INTEGRATION AND VALIDATION OUTCOMES

Deliverable D4.4

Co-funded by the European Union

6GSNS

The TARGET-X project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No: 101096614

Deliverable

INTEGRATION AND VALIDATION OUTCOMES

GRANT AGREEMENT	101096614
PROJECT TITLE	Trial Platform foR 5G EvoluTion – Cross-Industry On Large Scale
PROJECT ACRONYM	TARGET-X
PROJECT WEBSITE	www.target-x.eu
PROGRAMME	HORIZON-JU-SNS-2022-STREAM-D-01-01 — SNS Large Scale Trials and Pilots (LST&Ps) with Verticals
PROJECT START	01-01-2023
DURATION	34 Months
DELIVERABLE TYPE	Deliverable
CONTRIBUTING WORK PACKAGES	WP4
DISSEMINATION LEVEL	Public
DUE DATE	M33
ACTUAL SUBMISSION DATE	M33
RESPONSIBLE ORGANIZATION	I2CAT Foundation
EDITOR(S)	Jad Nasreddine
VERSION	1.0
STATUS:	Final
SHORT ABSTRACT	This deliverable presents the outcomes of the use case validation, including the detailed, end-to-end description of the experiments. The performance achieved will be compared to the KPIs established in T4.1 and recommendations for future enhancements of the 5G architecture towards 6G, as well as for the design of novel automotive vertical services will be provided.
KEYWORDS	Testbed, 5G, MEC, use case validation, KPIs automotive, digital twins, cooperative perception, predictive QoS, tele-operated driving.
CONTRIBUTOR(S)	Cristian Armesto (NEUTROON), Jad Nasreddine (I2CAT), Mahdi Darroudi (NEUTROON), Martín Trullenque (I2CAT), Paul Salvati (IDIADA), Ramon Roca Oliver (I2CAT)

Disclaimer

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the other granting authorities. Neither the European Union nor the granting authority can be held responsible for them.

Executive Summary

This deliverable presents the outcomes of the integration and validation activities conducted within the Work Package 4 of TARGET-X project, focusing on the deployment and assessment of advanced 5G-enabled use cases in automotive and industrial environments. Six use cases were implemented, including cooperative perception, digital twins, predictive quality of service for tele-operated driving, Visibility, Insights, Signal Telemetry, and Analytics (VISTA)-based network monitoring, remote power consumption monitoring, and remote environment monitoring. Each use case was validated through field trials at the IDIADA proving grounds, integrating real vehicles, edge/cloud infrastructure, network exposure Application Programming Interfaces (APIs), and dynamic service orchestration mechanisms. Performance was measured using Key Performance Indicators (KPIs) such as latency, reliability, throughput, jitter, replicability, and computational efficiency. Results demonstrate significant improvements in service reliability, responsiveness, and resource efficiency, validating the proposed solutions and confirming their suitability for next-generation connected and automated mobility applications.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

3GPP	3rd Generation Partnership Project
5GC	5G Core
AEF	Action Enforcement Function
AI	Artificial Intelligence
API	Application Programming Interface
AWS	Amazon Web Services
CAM	Cooperative Awareness Messages
CAV	Connected and Automated Vehicle
C-ITS	Cooperative Intelligent Transport Systems
CPE	Customer Premise Equipment
DMF	Decision Making Function
E2E	End to End
FSTP	Financial Support for Third Parties
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HMI	Human Machine Interface
CPM	Collective Perception Messages
CWS	The Collision Warning Service
DENM	Decentralized Environmental Notification Message
DFT	Dynamic Phasor Conversion
IVS	In-Vehicle System
KPI	Key Performance Indicator
LDMS	Local Dynamic Map Service
MEC	Mobile Edge Computing
MEF	Metric Exposure function
MQTT	Message Queuing Telemetry Transport
NDT	Network Digital Twin
NR	New Radio
PF	Prediction Function
PoC	Proof of Concept
pQoS	Predictive QoS
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RSSI	Received Signal Strength Indicator
RTK	Real-time kinematic positioning
RTT	Round Trip Time
RTSP	Real-Time Streaming Protocol
SLA	Service-Level Agreement
SINR	Signal to Interference and Noise Ratio
SNR	Signal to Noise Ratio

SDSM	Sensor Data Sharing Message
TAC	Tracking Area Code
TDD	Time Division Duplex
TMC	Traffic Management Center
ToD	Tele-operated Driving
ToV	Tele-operated Vehicle
UC	Use Case
UE	User Equipment
UPF	User Plan Function
VM	Virtual Machine
VISTA	Virtualized Infrastructure for Smart Transport Applications
V2I-DSS	Vehicle – Infrastructure Data Sharing Service
V2X	Vehicle to everything
ViL	Vehicle-in-the-Loop
WP	Work Package

1 Introduction

This document, Deliverable D4.4, presents the final results of the activities carried out within WP4 of the TARGET-X project. It builds upon the design specifications and component development reported in Deliverables D4.1 [1], D4.2 [2], and D4.3 [3], and constitutes the culmination of the experimental phase.

The primary objective of WP4 is to integrate the developed 5G-enabled components, services, and platforms into complete end-to-end (E2E) automotive systems and validate their performance through large-scale field trials. These trials were executed at IDIADA proving grounds representing realistic and operational environments. The trials cover a broad set of six use cases (UCs) targeting automotive applications, namely: Cooperative Perception for connected vehicles, digital twin for vehicle monitoring and control, predictive Quality of Service (QoS) for tele-operated driving, Visibility, Insights, Signal Telemetry, and Analytics (VISTA)-based network monitoring, remote power consumption monitoring, and remote environmental monitoring.

Each use case integrates various 5G Core (5GC) and Radio Access Network (RAN) components, Mobile Edge Computing (MEC) and cloud computing infrastructures, service orchestration frameworks, and network exposure Application Programming Interface (API). The integration activities involved multiple partners and technologies, requiring close coordination to ensure interoperability and performance alignment.

The validation phase focused on assessing Key Performance Indicators (KPIs) such as latency, reliability, throughput, jitter, computational efficiency, replicability, and energy efficiency. These KPIs were measured under realistic operational conditions, reflecting safety-critical, time-sensitive, and resource-intensive scenarios. The results provide empirical evidence of the technical maturity of the developed solutions and their potential for scalable deployment in future 5G and beyond (B5G) networks.

Furthermore, the deliverable captures the lessons learned during the WP lifespan, troubleshooting, and cross-partner collaboration, which are essential to guide future deployments, standardization contributions, and industrial adoption of the TARGET-X outcomes.

1.1 Relation to other activities

In addition to the primary focus of WP4 efforts, this document also details the collaboration between WP4 and WP3 to implement and evaluate use case 5: Remote power consumption monitoring. Furthermore, it provides the results of the integration of TARGET-X dynamic service orchestrator (See D4.3 [3]) with one of the Financial Support for Third Parties (FSTP) projects, Impact-xG, in use case 6: Remote environmental monitoring.

1.2 Document structure

The document is structured as follows: Section 2 provides a brief reminder of the use cases, their components, and requirements. In addition, the measurement methodology and the testcases are detailed in this section. Section 3 presents and discusses the outcomes of the validation tests performed at IDIADA for the six use cases. Section 4 discusses the lessons learned collected throughout all the phases of WP4. Section 5 concludes the document.

2 Trials Summary

For each WP4 use case, this section provides a brief description of the use case (more information can be found in Deliverable D4.1 [1]), the main components used to deploy the use case, the measurement methodology, and a description of the testcases.

2.1 Use case 1: Cooperative perception

2.1.1 Use case description

The cooperative perception use case is based on two scenarios, one based on an intersection with no visibility and the second is developed in a straight line with adverse climatic conditions (fog, heavy rain, etc.) that difficult the visibility on the road.

2.1.1.1 Scenario 1: Zero Visibility Intersection

Figure 1 shows two vehicles that are approaching an intersection with zero visibility. To ensure safety, these vehicles share their position, speed, and vehicle information with the Cooperative Intelligent Transport Systems (C-ITS) platform through Cooperative Awareness Messages (CAMs).

Figure 1: UC1 representation [1].

The C-ITS is deployed over the cloud using Amazon Web Services (AWS) in Malaga or over the Edge (Edge Server) and is accessible through via IDIADA 5G network. In this use case, the Collision Warning Service (CWS) continuously monitors CAMs from both vehicles to assess potential collision risks. When a hazardous situation is detected, the CWS promptly issues an alert, enabling the vehicles to take appropriate evasive actions. The warning is sent through a Decentralized Environmental Notification Message (DENM) message, which is displayed on a Human Machine Interface (HMI) in the CAVride (See Figure 2). Furthermore, the HMI has both the Local Dynamic Map Service (LDMS) and the Vehicle – Infrastructure Data Sharing Service (V2I-DSS) implemented. The V2I-DSS allows the vehicle to receive external data from the environment (from other vehicles); the In-Vehicle System (IVS) data from the sensors and the external data from the network will be merged so that both the environment and the warning events sent from the CWS can be seen on the vehicles' HMI.

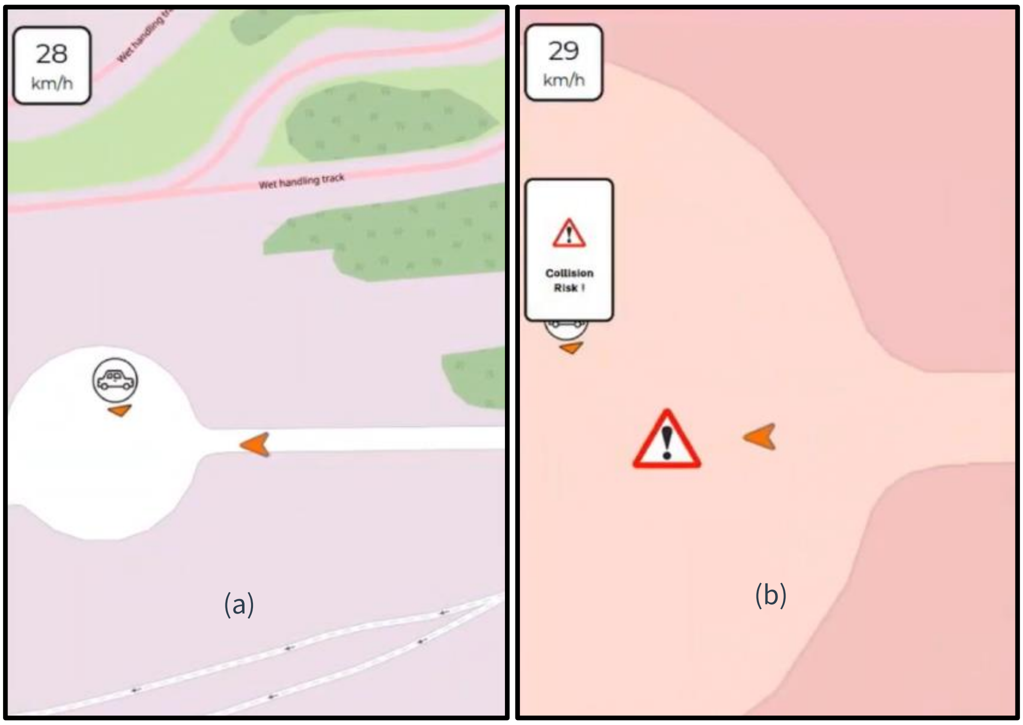


Figure 2: The HMI in the vehicle of UC 1 scenario 1 when (a) no DENM warning has been received and (b) when a DENM warning has been received.

2.1.1.2 Scenario 2: Road Damaged Vehicle

Figure 3 shows a roadside damaged vehicle in the same path as another vehicle in movement. The complexity in this scenario is the lack of visibility due to weather conditions (e.g., heavy rain, fog, or snowfall). Thanks to the CAMs and Collective Perception Messages (CPMs) messages sent by the vehicles themselves (both from the damaged vehicle and the moving vehicle) through the 5G network, the CWS can identify the risk and notify the vehicles using DENM messages of the existence of a damaged vehicle on the same trajectory to avoid a possible accident.

In this use case, all connected and automated vehicles (CAVs) can generate CAM and CPM messages, which are transmitted to the C-ITS and then relayed to nearby vehicles through the 5G network. The value of this use case lies in the ability of modern vehicles to gather extensive information about themselves and their surroundings. By processing and interpreting this data, potential events can be predicted in advance. This enables vehicles to receive the right information at the right moment, improving their situational awareness and allowing them to adapt their driving strategies—ultimately enhancing road safety.

A key distinction between CAM and CPM messages is that CPMs are generated by a vehicle's perception systems (or by roadside infrastructure), such as lidar, radar, or cameras. In Scenario 1, CPMs are not used because the zero-visibility intersection prevents the perception systems from detecting the other vehicle. Instead, CAM messages are employed to share information with the C-ITS platform. When weather conditions provide good visibility, a vehicle can detect the damaged vehicle and transmit CPMs to the C-ITS. However, under poor visibility, the CWS relies on CAMs sent by the damaged vehicle, which is identified as such because it remains stationary. This approach is feasible only when the damaged vehicle is connected to the C-ITS. Accordingly, the implemented CWS integrates both approaches, and this document presents the performance results for each case.

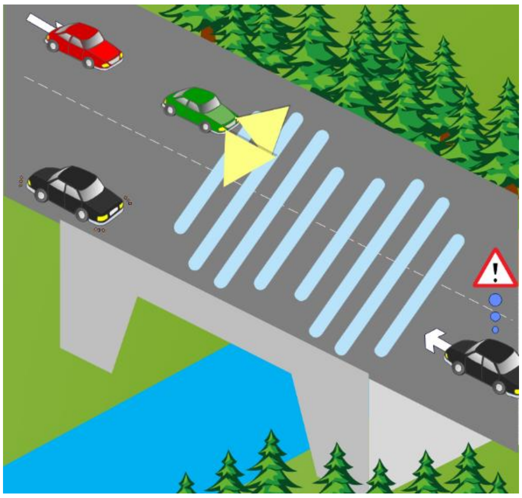


Figure 3: UC2 representation [1].

The DENM will be displayed by the vehicle through its HMI (See Figure 4), which runs both the LDMS and the V2I-DSS to represent the situation. Once the HMI receives the DENM, the vehicle in motion can adjust its driving strategy, taking appropriate precautions in response to the reduced visibility and the newly detected event.

