write messages. Tap Send button to send messages to other users.
9. While video content is playing. [Chat] User 2 locates the Tab Menu on the right of the screen and taps on the Chat tab. Tab changes to Chat view and on the bottom of the screen User 2 can use the input function to write messages. Tap Send button to send messages to other users.
10. [Video Player] User 2 pauses the video. (User 1 video player should sync and pause)
11. [Video Player] User 2 resumes the video. (User 1 video player should sync and resume)
12. [Video Player] User 1 pauses the video. (User 2 video player should sync and pause)
13. [Video Player] User 1 resumes the video. (User 2 video player should sync and resume)
14. End of test.

### 7.2.7 Measurements transmitted to the database

As defined in Table 4 of Section 2.4, the metrics that will be collected and stored in the database are the following: service, user, coordinates, mobilityinterruption, interruption, interruptionflag, bandwidth, fps, reliability, applicationlatency, UL latency, DL latency, timestamp and MEC.

### 7.2.8 Test results and success criteria

Users involved in the test case are able to use the EMT service. Communicate with each other using the chat functionality while consuming synchronized video streaming without interruptions. Successful if the transmitted metrics are inside the thresholds of the target values defined in D2.1 [1].

## 7.3 UC4-TC 06

One static user and one user in motion consuming synchronized video content while using the chat with cell handover between edge servers managed by different operators.

### 7.3.1 Test case type

UC4 EMT service performance evaluation.

### 7.3.2 Test case pre-conditions

Test Case 03 to be successful.

### 7.3.3 Checklist

Performing the test case will require:

- 2 Users.
- 2 Mobile Devices.
- 2 5G Sim cards.
- 1 Vehicle.
- 2 Edge servers managed by different operators.

### 7.3.4 Test case purpose

The purpose of this test case is to show the coexistence of one static user and a user in motion while experiencing the EMT service capabilities. The user in motion will travel from base station A to base station C, so an edge migration is required in order to provide the EMT service capabilities. Edge to edge migration occurs between different operators and the user in motion generates the necessary mobility metrics for this to happen.

### 7.3.5 Equipment involved

Mobile devices are able to connect to the 5G network and allow the user to interact with the EMT service though the internet browser.

#### 7.3.5.1 Vehicles

This test case involves a user in motion, therefore a vehicle is required. For every vehicle involved:

- Purpose: The purpose of the vehicle is to transport one of the users from base station A to base station B.
- Hardware needed - Nord One Plus mobile phones.
- Software needed - Internet browser able to load the service (client-server).
- People involved: Car driver is required.

#### 7.3.5.2 Networks

In addition to the common test cases network infrastructure presented in sub-section 7.2.5.2, for this specific test case 06 for UC4 it was used:

- 5G network available through the testing area.
- Hardware needed: Two edge servers managed by different operators.

- Software needed: NearbyOne Orchestrator in order to deploy the services.
- People involved – not required

### 7.3.6 Test steps

1. Users 1 and 2 are located at base station A.
2. User 1 opens Chrome Browser and goes to the EMT application URL.
3. Chrome browser will load the welcome page, User 1 taps the Create a Room button.
   (Chrome browser will load and User 1 will join the new created room)
4. Now that User 1 is in a room, now is time to invite User 2. User 1 shares the link with User 2 through a messaging application.
5. User 2 taps on the link sent by User 1.
6. Chrome opens up and User 2 joins the room User 1 created. User 2 taps on the Sync Me! Button.
7. User 1 chooses a content from the list provided and tap on WATCH button. (Video player opens up for both users and plays the content)
8. While video content is playing. [Chat] User 1 locates the Tab Menu on the right of the screen and taps on the Chat tab. Tab changes to Chat view and on the bottom of the screen User 1 can use the input function to write messages. Tap Send button to send messages to other users.
9. While video content is playing. [Chat] User 2 locates the Tab Menu on the right of the screen and taps on the Chat tab. Tab changes to Chat view and on the bottom of the screen User 2 can use the input function to write messages. Tap Send button to send messages to other users.
10. [Video Player] User 2 pauses the video. (User 1 video player should sync and pause)
11. [Video Player] User 2 resumes the video. (User 1 video player should sync and resume)
12. [Video Player] User 1 pauses the video. (User 2 video player should sync and pause)
13. [Video Player] User 1 resumes the video. (User 2 video player should sync and resume)
14. [User in Motion] User 2 gets into the car and starts travelling to base station C.
15. [User in Motion] User 2 checks [Video Player] function is working without interruption in motion.
16. [User in Motion] User 2 arrives to base station C.
17. End of test.