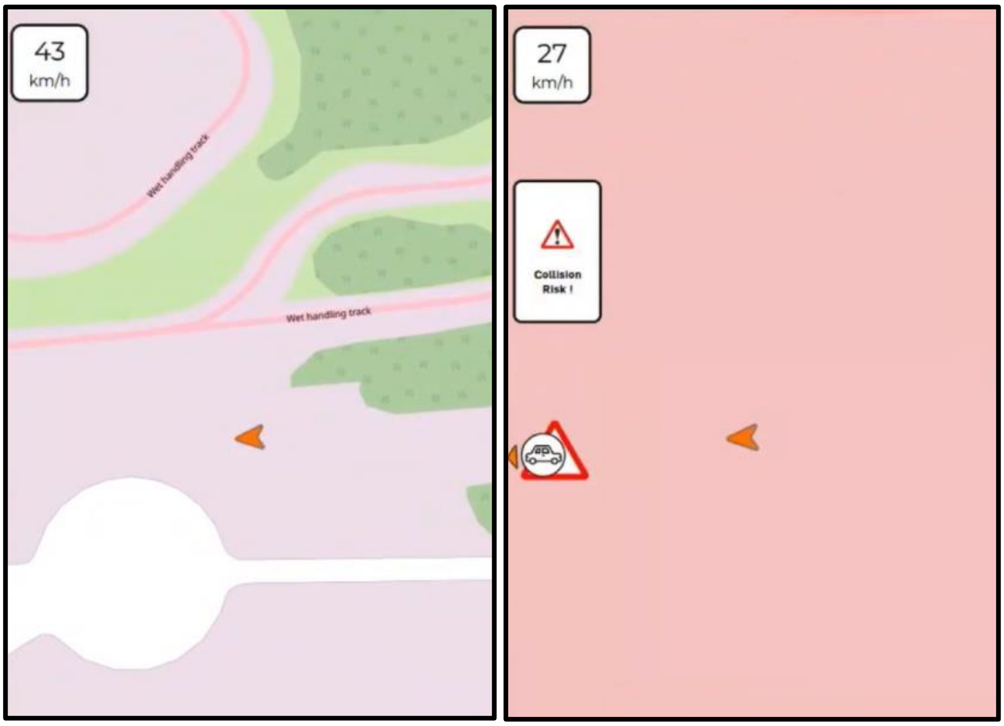


Figure 4: The HMI in the vehicle in UC 1 scenario 2 when (a) no DENM warning has been received and (b) when a DENM warning has been received.

2.1.2 UC1 components

Each vehicle has a 5G router that allows connectivity to the C-ITS platform, a Global Navigation Satellite System/Global Positioning System (GNSS/GPS) system with Real-time kinematic positioning (RTK) corrections that gives a high precision location and a computer to run the vehicle

C-ITS client. Detailed information about the architecture is showed in Figure 5 and detailed in D4.2 [2].

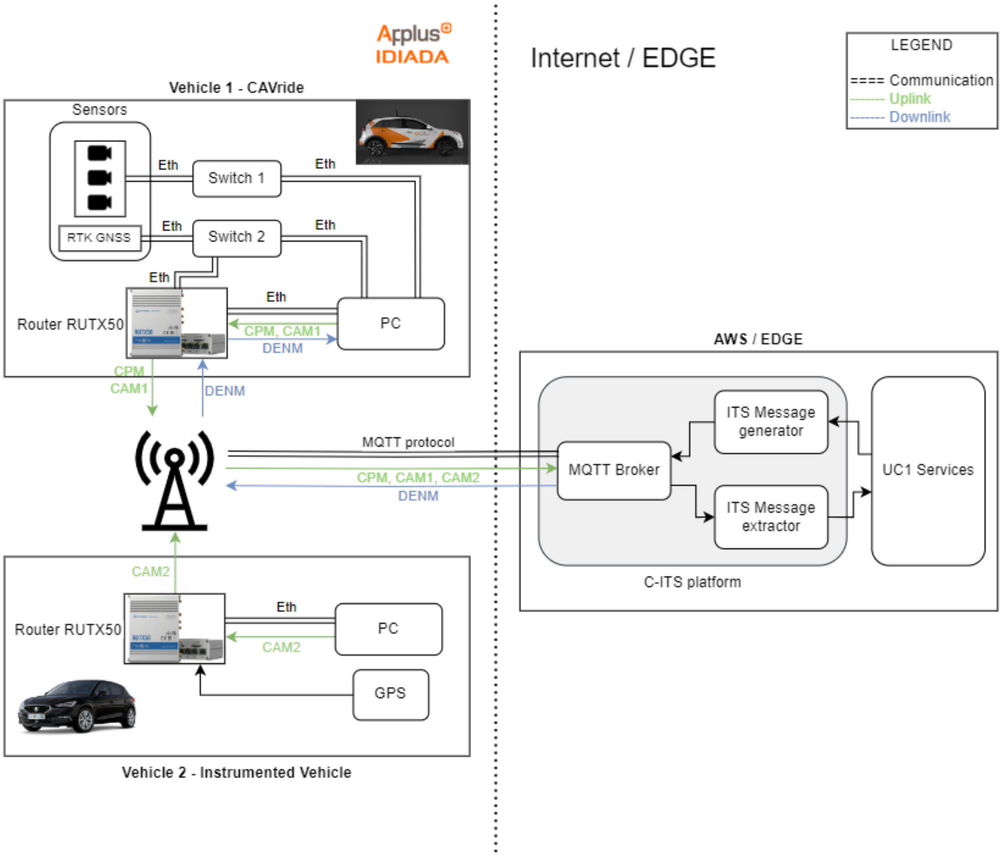


Figure 5: Functional Architecture of the cooperative perception architecture [2].

2.1.3 Measurement methodology and service KPIs

For UC 1, Figure 6 shows that inside the C-ITS platform runs three services for CAM, CPM and DENM that get the data from the vehicles and publish all the information in a Message Queuing Telemetry Transport (MQTT) broker. Behind this broker, the Service KPI generator built all the KPIs using Grafana for visualization.

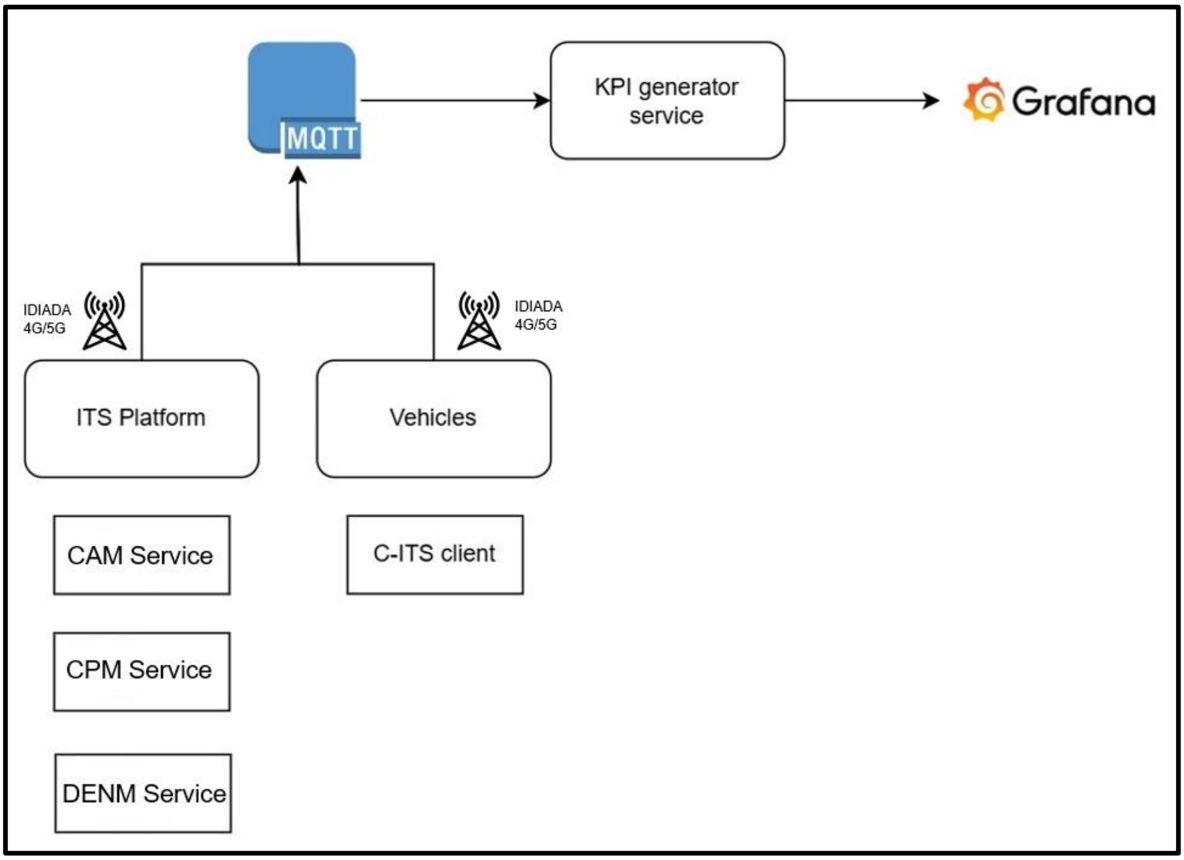


Figure 6: Measurement Methodology UC1.

The detailed KPI information for UC1 is showed in Table 1. As it will be explained in Section 2.1.4, each testcase will be repeated 15 times or rounds. The thresholds are those defined in D4.1 [1] except for the DENM TRIGGER LATENCY that is the sum of CAM/CPM latency, DENM latency, and processing time which is estimated to be around 100 ms.

Table 1: Detailed KPI information for UC1.

KPI	Description	Unit	Periodicity	Procedure	Threshold
CAM_Success_Rate	Percentage of CAM generated by a vehicle and received by the CWS.	%	Per round	This KPI is computed by comparing the number of CAM messages sent by each vehicle with the number received by the CWS. The percentage is calculated over the total CAM sent.	99.80%
CPM_Success_Rate	Percentage of CPM generated by a connected vehicle and	%	Per round	This KPI is computed by comparing the number of CPM messages sent by the obstacle-detecting vehicle with	99.80%

	received by the CWS.			the number received by the CWS. The percentage is derived from total CPM messages sent.	
DENM_Success_Rate	Percentage of DENM generated by infrastructure and received by the CWS.	%	Per round	This KPI is determined by comparing the number of DENM messages generated by the infrastructure with those received by the vehicles. The percentage is derived from total CPM messages sent.	99.80%
CAM_Latency	Time elapsed between CAM message generation and reception by the CWS.	ms	Per round	This is measured by comparing the *GenerationDeltaTime* field of CAM messages with their reception timestamps at the clients.	40-50 ms
CPM_Latency	Time elapsed between CPM message generation and reception by the CWS.	ms	Per round	This is measured by comparing the *referenceTime* field of the CPM content with its reception time at the client.	40-50 ms
DENM_Latency	Time elapsed between DENM message generation and reception by the client.	ms	Per round	This is measured by comparing the *referenceTime* field of the DENM content with the timestamp of reception at the client.	40-50 ms
DENM TRIGGER LATENCY	Time elapsed between DENM reception by the vehicle and the original CAM/CPM that triggered the collision warning.	ms	Upon collision detection	This is calculated by cross-referencing the DENM reception time with log entries from the Collision Warning Service that include the triggering CAM or CPM messages detected as relevant to the event.	200 ms

2.1.4 Description of the testcases

Scenario 1 is evaluated at the intersection of the testing track called Urban Area showed in Figure 7 and Figure 8.

Figure 7: Urban Area intersection where UC 1- scenario 1 is evaluated.

Figure 8: Urban Area intersection (Zoom) where UC 1- scenario 1 is evaluated..

It should be noted that in the demonstration video we did not use the same area because there is no blocks between the two roads. Instead we used the area showed in Figure 9 of the same testing track. In the area shown in Figure 9 of the test track, we can reproduce an intersection with a natural block between both roads, and we can prove the zero-visibility zone.

Figure 9: Intersection with blocked roads used for the video demonstration of UC 1 scenario 1.

The details of the testing KPIs, numbers of rounds, and more information are showed in Table 2.

Table 2: UC1 - Scenario 1 - Table information.

UC1 - SCENARIO 1	
SCENARIO	INTERSECTION
ROUNDS	15
SPEED	30Kmh
DETECTION	CAM
TEST EXECUTION	2hs
KPI 1	CAM Success Rate
KPI 2	DENM Success Rate
KPI 3	CAM Latency
KPI 4	DENM Latency
User KPI	DENM TRIGGER LATENCY

UC1, scenario 2 is evaluated in the same area of the testing track but using a straight road. In this case, it is impossible to reproduce the adverse climatic conditions, but perception and detection of the cars is performed using CAMs or CPMs as explained previously. The area used to perform the UC1 – SC2 is showed in Figure 10 and Figure 11.

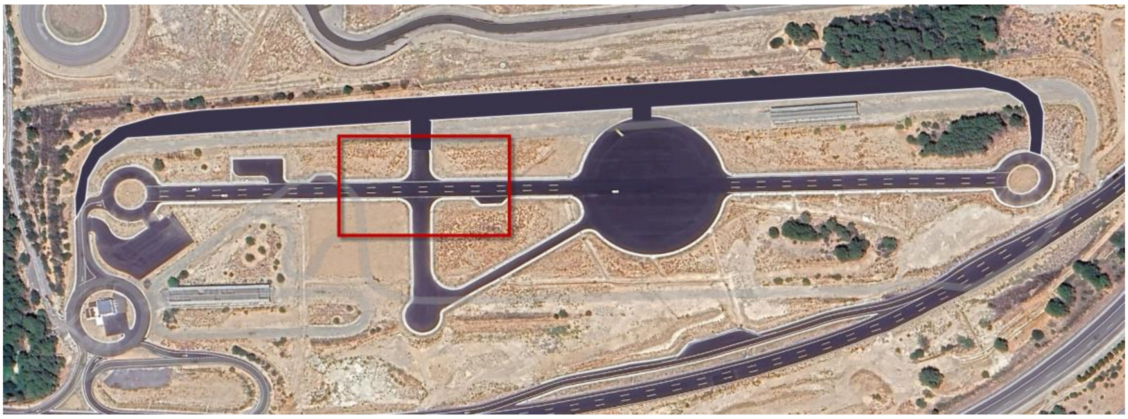


Figure 10: Urban Area straight road where UC 1- scenario 2 is evaluated.