### 7.3.7 Measurements transmitted to the database

Same as described in 7.2.7.

### 7.3.8 Test results and success criteria

Same as described in 7.2.8. Edge to edge migration between different operators should not have any impact on the user experience.

Funded by the Horizon 2020 Framework Programme of the European Union

## 7.4 UC4-TC 07

A number n of static users and m=0 users in motion consuming high resolution content, 360° video streaming and VR content.

### 7.4.1 Test case type

Performance evaluation of the TP service of UC4.

### 7.4.2 Test case pre-conditions

Testing infrastructure ready and application deployed.

### 7.4.3 Checklist

Performing the test case will require:

- 2 Users.
- 2 Mobile Devices.
- 2 5G Sim cards.

### 7.4.4 Test case purpose

The purpose of this test case is to display the feature of consuming high-resolution, 360 and VR media content when multiple users residing in the same network cell and hence testing 5G connectivity through a specific base station. Since users are not moving through the testbed, migration between edge nodes does not occur. Nevertheless, all the KPIs (described in detail below) that are unrelated to mobility of the users should be met.

### 7.4.5 Equipment involved

Testing infrastructure ready and application deployed. Mobile devices can connect to the 5G network and allow the user to interact with the TP service.

#### 7.4.5.1 Vehicles

This test case involves static users, there is no need for vehicles.

#### 7.4.5.2 Networks

In addition to the common test cases network infrastructure presented in sub-section 7.2.5.2, for this specific test case 07 for UC4 it was used:

- 5G network available through the testing area
- Hardware needed: One edge server managed by one operator.
- Software needed: NearbyOne Orchestrator to deploy the services.
- People involved: Not required.

### 7.4.6 Test steps

1. The user is located at base station A and connects to the network through her mobile device.
2. The user opens the TP service and accepts all permissions if prompted (by tapping on 'while using the app').
3. The user Signs in with her credentials (username and password).
4. Throughout this session the calculated metrics are uploaded to the TP cloud server every 5 seconds for storage and further evaluation, and usage from the project's components (Analytics module, Prometheus etc.).
5. The user taps on the "+" icon (middle icon on the bottom navigation bar) to create predefined tours or the 'Explore' icon on the bottom navigation bar (far right icon) to navigate through the map.
6. The user chooses to navigate through the map (the approach would be similar in the other case). So, once she finds her Point of Interest (POI) she selects it.
7. One to three available types of videos between high resolution, 360 and VR videos appear in the next page. In the case the user wants to stream a 360 or a VR video she should wear the head-mounted device at this point.
8. Then the user clicks on the blue buttons displaying the name of the videos (the availability of the videos depends on each specific site). Consequently, the XR player or the Video player modules (for details of the TP functional block see Fig. 14 in D2.2 [2]) of the application layer of the TP service sends a request to the corresponding XR manager or the Video manager in the TP edge layer to receive the content. In the case where the requested content is not available already in the edge layer the XR or the Video managers residing at the edge forwards the request to the corresponding managers residing at the TP cloud to receive the content.
9. Once the requested content starts streaming the viewing on the mobile begins.
10. End of the test

### 7.4.7 Measurements transmitted to the database

As defined in Table 4 of Section 2.4, the metrics that will be collected and stored in the database are the following: service, user, speed, coordinates, interruption, interruptionflag, bandwidth, fps, reliability, applicationlatency, UL latency, DL latency and timestamp. Since the user does not move in this case, speed will have zero value.

### 7.4.8 Test results and success criteria

Successful if the content selected streams without interruptions and the transmitted metrics are inside the thresholds of the target values defined in D2.1 [1].

## 7.5 UC4-TC 08

Repeat UC4-TC7 with n=0 static users and m=2 users in motion with cell handover within the same 5G network.

### 7.5.1 Test case type

Performance evaluation of the TP service of UC4.

### 7.5.2 Test case pre-conditions

UC4-TC7 to be successful.