Figure 11: Urban Area straight road (Zoom) where UC 1- scenario 2 is evaluated.

The details of the testing KPIs, numbers of rounds and more information is showed in Table 3 and Table 4 for both cases when CAM and CPM are used for detection.

Table 3: UC1 - Scenario 2 - Table information - CAM.

UC1 - SCENARIO 2	
SCENARIO	STRAIGHT LINE
ROUNDS	15
SPEED	50Kmh
DETECTION	CAM
TEST EXECUTION	2 hs
KPI 1	CAM Success Rate
KPI 2	DENM Success Rate
KPI 3	CAM Latency
KPI 4	DENM Latency
User KPI	DENM TRIGGER LATENCY

Table 4: UC1 - Scenario 2 - Table information – CPM.

UC1 - SCENARIO 2	
SCENARIO	STRAIGHT LINE
ROUNDS	15
SPEED	50Kmh
DETECTION	CPM
TEST EXECUTION	2 hs
KPI 1	CPM Success Rate
KPI 2	DENM Success Rate
KPI 3	CPM Latency
KPI 4	DENM Latency
User KPI	DENM TRIGGER LATENCY

2.2 Use case 2: Digital twins

The Digital Twin allows to merge the physical and digital worlds by introducing data from the real world into a simulation and the other way around. In particular, the TARGET-X Digital Twin is focused on saving the V2X data exchanged between the ITS station in a certain road segment or scenario, allowing to use such data for simulations including more complex elements (e.g. autonomous vehicles) and/or for testing of vehicle functions in the real world by means of a Vehicle-In-the-Loop (ViL) validation approach.

2.2.1 Use case description

UC2 reproduces a Digital Twin from UC1 - Scenario 1, in which a digital object (like a vehicle) will be added to the system and showed in the HMI so that the system reacts in the same way as it will do it if a real vehicle should be involved.

Also known as "Replay", the Digital Twin will allow to capture and save all V2X communications from a scenario and replay them at any time, to (virtually) reproduce the scenario again.

2.2.2 UC2 components

Figure 12 shows the high-level architecture of the solution to provide replay functionality, where an external stack will provide a database, a "Recorder", and a "Replay" service.

The recorder is able to connect to the external MQTT broker in which all the messages from Geomessaging will be published by Message extractor.

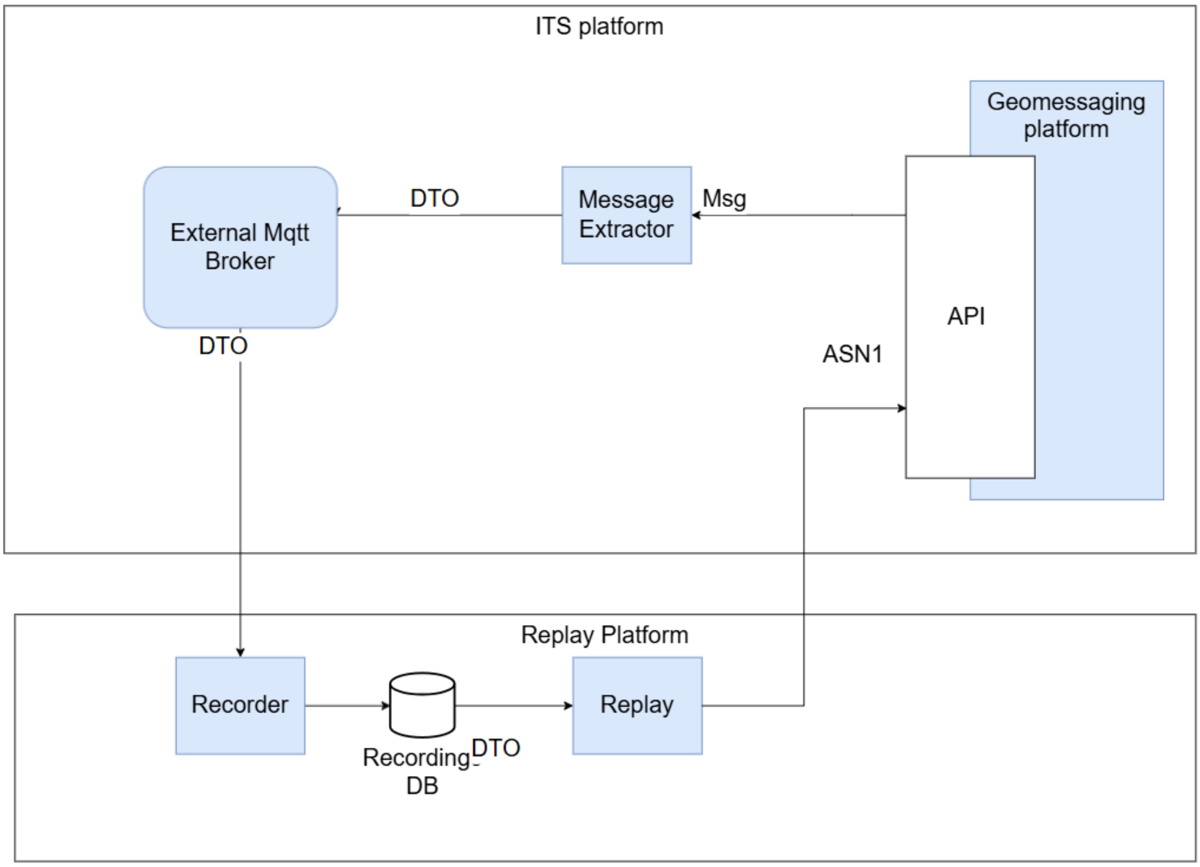


Figure 12: Digital Twin high level architecture.

2.2.3 Measurement methodology and service KPIs

The performance of this use case is evaluated using the replay reliability (See Table 5). This KPI measures the deviation in the periodicity between the original messages stored in the MQTT broker and the replayed messages by the Digital For instance if Original Msg1 is sent at 12:00:00 and Msg2 at 12:00:01 with 1-second interval, and the replayed Msg1 is sent at 15:00:05 and Msg2 at 15:00:07 with 2-seconds interval, the deviation will be 1000 ms. This computation is performed for all replayed message sequences. The final KPI includes the average, standard deviation, maximum, and minimum deviation values.

Table 5: Detailed KPI information for UC2.

KPI	Description	Unit	Periodicity	Threshold
REPLAY_RELIABILITY	Measures the average deviation in message periodicity between the original and the replayed messages.	ms	Per round	10 ms

2.2.4 Description of the testcases

By recording the performance of a vehicle during the UC1 – Scenario 1 tests, we can replay its actions as a "vehicle-in-the-loop" to recreate the same scenario while modifying key parameters, such as speed, of the real vehicle. This approach enables us to test vehicles at higher speeds without any risk of collision. The area where UC2 was implemented using modified parameters derived from UC1 is illustrated in Figure 13 and Figure 14.

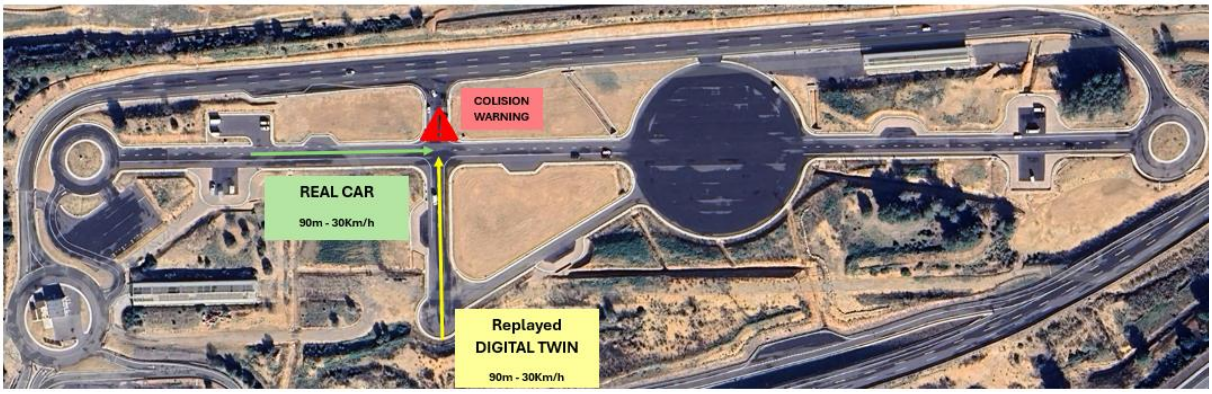


Figure 13: Urban Area intersection selected for UC2 – 30 km/h.

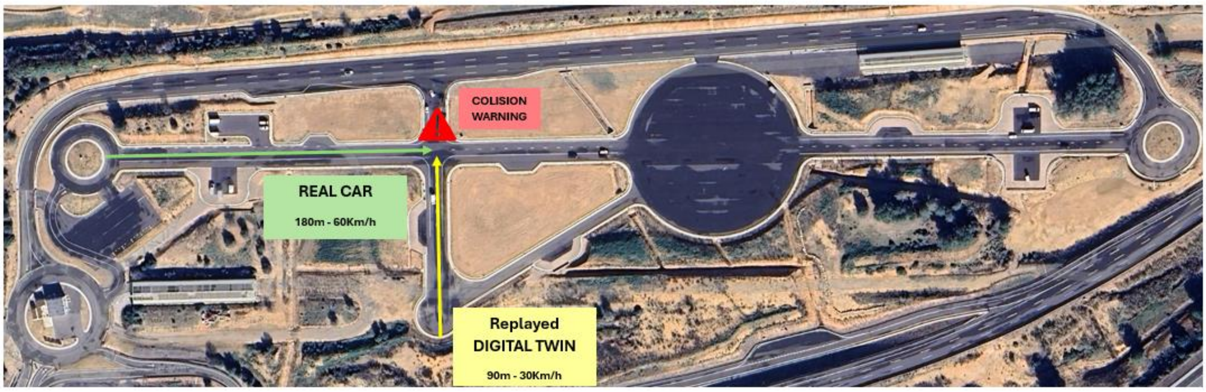


Figure 14: Urban Area intersection selected for UC2 – 60 km/h.

2.3 Use case 3: predictive QoS for Tele-operated driving

2.3.1 Use case description

Without visibility into network conditions, a Tele-Operated Vehicle (ToV) could become stranded if connectivity issues arise in a specific area, requiring manual intervention or even a tow truck. To address this, the ToV leverages information made available by the TARGET-X network exposure interface. With the help of the prediction function developed in TARGET-X, the remote driver is alerted in advance, enabling timely breaking or rerouting decisions. This approach helps avoid abrupt stops, reducing both inconvenience and the risk of road obstruction.

In D4.1 and D4.2m this UC was described deeply and here we are going to give a brief description of it. Two objectives are wanted in UC3:

- To evaluate the performance of Tele-Operated Driving (ToD) over 5G networks.
- To highlight the importance of accessing network performance data via advanced 5G capabilities, such as network exposure Application Programming Interfaces (APIs), to prevent sudden braking during tele-operated driving.

2.3.2 UC3 components

Figure 15 illustrates two parts. On the **left side**, the remote-control room is shown, which includes the steering wheel and pedals, a screen displaying the camera feeds, a laptop used to manage the remote driving, and a 5G router that ensures mobile connectivity between the CAV-Ride and the control room.

On the **right side**, the car components are displayed: three cameras providing a 180° field of view, a drive kit that enables remote control of the vehicle, a laptop for coordinating the remote driving, and a 5G router for maintaining the connection with the control room.

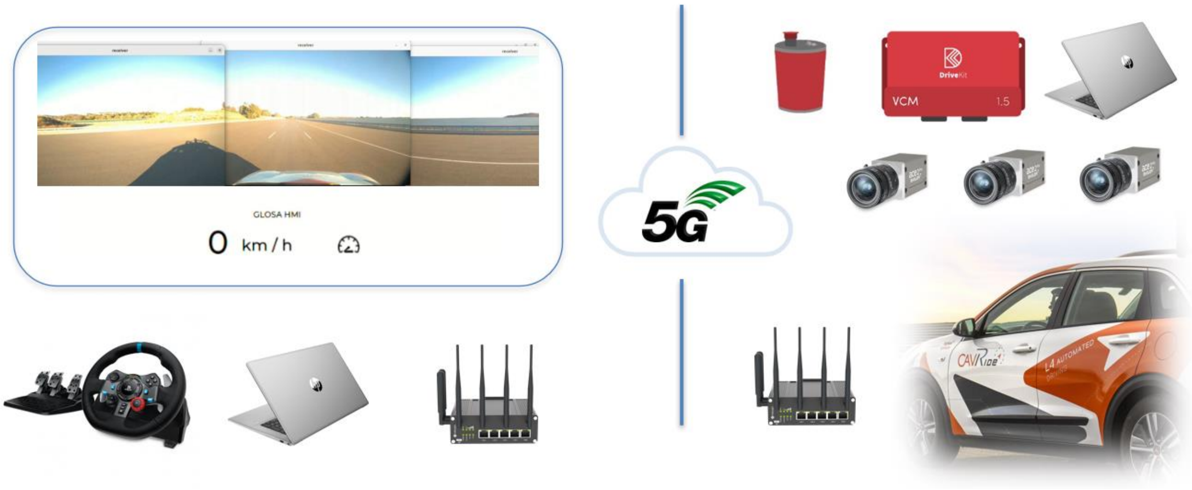


Figure 15: UC3 components.

2.3.3 Measurement methodology and service KPIs

For UC3, the measurements methodology is the same as UC1, but, adding all related services for remote driving as showed in Figure 16.