### 7.5.3 Checklist

- 2 Users.
- 2 Mobile Devices.
- 2 5G Sim cards.
- 1 Vehicle.
- 2 Edge servers managed by the same operator.

### 7.5.4 Test case purpose

This test case involves two users who are moving with the same means of transportation (i.e. car) and both are consuming high-resolution 360 video streaming or VR media content and thus interfering with each other since they are using the same base station. Moreover, migration between edge nodes is taking place as they move from one base station to another, demonstrating the "Follow-me" concept.

### 7.5.5 Equipment involved

Testing infrastructure ready and application deployed. Mobile devices can connect to the 5G network and allow the user to interact with the TP service.

#### 7.5.5.1 Vehicles

This test case involves two users in motion, therefore a vehicle is required. For every vehicle involved:

- Purpose: The purpose of the vehicle is to transport two users from base station A to base station B.
- Hardware needed – Nord One Plus mobile phones.
- Software needed – TP mobile application installed in the mobile devices and able to access the TP service.

- People involved: Car driver is required.

##### 7.5.5.2 Networks

In addition to the common test cases network infrastructure presented in sub-section 7.2.5.2, for this specific test case 08 for UC4 it was used:

- 5G network available through the testing area:
- Hardware needed: Two edge servers managed by two different operators.
- Software needed: NearbyOne Orchestrator to deploy the services.
- People involved: Not required.

### 7.5.6 Test steps

1. The users board the car located at base station A and connect to the network through the mobile devices.
2. They open the TP service and accept all permissions if prompted (by tapping on 'while using the app').
3. They are signing in with their credentials (username and password).
4. Throughout this session the calculated metrics are uploaded to the TP cloud server every 5 seconds for storage and further evaluation, and usage from the project's components (Analytics module, Prometheus etc.).
5. Then they tap on the "+" icon (middle icon on the bottom navigation bar) to create predefined tours or the 'Explore' icon on the bottom navigation bar (far right icon) to navigate through the map.
6. The users choose to navigate through the map (the approach would be similar in the other case). So, once they find their Point of Interest (POI) they select it.
7. One to three available types of videos between high resolution, 360 and VR videos appear in the next page. In the case the users want to stream a 360 or a VR video they should wear the head-mounted device at this point.
8. Then the users click on the blue buttons displaying the name of the videos (the availability of the videos depends on each specific site). Consequently, the XR player or the Video player modules (for details of the TP functional block see Fig. 14 in D2.2 [2]) of the application layer of the TP service send a request to the corresponding XR manager or the Video manager in the TP edge layer to receive the content. In the case where the requested content is not available already in the edge layer the XR or the Video managers residing at the edge forwards the request to the corresponding managers residing at the TP cloud to receive the content.
9. Once the requested content is available the viewing begins.
10. The car starts moving towards base station B while the users are watching their preferred media content.
11. At some point the service migration takes place and the video stream happens through the new edge node. This is done seamlessly to the user, that is, there should not be any interruptions.
12. Any further request will be forwarded in the new edge node.
13. End of test.

### 7.5.7 Measurements transmitted to the database

As defined in Table 4 of Section 2.4, the metrics that will be collected and stored in the database are the following: service, user, speed, coordinates, interruption, interruptionflag, bandwidth, fps, reliability, applicationlatency, UL latency, DL latency and timestamp.

### 7.5.8 Test results and success criteria

Successful if the content selected streams without interruptions and the transmitted metrics are inside the thresholds of the target values defined in D2.1 [1].

## 7.6 UC4-TC 09

Repeat UC4-TC8 with n=0 static users and m=2 users in motion in a cross-border scenario.

### 7.6.1 Test case type

Performance evaluation of the TP service of UC4.

### 7.6.2 Test case pre-conditions

UC4-TC8 to be successful.

### 7.6.3 Checklist

- 2 Users.
- 2 Mobile Devices.
- 2 5G Sim cards.
- 1 Vehicle.
- 2 Edge servers managed by different operators.

### 7.6.4 Test case purpose

This test case repeats UC4-TC8 but in this instance the users are crossing the borders and thus roaming is also necessary since different MNOs provide the network service in the two countries. Furthermore, migration of the content between edge nodes will also be tested but in a cross-border scenario.

### 7.6.5 Equipment involved

Testing infrastructure ready and application deployed. Mobile devices can connect to the 5G network and allow the user to interact with the TP service.

#### 7.6.5.1 Vehicles

This test case involves two users in motion, therefore a vehicle is required.
For every vehicle involved:

- Purpose: The purpose of the vehicle is to transport two users from base station A to base station C.
- Hardware needed – Nord One Plus mobile phones.
- Software needed – TP mobile application installed in the mobile devices and able to access the TP service.
- People involved: Car driver is required.

#### 7.6.5.2 Networks

In addition to the common test cases network infrastructure presented in sub-section 7.2.5.2, for this specific test case 09 for UC4 it was used:

- 5G network available through the testing area.
- Hardware needed: Two edge servers managed by two different operators.
- Software needed: NearbyOne Orchestrator to deploy the services.
- People involved: Not required.

### 7.6.6 Test steps

1. The users board the car and connect to the network through the mobile devices.
2. They open the TP service and accept all permissions if prompted (by tapping on 'while using the app').
3. They are signing in with their credentials (username and password).
4. Throughout this session the calculated metrics are uploaded to the TP cloud server every 5 seconds for storage and further evaluation, and usage from the project's components (Analytics module, Prometheus etc.).
5. Then they tap on the "+" icon (middle icon on the bottom navigation bar) to create predefined tours or the 'Explore' icon on the bottom navigation bar (far right icon) to navigate through the map.
6. We assume that the users choose to navigate through the map (the approach would be similar in the other case). So, once they find their Point of Interest (POI) they select it.
7. One to three available types of videos between high resolution, 360 and VR videos appear in the next page. In the case the users want to stream a 360 or a VR video they should wear the head-mounted device at this point.
8. Then the users click on the blue buttons displaying the name of the videos (the availability of the videos depends on each specific site). Consequently, the XR player or the Video player modules (for details of the TP functional block see Fig. 14 in D2.2 [2]) of the application layer of the TP service send a request to the corresponding XR manager or the Video manager in the TP edge layer to receive the content. In the case where the requested content is not

available already in the edge layer the XR or the Video managers residing at the edge forwards the request to the corresponding managers residing at the TP cloud to receive the content.

9. Once the requested content is available the viewing begins.
10. The car starts moving towards Base station C while the users are watching their preferred media content.
11. At some point the service migration takes place and the video stream happens through the new edge node handled by a different operator. This is done seamlessly to the user, that is, there should not be any interruptions.
12. Any further request will be forwarded in the new edge node.
13. End of test.

### 7.6.7 Measurements transmitted to the database

As defined in Table 4 of Section 2.4, the metrics that will be collected and stored in the database are the following: service, user, speed, coordinates, interruption, interruptionflag, bandwidth, fps, reliability, applicationlatency, UL latency, DL latency and timestamp.

### 7.6.8 Test results and success criteria

Successful if the content selected streams without interruptions and the transmitted metrics are inside the thresholds of the target values defined in D2.1 [1].

## 7.7 UC4-TC 10

Repeat UC4-TC9 with n=0 static users and m=2 users in motion in a cross-border scenario and high speed. All requirements, steps and results are the same as in Test Case 9.

## 7.8 UC4-TC 11

Repeat UC4-TC10 with n=0 static users and m>=2 users in motion in a cross-border scenario, high speed and high density. All requirements, steps and results are the same as in Test Case 9.

# 8. Conclusions

Deliverable D6.1 serves as a comprehensive package, encompassing pivotal elements to conduct the experiments of the small-scale trials. Firstly, it offers the methodology designed to measure service Key Performance Indicators across the entire spectrum of use cases. Secondly, it establishes a clear data format for transmitting measurements to the centralized KPI database platform, ensuring seamless data integration and analysis. Lastly, it furnishes an elaborate description of the test cases, meticulously crafted to facilitate the execution of small-scale trials and experiments. The outcomes and discoveries stemming from the test cases elaborated in this document will be diligently reported in D6.2.

Funded by the Horizon 2020 Framework Programme of the European Union

# 9. References

[1] 5GMED, «D3.3. First release of 5GMED».

[2] 5GMED, «D5.2. Initial design for FRMCS and railways infotainment test cases».

[3] 5GMED, «D2.1. Use Case story definition, requirements and KPIs».

[4] 5GMED, «D2.2 Initial definition of 5GMED test cases, deployment options and tools».