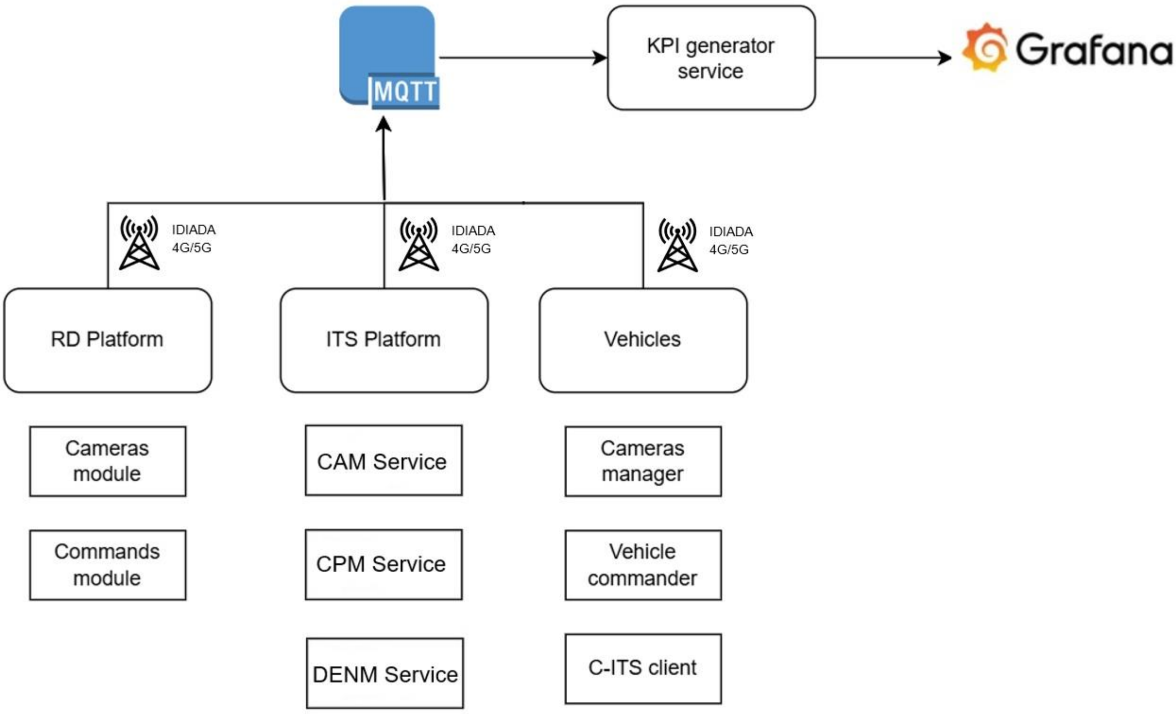


Figure 16: UC3 measurements methodology

The evaluated KPIs for UC3 is showed in Table 6. The thresholds are those defined in D4.1 [1] except for the video jitter that is added here for its importance to the quality of experience perceived by the remote driver.

Table 6: UC3 detailed KPI information.

KPI	Description	Procedure	Threshold
Command_Latency	Time elapsed between command generation and reception by the vehicle.	Commands are tagged with the generation timestamp in the header. Upon reception, the vehicle compares the timestamp with the current time to compute latency.	20-50 ms
Data transfer Latency	Time elapsed between video generation and reception by the remote center.	Each video frame is tagged with its generation time. The receiver uses this to calculate the time taken to receive a complete frame.	100 ms
Command_Reliability	Success rate of commands received by the vehicle.	Each command sent to the vehicle is tagged with a unique incremental counter in the message header. Upon reception, the vehicle checks the sequence to identify any missing commands. The success rate is calculated accordingly.	99%

Data transfer Reliability	Percentage of successfully received video frames.	Each frame has a unique ID (incremental counter). The remote center detects dropped or corrupted frames and calculates the ratio of received frames over total frames sent.	95%
Uplink service data rate	Uplink bandwidth per vehicle camera.	Calculated by summing the total size of video data sent per camera over the network.	10-50 Mbps
Video_Jitter	Variation in video frame latency (jitter).	Computed using the variation between consecutive frame latencies	150 ms [4]

2.3.4 Description of the testcases

UC3 testcases is described in section 6.3 of D4.2, in this deliverable we are going to explain the different stages of this UC.

The UC3 is deployed in the Zone 1 of the urban area test track, showed in a red square on Figure 17.

Figure 17: UC3 testing area.

This area is composed of two roundabouts in which connected and ToV are located and driving. To evaluate the performance of ToD in different conditions, we performed different scenarios.

Figure 18 shows Scenario "**No network problems**" in which the mobile network is all OK, up and running and both vehicles are driving without any kind of problems.

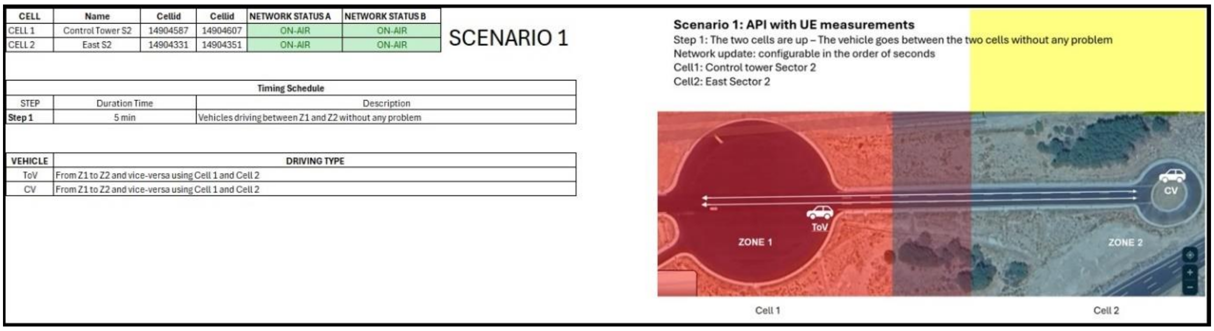


Figure 18: UC3 No network problems scenario.

For "**Sc. 1: Cell down-API activated from CV**", Figure 19 illustrates the setup, where the ToV is positioned in the small roundabout and the connected vehicle is in the larger one, where the cell has been deactivated. Both the network exposure API and the prediction function are enabled. As the ToV approaches the larger roundabout, the prediction function issues a warning to the remote driver's HMI. The driver then follows the recommendation and brings the vehicle to a smooth stop. The prediction function continuously receives real-time information from the connected vehicle regarding the network status.

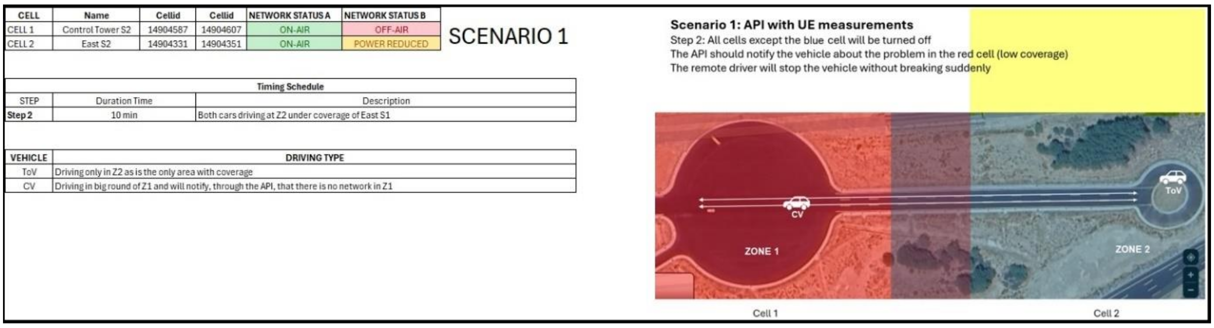


Figure 19: UC3 Sc. 1 - Cell down-API activated from CV.

Figure 20 shows that during "**Sc. 2: Cell down-API de-activated**" the mobile network will remain exactly the same way as in UC3 Sc. 1. However, in this case the network exposure API and prediction function are deactivated, and **NO warning** from the prediction function will be sent. The ToV will continue driving towards the big roundabouts and suddenly will suffer from the lack of coverage.

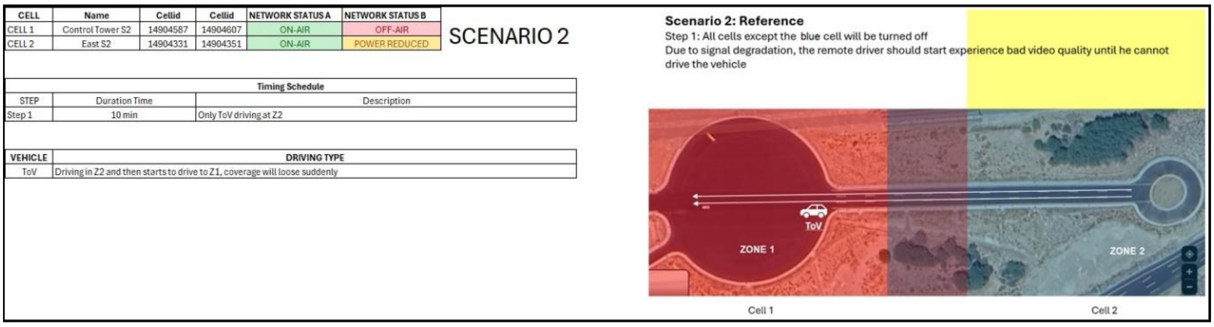


Figure 20: UC3 Sc. 2 - Cell down-API de-activated.

"**Sc. 3: Cell down-API activated from OEM**", illustrated in Figure 21, is the same as Sc. 1 but network status is obtained from the network dashboard instead of the CV. The prediction function gets information about cell status change every 15 minutes.

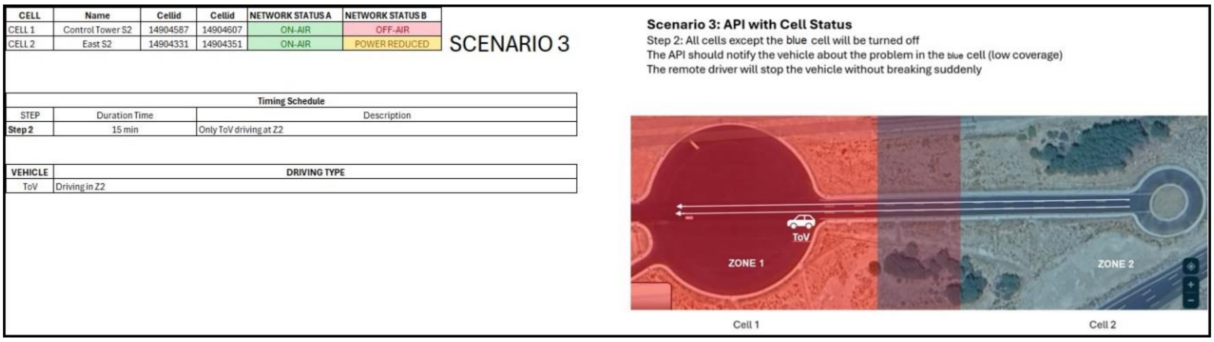


Figure 21: UC3 Sc. 3 - Cell down-API activated from OEM.

Scenario 4, shown in Figure 22, has Mobile Network full on air and with full power. Remote driving performed normally. In contrary to the previous scenarios where the remote system was connected through a 5G modem, this system is connected directly to the User Plane Function (UPF) as an edge system. In this scenario, the ToV changes the attachment to the mobile network between 5G and 4G, referred in the results by "**Sc.4 – 5G**" and "**Sc.4 – 4G**".

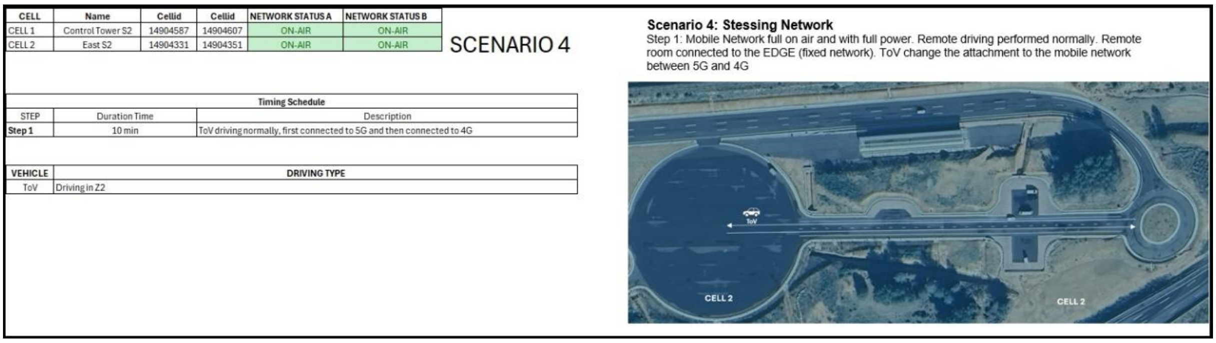


Figure 22: UC3 Scenario 4.

2.4 Use case 4: VISTA

2.4.1 Use case description

VISTA (Visibility, Insights, Signal Telemetry, and Analytics) is a Neutroon-developed tool within the TARGET-X project. It was introduced in section 3.2.1 of D4.3 [3] and it provides a comprehensive view of network performance from the User Equipment (UE) perspective, offering deep insights and analytics for monitoring, troubleshooting, and AI model training. This section describes the executed Proof of Concept (PoC) while running in a vehicle and reports user experience network quality in an urban environment. The use case also integrates the dynamic service orchestrator (Figure 23), developed in TARGET-X and described in D4.3, to switch between two modes: data transmission mode when network conditions allow it, and backup mode where measurements are locally stored when the network conditions do not allow the data to be reliably transmitted.

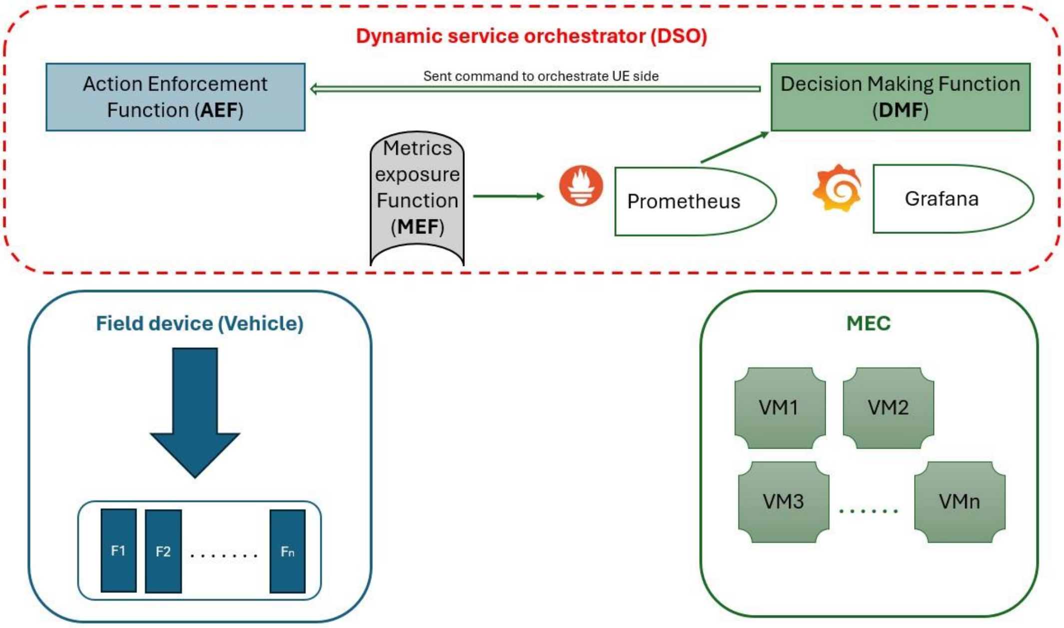


Figure 23: Dynamic service orchestrator, the general architecture [3].

VISTA offers a real-time dashboard for immediate monitoring and can also feed analysis functions within the network, for instance the Decision-Making Function (DMF) in this specific use case. Additionally, the long-term collected network metrics are useful for troubleshooting, identifying root causes of network issues, and serving as a rich dataset for Artificial Intelligence (AI)-model training. The collected metrics can be exposed to third-party applications through the Metrics Exposure Function (MEF) of the dynamic service orchestrator.

The use case involves also designing and developing Customer Premise Equipment (CPE) registered in the network. This CPE does not only provide connectivity but also hosts VISTA and it is used in the TARGET-X project as a PoC.

Apart from the basic mission of VISTA to collect metrics from user experiences point of view, it was also used in use cases 5 and 6 to monitor network connectivity quality and trigger the orchestrator once required (See Section 2.5 and Section 2.6).

2.4.2 UC4 components

Figure 24 shows the manufactured CPE instance with the embedded VISTA tool. This CPE features an Intel®Core™ i3-N305 CPU motherboard, connected to a Quectel RM500Q-GL module for 5G connectivity. Running Ubuntu 22.04, the CPE can easily host applications and services. It also includes two ethernet ports and an added WiFi 6 module (WiFi 6 AX200 NGW) so that the CPE can function as a router for devices behind it.

Figure 24: Developed CPE to host VISTA [3].

The VISTA tool can identify UEs (e.g. a vehicle in this test) with low quality connection and cannot stream metrics to the MEC server through the DMF. Then, it adapts to the situation using the AEF, which can stop streaming collected metrics and store them locally until the connection quality returns to normal.

2.4.3 Measurement methodology and service KPIs

VISTA relies on the MEF that runs in the operational UE and collects connectivity KPIs that are explained in Table 1 of Deliverable D4.3. The evaluated KPIs in this use case are listed in Table 7.

Table 7 MEF connectivity KPIs.

NAME	UNIT
SIGNAL TO INTERFERENCE AND NOISE RATIO (SNR)	dB
REFERENCE SIGNAL RECEIVED POWER (RSRP)	dBm
REFERENCE SIGNAL RECEIVED QUALITY (RSRQ)	dBm
RECEIVED SIGNAL STRENGTH INDICATOR (RSSI)	dBm
CELL NUMBER	-
TRACKING AREA CODE (TAC)	-
ROUND TRIP TIME (RTT) TO MEC	ms
RTT TO CLOUD	ms
USER EXPERIENCE DATA RATE (DOWNLINK)	Mbps
USER EXPERIENCE DATA RATE (UPLINK)	Mbps

Figure 25 shows how exposed metrics from network modules in the UE are collected by MEF and are sent to Prometheus server that is placed in a virtual environment on the MEC server. These metrics are passed to Grafana for monitoring and reporting purposes.

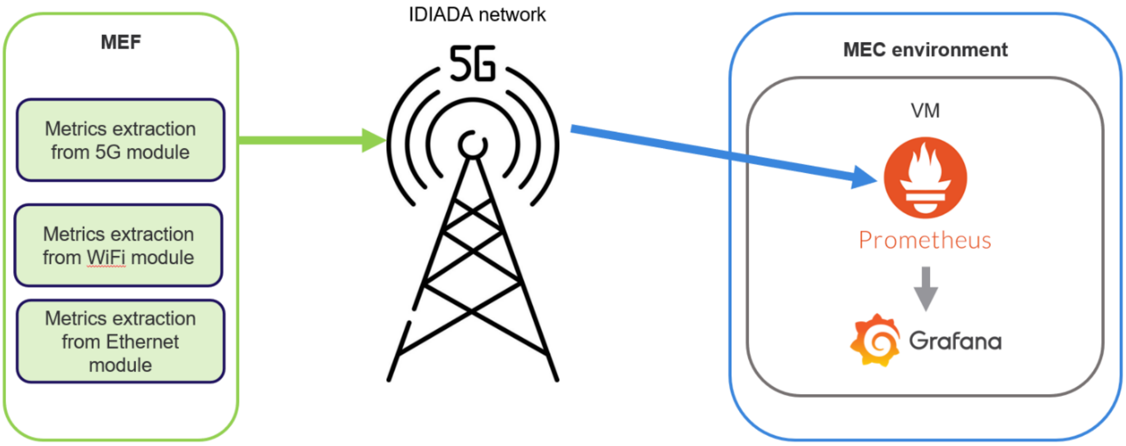


Figure 25: VISTA metrics collection methodology.

Figure 26 shows how metrics that are listed in tables 7 are reported in the Grafana dashboard.

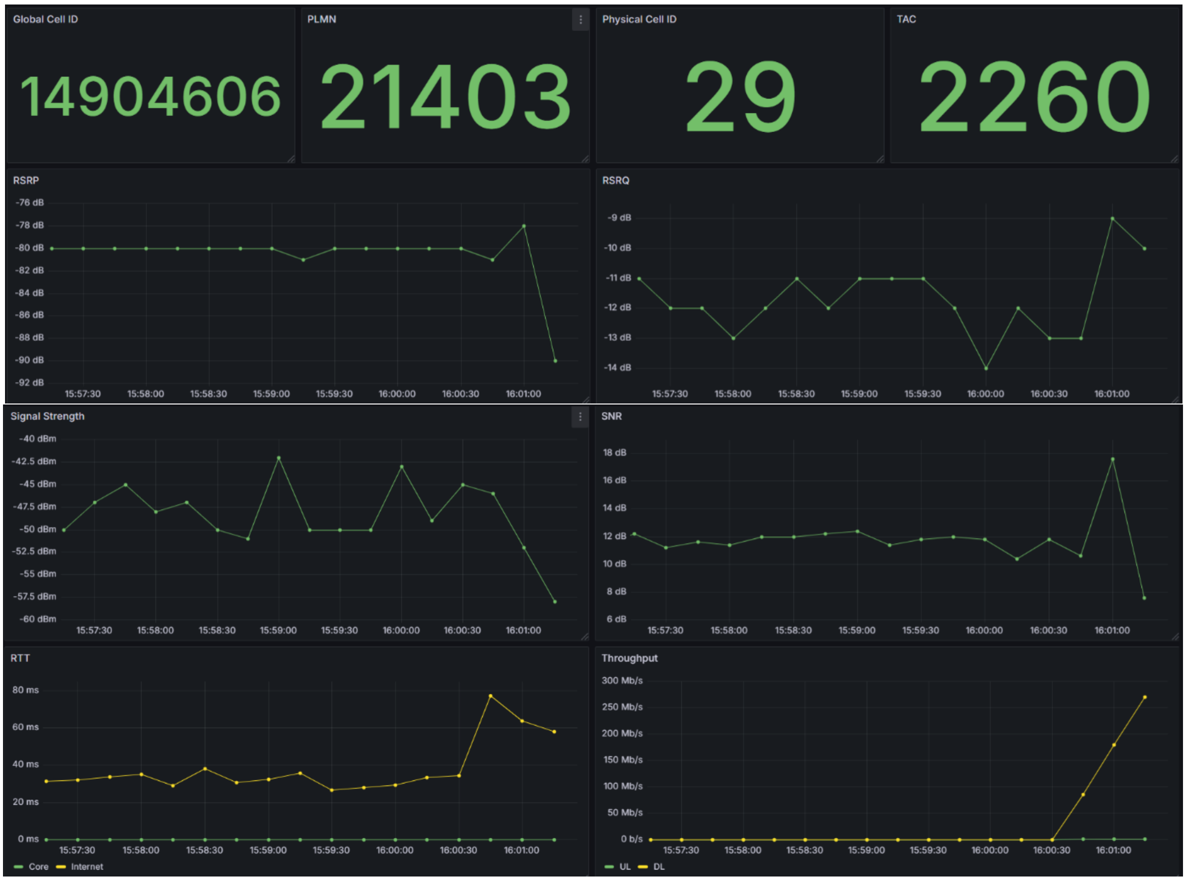


Figure 26: Grafana dashboard to report metrics that are collected by VISTA.

Colorful objects in Figure 27 represent elements that are developed as elements of VISTA tool. Container 1 and Container 2 are two agents for collecting defined metrics from the CPE (Metrics in Table 7). Container1 streams raw data to the Virtual Machine (VM) inside the MEC while container 2 stores them locally. Prometheus and Grafana are placed in the MEC environment to show collected metrics. The DMF evaluates metrics and decides whether the connection quality is good enough to keep sending raw data to Prometheus or should be switched to store them locally. Since the CPE has limited resources, it is advantageous to offload computationally demanding processes to the MEC. However, data collection at the MEC requires reliable network connectivity. Therefore, the optimal solution is a hybrid approach that dynamically switches between local and MEC processing based on connectivity quality. To enable this, the use case owner can define a threshold, derived from one or more network connectivity metrics, to determine when to switch between the two modes. Once a change is triggered, the DMF issues a command to the AEF, which then orchestrates the required actions on the CPE side (e.g., enabling or disabling container 1 and container 2). It is worth mentioning that the DMF considers buffer time after reaching the threshold to trigger the orchestration to avoid ping-pong changes in network orchestration.

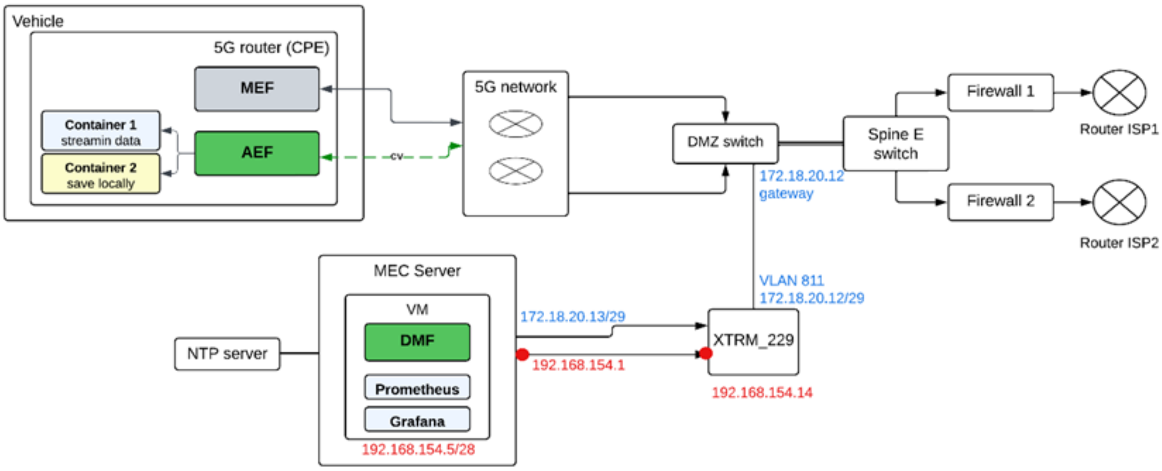


Figure 27: VISTA architecture and connectivity diagram within IDIADA network [3].

In the tested scenario, we used SNR to set the threshold to trigger the orchestrator. The dashed red line in the SNR diagram of Figure 25 illustrates that the threshold is set to 9 dB. Considering the 9 dB for SNR is just an example to validate the orchestrator functionality. This value can be customized based on service requirements. Whenever the SNR value is above 9 dB, the AEF runs the container 1 in the CPE based on DMF commands to send all raw data to the VM in the MEC and report them in the Grafana. Once the SNR goes below 9 dB and stays for 30 seconds, DMF concludes that the connection quality is not good anymore and triggers the AEF to switch from container 1 to container 2 that will start storing raw data locally in the CPE. The situation returns to normal once the field device gets good SNR for more than 30 seconds.

2.4.4 Description of the testcases

The use case was tested in an urban environment (i.e., IDIADA proving grounds) where a vehicle was moving around while the CPEs were collecting metrics (See Figure 28). The same testing area was used for UC 5. We performed more than 10 rounds to experience various spots.

Figure 28: UC4 and UC 5 testing area.

The CPE experienced different qualities connectivity conditions. The bottom part of Figure 29 shows the SNR that is reported by the UE during 75 minutes of the measurement campaign, where the measured SNR ranged between –6 dB and 23.4 dB. As mentioned before, we set the DMF threshold to 9 dB, which is highlighted with the dash red line in this part of the figure. The upper part of the figure depicts the visualized RSRQ on the Grafana dashboard hosted in the MEC. Once the SNR is higher than 9 dB, container 1 is activated in the CPE and the RSRQ are sent to the MEC and shown in the Grafana. In the case when SNR drops below 9 dB, container2 is activated in the CPE and data are stored locally. This is why the Grafana shows nothing in the corresponding time periods. Furthermore, the red boxes in Figure 29 shows the time periods where even though the SNR exceeded the 9 dB, the DMF was not triggered since the duration of exceeding has been short.

Figure 29: Time periods when the orchestrator is not triggers although the SNR is below 9 dB.

2.5 Use case 5: Remote power consumption monitoring tool

2.5.1 Use case description

As it is introduced in section 3.2.2 of D4.3, the remote power consumption monitoring tool relies on the VILLAS framework solution [5]. In a real scenario, this tool can be used to monitor the energy consumption of certain functionalities in (autonomous) electric vehicles. In this use case and since the 5G module plays an essential role in autonomous vehicles, we decided to monitor the power consumption of the CPE with embedded 5G module that delivers 5G connectivity to devices and functions inside the vehicle. This experience can be expanded to all other electric devices inside vehicles.

The VILLAS solution relies on VILLASnodes, which are developed in containers. The first container that runs in the field device (the device that is operating in the vehicle) hosts VILLASnode1 that receives analog energy KPIs as input and timestamps them. There is also VILLASnode2, which is within another container, that hosts the Dynamic Phasor Conversion (DFT) algorithm to perform phasor estimation. The DFT converts thousands of samples that are received as raw data from the VILLASnode1 into a few ones to be further reported in Grafana through Prometheus server.

There is a high data rate between VILLASnode1 and VILLASnode2. Furthermore, VILLASnode2 consumes very much computational resources for DFT to perform the estimation job and therefore, it consumes more power. Hence, it is beneficial in the case of electric vehicle if the VILLASnode2, Prometheus and Grafana dashboard are placed in the MEC or in the cloud. However, the network conditions might fluctuate and the data flow between VILLASnode1 and VILLASnode2 can be interrupted. Therefore, as in use case 4, we will also use the dynamic orchestrator to activate VILLASnode2 either in the field device or the MEC depending on network conditions.

2.5.2 UC5 components

Figure 30 depicts the architecture and elements that are developed for the remote power consumption use case. VISTA is the tool that is developed to use case 4 in this document to collect user experienced metrics. DMF has the same functionality as it is introduced in section 2.4.3 to decide whenever the orchestrator should be triggered. The AEF in this architecture is an adapted and customized version that is compatible with UC5. Container 1 hosts VILLASNode1, while container 3 hosts VILLASNode2. Container 2 performs the role of proxy and always runs in the field device that is mounted in the vehicle and shown in Figure 30. VISTA is a tool that is developed in use case 4 and uses the MEF to collect user experienced metrics.

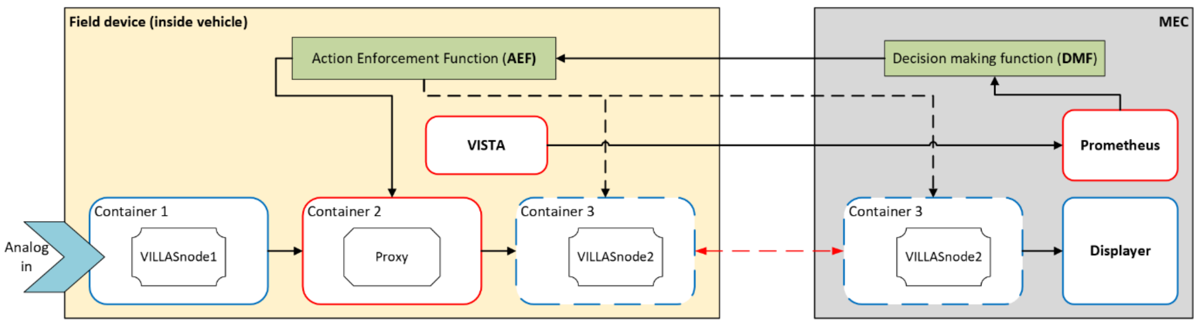


Figure 30: Remote power consumption monitoring tool architecture.

The innovation in this use case is that container 3, which hosts VILLASNode2, migrates from the vehicle to the MEC to adapt to network conditions, which are assessed and evaluated using VISTA.

The default mode is to run container 3 on the MEC to exploit the high computational resources, as well as save power in the vehicle, which is essential in the electric vehicles. However, in the case of low connectivity quality, there will be no way but to run container 3 in the field device (i.e. vehicle) and only send estimated results. In this case, the measurement results will be reliably collected and shown in the MEC but at the price of increased power consumption in the vehicle.

Figure 31 shows the installation that is developed in the testing vehicle. The setup comprises several key components. The first is the field device, which hosts all core elements of the VILLAS framework. Adjacent to it is the analog-to-digital converter, which processes analog signals and feeds the resulting digital data into the field device. Positioned in the top-left corner of Figure 31 is a black GPS unit that provides precise geolocation data for the vehicle—an essential input for the proper functioning of VILLAS nodes. Finally, in the top-right corner, is the Customer Premises Equipment (CPE), the device whose power consumption has been continuously monitored throughout the testing phase.

As the clarification, this testbed (and corresponding photo) is also used for relevant experiences that are reported in D3.6.

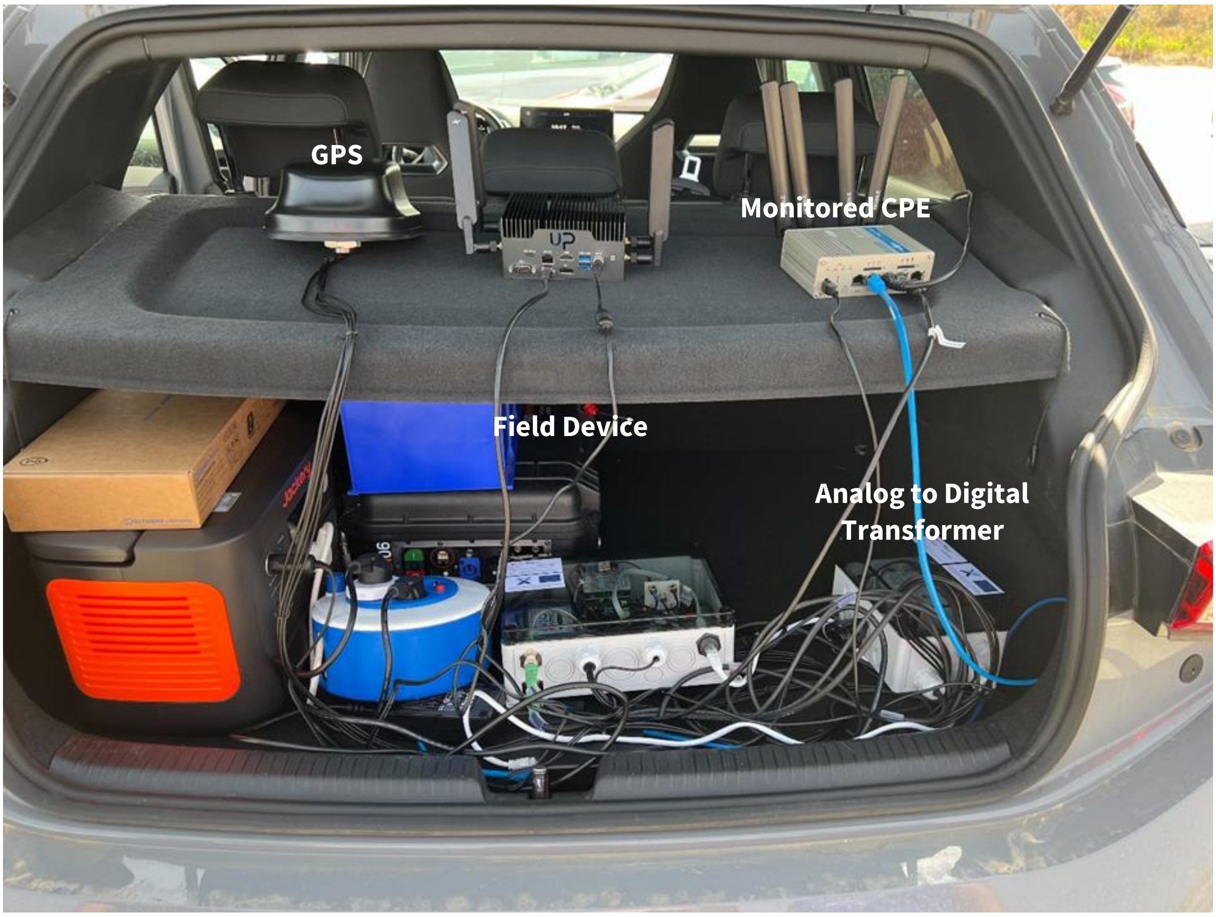


Figure 31: Vehicle installation to perform PoC of UC2.

2.5.3 Measurement methodology and service KPIs

For the validation and performing measurements, we drove 75 minutes across urban environment inside and around IDIADA test road. To experience very low network quality, we moved out of IDIADA and stayed very far from the base station. The experienced SNR ranged between –6 dB and 23.4 dB.

Figure 32 shows the Grafana dashboard that exposes estimated metrics that are generated by the DFT algorithm in the VILLASNode2. These KPIs are voltage, current, and power consumed by a standard vehicle device (Th CPE in this test). In the button left, the figure depicts the connection SNR during the time slot when the vehicle was moving across the network. As marked with the dashed red line, we considered SNR = 5 dB as the threshold to trigger orchestrator to migrate container3 from vehicle to the MEC server and vice versa. Like the previous use case, we selected 5 dB as an example to validate the orchestrator functionality, and it can be further customized with use case owner. If the SNR stays for more than 30 seconds below 5 dB, the DMF activates container 3 in the CPE. In the case of having SNR higher than 5 dB, container 3 will run in the MEC server. The figure in in the left side of the middle row shows the data throughput send from the CPE to the MEC. The three other figures show the reported values for the energy consumption of a standard vehicle device that is installed in the vehicle.

Figure 32: Remote power consumption monitoring tool reporting dashboard.

2.5.4 Description of the testcases

The first observation from the throughput diagram in Figure 32 is that the throughput value is close to zero when the SNR is below 5, and the uplink value jumps to close to 4Mbps when the SNR in higher than 5. The reason is that in the primer case, container 3 runs in the field device and VILLASNode2 only sends very short messages to 5G network. While in the latter case, container 3 runs in the MEC and raw data from container1 will be sent to container 3 through the 5G network.

Besides, power consumption diagrams, the power, voltage and current staid uniform and reported without any interruption. It means that container 3 migration from field device into MEC didn't affect the application layer functionality.

2.6 Use case 6: Remote environment monitoring tool for automated vehicles

2.6.1 Use case description

Remote environment monitoring tool for automated vehicles is designed to predict remote driver video Quality of Experience (QoE) by monitoring network KPIs in ToD scenarios. A fast and reliable mapping between these KPIs and the QoE is crucial for safe ToD. The developed tool will play the role of a watchdog that notifies the remote driver about possible loss of video quality so he can make proactive decisions. This use case is based on the Impact-XG project, which is one of TARGET-X FSTP projects. Figure 33 shows the elements of this service.

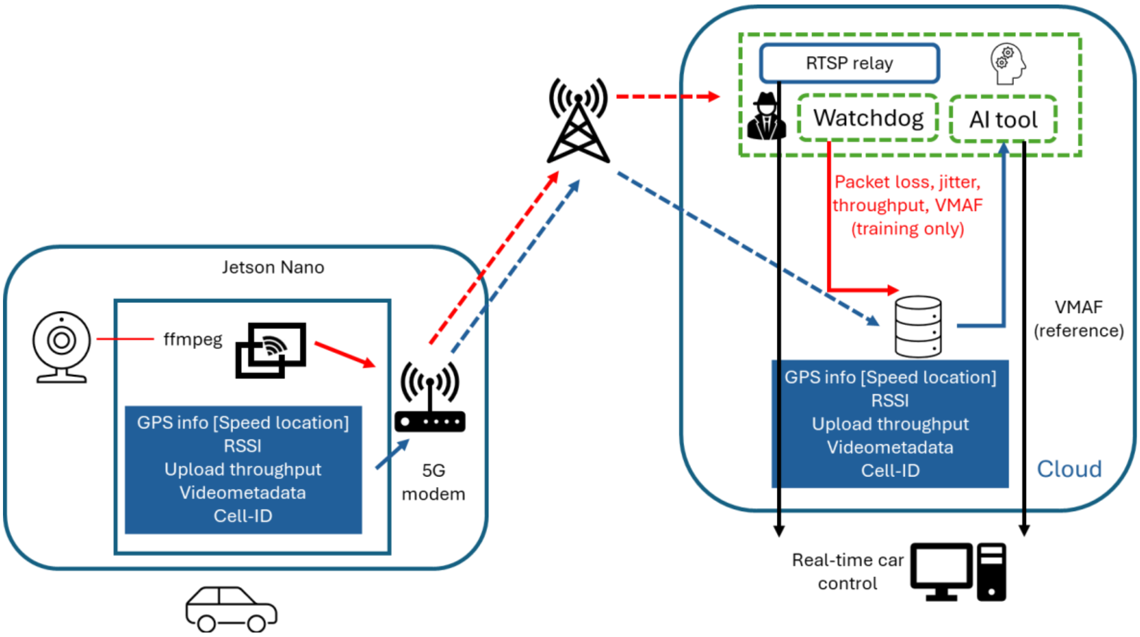


Figure 33: Impact-xG project action flow.

The left box in Figure 33 represents modules in the vehicle and the right box includes modules in the MEC. In the vehicle, cameras' video streams are encoded by Jetson Nano and sent through 5G network toward the MEC. The Real-Time Streaming Protocol (RTSP) relays the video to the remote car control. The watchdog box in the MEC continuously collects service KPIs and assess them. The assessment is performed using the embedded AI tool. The impact-xG modules are integrated with the dynamic service orchestrator to maintain the service availability even when network conditions are not very suitable to send the video with high quality (See D4.3 for more details).

The deployment and validation of the use case functionality is offered by the use case owner [6] and is out of scope of this document. What is relevant to this document is to assess the performance of the integrated dynamic service orchestrator in the use case.

2.6.2 UC6 components

Figure 34 depicts the integrated architecture of the impact-xG modules and the dynamic service orchestrator mechanism of the TARGET-X testbed. The MEF, in this case, benefits from the VISTA tool to expose network KPIs. The DMF is used to trigger the orchestrator based on the preset threshold on SNR value. As it is explained in D4.3, the metric to be used to trigger the orchestrator and the threshold value are configurable. The AEF is an adapted function to collaborate with UC6 APIs and functions.

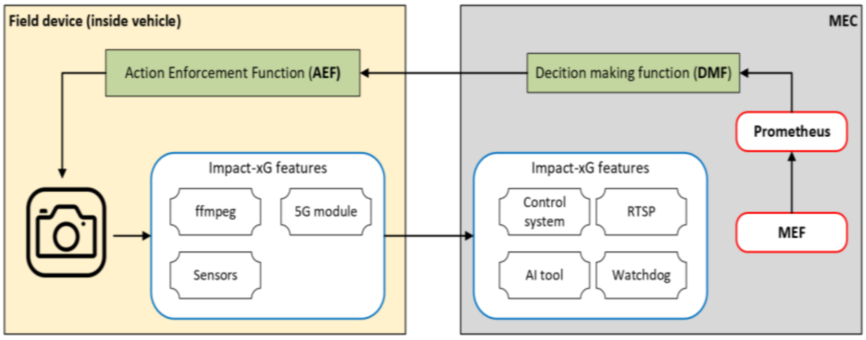


Figure 34: Remote environment monitoring tool for automated vehicles architecture.

Figure 35 Shows the remote environment monitoring dashboard that has been used to validate the dynamic orchestrator functionality. The left side window streams video from surrounding while driving and the plots on the right show connectivity KPIs (i.e., throughput, number of packet lost, packet loss rate, jitter, vehicle speed) and QoE score obtained using perceptual metrics, e.g., Video Multi-Method Assessment Fusion (VMAF) [7].

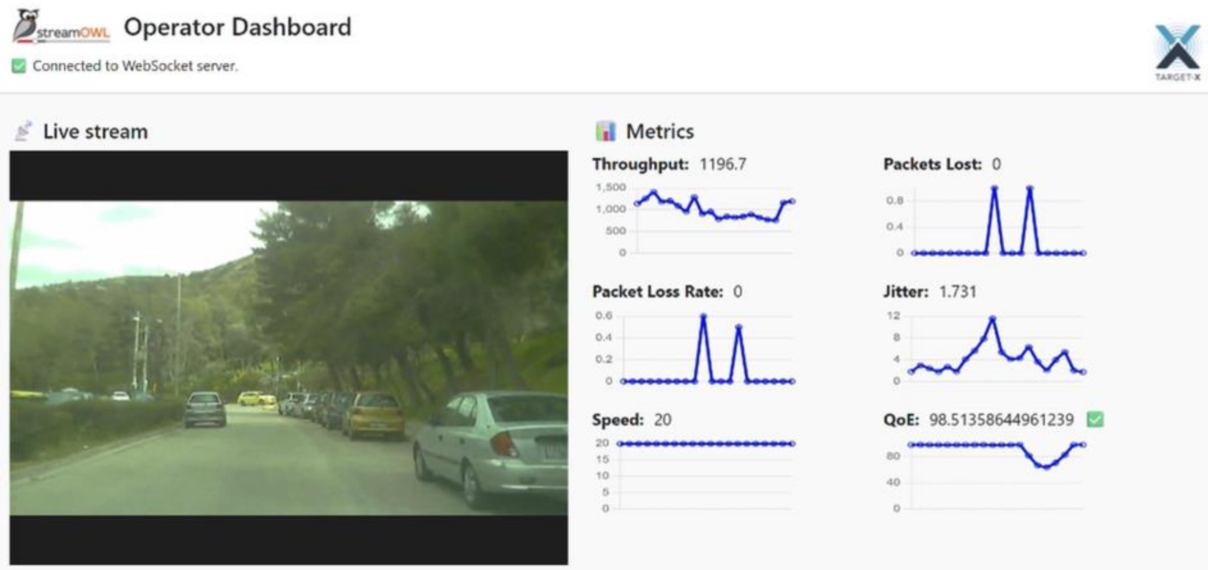


Figure 35: Remote environment monitoring dashboard (Throughput in Kbps, Jitter in ms, Speed in Kmph).

2.6.3 Measurement methodology and service KPIs

The orchestrator in this use case relies on network KPIs that are extracted from VISTA tool. Like UC5, these KPIs are stored in the Prometheus in the MEC server. Service level metrics that are shown in figure 29 are collected and reported from the application and are out of scope of this document.

What is essential for this use case is to be adapted by orchestrator by decreasing the throughput. Both SNR and throughput metrics that are extracted from VISTA are reported in Section 3.6.

2.6.4 Description of the testcases

Like previous dynamic orchestrator use cases, we drove around urban environment in the IDIADA testroad to experience various network qualities. We considered good quality where the SNR is higher than 9 dB, and the situation with poor connection quality, where the SNR is lower than 9 dB.

The threshold value is configurable, and we set it to 9 dB just to have a solid configuration to perform the PoC.

Since streaming video was not allowed in the test road, we used a recorded video to feed the next functions. During the travel, whenever the SNR is higher than 9 dB, cameras' videos will be streamed with the highest quality. However, once the SNR decreases below 9 dB, the DMF will send a command to the AEF to set the limit to 800 Kbps.

Figure 35 shows that in the normal situation, the throughput goes beyond 1 Mbps since there is no limitation. Once the SNR decreases and some packets are dropped, the DMF triggers the AEF to limit streaming quality to 800 Kbps. The orchestrator returns to normal once the SNR backs again higher than 9 dB.

3 Trials results

This section presents and interprets the results obtained in the trials described in Section 2. First, the results of three main use cases are presented in Sections 3.1 (Cooperative perception), in Section 3.2 (Digital twins), and in Section 3.3 (predictive QoS for Tele-operated driving). Then the results of the three use cases related to the cloud continuum paradigm are presented in Section 3.4 (VISTA), in Section 3.5 (Remote power consumption monitoring tool), and in Section 3.6 (Remote environment monitoring tool for automated vehicles).

3.1 Use case 1: Cooperative perception

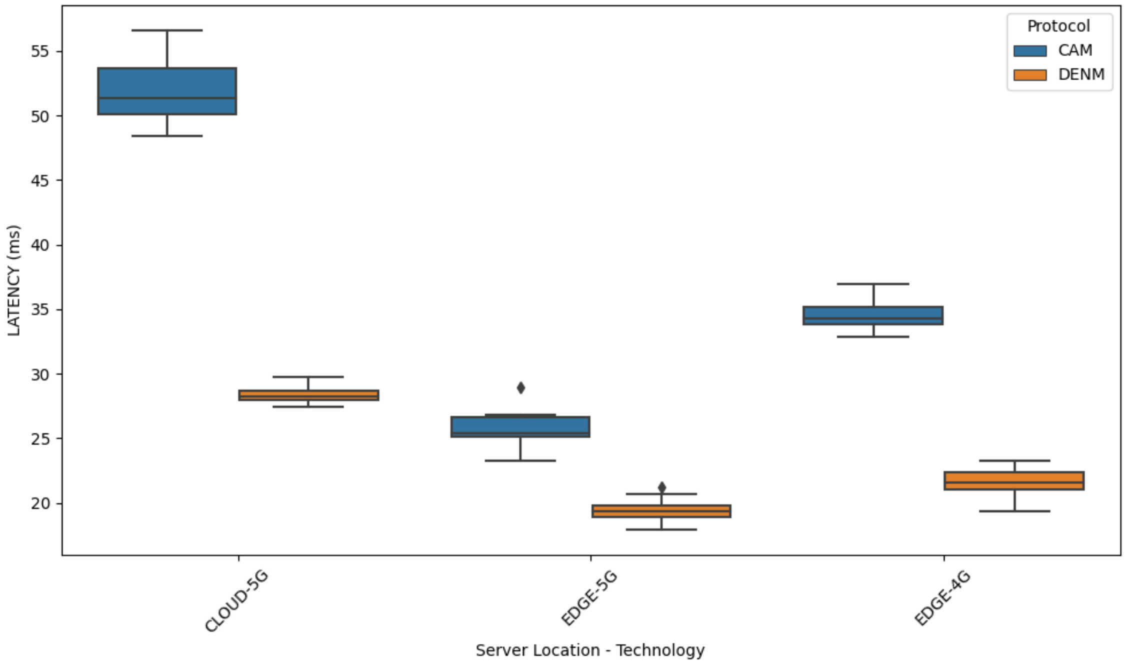
This section presents the performance evaluation results of UC1 in terms of latency and reliability for CAM, DENM, and CPM messages collected when driving the connected vehicle in the Urban area track as described in Section 2.1.4.

3.1.1 Latency

This section presents the results of the service-level latency obtained in tests held in the two scenarios of use case 1. In Figure 36, the boxplots[1] of the CAM and DENM in scenario 1 of UC1 are depicted. As mentioned in Section 2, the CPM is not evaluated in this scenario. The figure shows the results of the latency when the C-ITS is hosted either at the edge directly connected to the UPF or in AWS server in the cloud (Malaga). Also, it shows performance difference when 4G or 5G is used. In addition, the table shows the average, median, standard deviation, minimum, maximum, and the average number of measurements made in each round out of the 15 rounds.

The first result we can observe in the figure is that the median service one way latency is reduced to the half (from 51.3 ms to 25.8 ms) when using the edge instead of the cloud, which confirms the network round-trip latency results in Deliverable D4.3 [3]. The difference in the latency in D4.3 comes from the fact that in D4.3, the network-level latency was measured, whereas in this document the service-level latency is measured, and it includes application processing time in addition to network latency. It should be noted that this conclusion cannot be generalized as it will depend on the location of the cloud server, which is in this case hosted in Malaga, Spain. Another result is that the DENM message latency is in general much lower than the one of CAM (A reduction in the average between 25% and 45% can be seen in the table). Finally, the average latency when using 5G is 75% (resp. 90%) less than the one in 4G for CAM messages (resp. DENM). This difference is because DENM is transmitted in downlink while CAM is transmitted in uplink. In fact, the uplink has less resources due to the regulated frame structure with a downlink to uplink time slot ratio of up to 3:1 is normally used. In addition, random-access procedure is required to send the messages, which might add some delay.

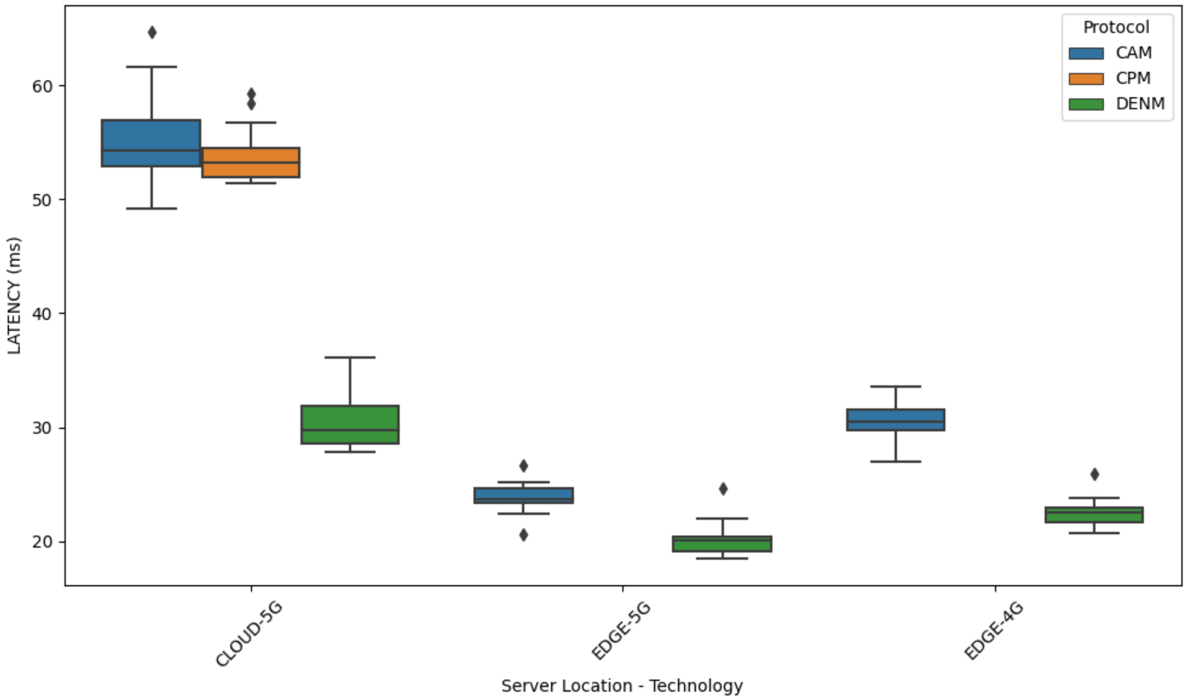
[1] The box in the boxplot plot represents the interquartile range (IQR): The bottom edge represents the first quartile (Q1, 25th percentile), the top edge represents the third quartile (Q3, 75th percentile), and the horizontal line inside the box represents the median. The whiskers extend from the box to show the range of the data excluding outliers: Lower whisker is Q1 – 1.5×IQR and upper whisker is Lower whisker Q3 + 1.5×IQR). points plotted as circles (or other markers) beyond the whiskers. The outliers are values that lie outside the range



	COUNT	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
CAM CLOUD – 5G	512	51.9	51.3	2.3	55.8	48.4	56.6
CAM EDGE – 5G	573	25.8	25.8	1.3	27.7	23.3	28.9
CAM EDGE – 4G	599	34.5	34.3	1.0	36.0	32.9	37.0
DENM CLOUD – 5G	120	28.4	28.2	0.7	29.7	27.5	29.7
DENM EDGE – 5G	90	19.4	19.4	0.8	20.9	17.9	21.2
DENM EDGE – 4G	92	21.6	21.6	0.8	22.9	20.5	23.2

Figure 36: Service-level latency (ms) for CAM and DENM messages in Scenario 1 of UC 1.

Figure 37 shows the results of scenario 2. As mentioned in Section 2, the CPM is only evaluated with 5G and using the C-ITS in the cloud. The results confirm the conclusions in scenario 1 in terms of better performance in 5G than 4G (decrease between 10% and 22%), and in edge compared to cloud (between 33% and 56%). Also, the latency of DENM is lower than in CAM. In addition, the CPM behavior is very similar to the CAM behavior as both of them are transmitted on the uplink.



	COUNT	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
CAM CLOUD – 5G	364	54.7	54.0	2.8	58.8	50.1	61.1
CAM EDGE – 5G	505	23.8	23.7	1.0	25.8	22.5	26.7
CAM EDGE – 4G	590	30.5	30.4	1.6	32.6	27.0	33.6
DENM CLOUD – 5G	113	30.4	30.2	1.1	32.8	28.9	33.0
DENM EDGE – 5G	108	20.2	20.1	1.6	23.1	18.4	24.6
DENM EDGE – 4G	98	22.5	22.5	1.3	24.7	20.7	26.0
CPM CLOUD – 5G	349	53.5	53.0	2.0	56.7	51.5	59.3

Figure 37: Service-level latency (ms) for CAM, DENM, and CPM messages in Scenario 2 of UC 1.

3.1.2 Reliability

This section presents the results of reliability obtained in tests held in the two scenarios of use case 1. In Figure 38, we show the statistics of reliability when the C-ITS is hosted either at the edge directly connected to the UPF or in AWS server in the cloud. Also, it shows performance difference when 4G or 5G is used: the average, median, standard deviation, minimum, maximum, and the average number of measurements made in each round out of the 15 rounds. The boxplots are not shown because the standard deviation is very small in most of the cases and the average reliability is very close to 100%.

	COUNT	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
CAM CLOUD – 5G	512	99.9	100.0	0.1	100.0	99.5	100.0
CAM EDGE – 5G	573	100.0	100.0	0.0	100.0	100.0	100.0
CAM EDGE – 4G	599	99.9	100.0	0.2	100.0	99.2	100.0
DENM CLOUD – 5G	120	100.0	100.0	0.0	100.0	100.0	100.0
DENM EDGE – 5G	90	100.0	100.0	0.0	100.0	100.0	100.0
DENM EDGE – 4G	92	100.0	100.0	0.0	100.0	100.0	100.0

Figure 38 Reliability (%) for CAM and DENM messages in Scenario 1 of UC 1.

In Figure 39, we show the statistics of reliability when the C-ITS is hosted either at the edge directly connected to the UPF or in AWS server in the cloud. Also, it shows performance difference when 4G or 5G is used: the average, median, standard deviation, minimum, maximum, and the average number of measurements made in each round out of the 15 rounds. The boxplots are not shown because the standard deviation is very small in most of the cases and the average reliability is very close to 100.

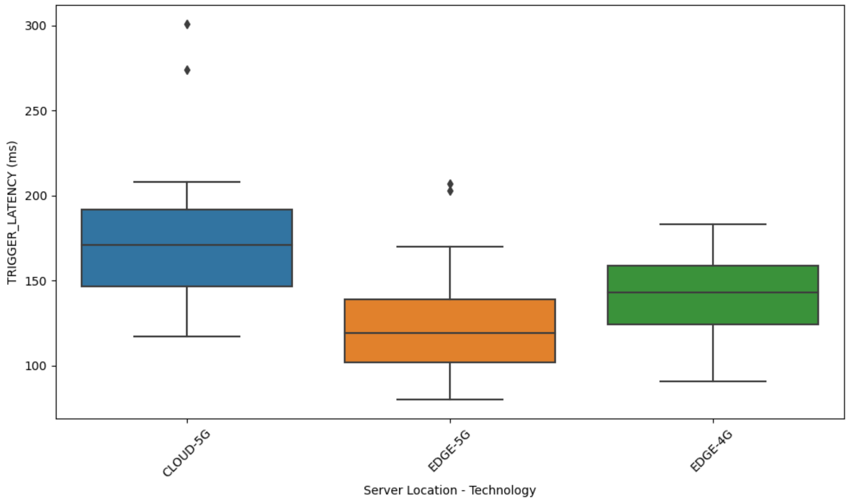
	COUNT	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
CAM CLOUD – 5G	364	100.0	100.0	0.1	100.0	99.7	100.0
CAM EDGE – 5G	505	100.0	100.0	0.1	100.0	99.6	100.0
CAM EDGE – 4G	590	100.0	100.0	0	100.0	100.0	100.0
DENM CLOUD – 5G	113	100.0	100.0	0	100.0	100.0	100.0
DENM EDGE – 5G	108	100.0	100.0	0	100.0	100.0	100.0
DENM EDGE – 4G	98	100.0	100.0	0	100.0	100.0	100.0
CPM CLOUD – 5G	349	99.8	100.0	0.2	100.0	99.2	100.0

Figure 39: Reliability (%) for CAM, DENM, and CPM messages in Scenario 2 of UC 1.

3.1.3 User KPI: DENM TRIGGER LATENCY

This section presents the results of DENM trigger latency in the two scenarios of use case 1. In Figure 40, the boxplots of the DENM trigger latency for scenario 1 are depicted. The figure shows the results of the latency when the C-ITS is hosted either at the edge directly connected to the UPF or in AWS server in the cloud (Malaga). Also, it shows performance difference when 4G or 5G is used. In addition, the table shows the average, median, standard deviation, minimum, maximum, and the average number of measurements made in each round out of the 15 rounds.

The figure shows that the average latency is reduced by 25% (From 170 ms to 127 ms) using the edge instead of the cloud, which confirms the network round-trip latency results in Deliverable D4.3 [3] and the results of the one way latency in Section 3.1.3. Moreover, the average latency when using 5G is 9 % less than the one in 4G (From 140 ms to 128 ms).

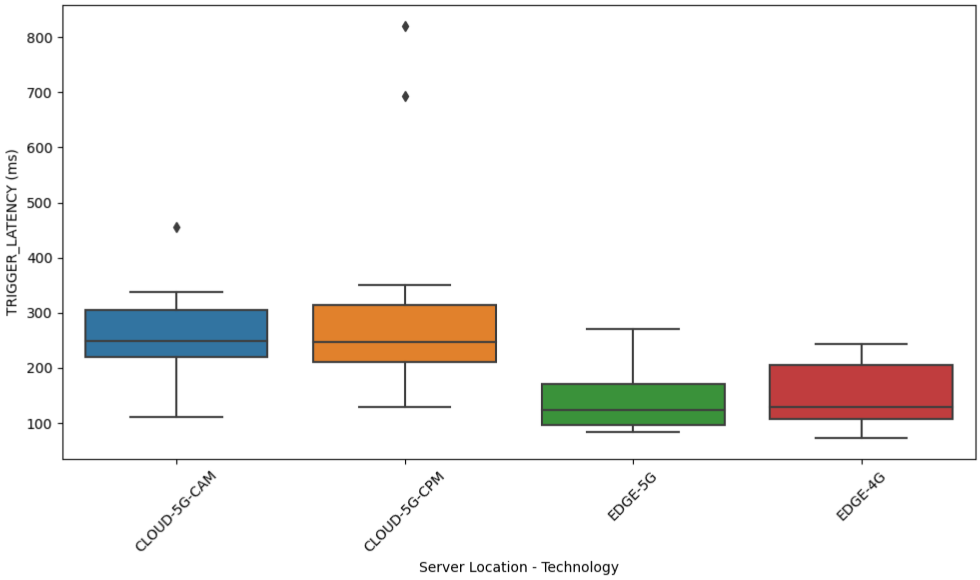


	COUNT	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
CLOUD – 5G	120	170.8	171.0	48.7	282.1	117.0	301.0
EDGE – 5G	90	127.8	119.0	39.6	204.6	80.0	207.0
EDGE – 4G	92	140.3	143.0	27.5	178.2	91.0	183.0

Figure 40: DENM trigger latency (ms) in Scenario 1 of UC 1.

In Figure 41, the boxplots of the DENM trigger latency for scenario 2 are depicted. The figure shows the results of the latency when the C-ITS is hosted either at the edge directly connected to the UPF or in AWS server in the cloud (Malaga). It should be noted that we removed two outliers (above 1000 ms) to make the figure clearer. Also, it shows performance difference when 4G or 5G is used. As a difference from Figure 40, this figure also shows the latency when CPM is used. In addition, the table shows the average, median, standard deviation, minimum, maximum, and the average number of measurements made in each round out of the 15 rounds.

The figure shows that the average latency is reduced by 44% (From 255 ms to 143 ms) using the edge instead of the cloud, which confirms the network round-trip latency results in Deliverable D4.3 [3] and the results of the one way latency in Section 3.1.3. Moreover, the average latency when using 5G is 6 % less than the one in 4G (From 152 ms to 144 ms). An important remark in this case is that the average latency has exceeded the threshold of 200 ms when the C-ITS was hosted in the cloud. This could be due to extra latency between IDIADA and the C-ITS in Malaga or due to some slow processing in the AWS.



	COUNT	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
CLOUD – 5G – CAM	113	255.3	248.0	60.6	325.0	116.0	337.0
CLOUD – 5G – CPM	349	550.8	263.0	735.1	1900.2	130.0	2916.0
EDGE – 5G	108	143.7	124.0	56.4	251.2	84.0	271.0
EDGE – 4G	98	152.3	129.0	47.5	221.0	73.0	233.0

Figure 41: DENM trigger latency (ms) in Scenario 2 of UC 1.

3.1.4 Results summary

These results showed the positive impact of using edge instead of cloud and 5G instead of 4G, in terms of reducing latency, especially for the DENM trigger latency. The average latency when using edge is always in the range specified (40-50 ms) in Deliverable D4.1, whereas it can be higher than the maximum when using the cloud. Although there is up to 10 % more latency than the threshold, the functionality of the cooperative perception was not affected during the demo. However, this shows that the fluctuation in the latency when using the cloud are not controllable and might lead, in some cases, to problems in the performance. For reliability the value is always higher than 99.8%. The average reliability is in all cases above the threshold specified in Deliverable D4.1 (99.8%).

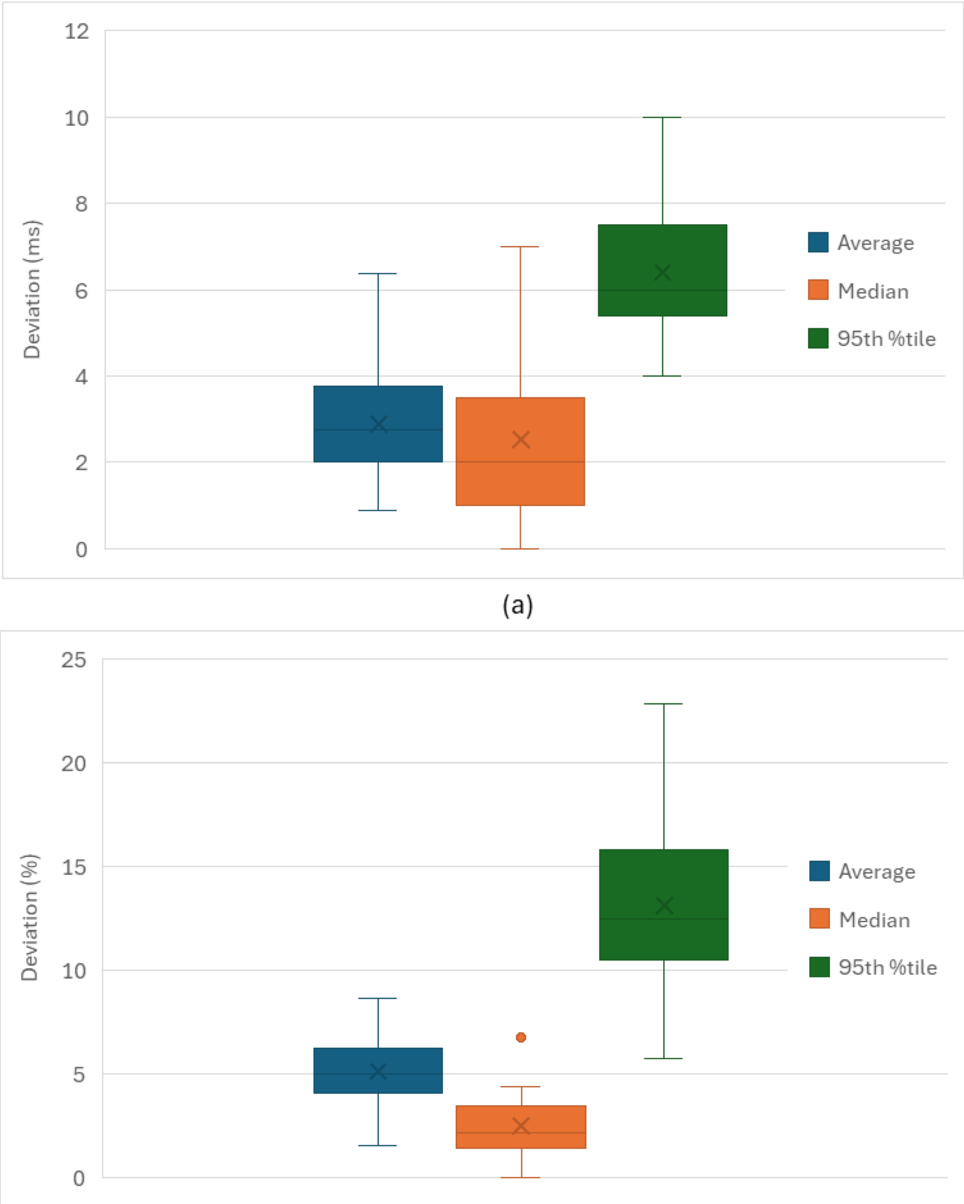
3.2 Use case 2: Digital twins

This section presents the performance evaluation results of UC1 in terms of repeatability as described in Section 2.1.4.

3.2.1 Replicability

This section presents the results of replicability obtained in tests held in use case 2. In Figure 42, the boxplot of the average, median, and 95th percentile boxplots over 17 replays in the digital twin are depicted in absolute value (a) and in percentage from the real time (b). The average and median deviation percentage are always below 10% (corresponding to 7 ms) with averages below 5%

(corresponding to 4 ms). The 95th percentile has an average of around 13% (corresponding to 10 ms) but can reach values up to 23% (corresponding to 14 ms).



(b)

Figure 42: Replay reliability boxplots in use case 2: (a) in absolute value (ms) and (b) in %.

The detailed statistics corresponding to the results are shown in Figure 43.

	AVERAGE	MEDIAN	STD	95TH %TILE
R1	5.3	3.0	11.5	12.5
R2	8.6	6.8	6.6	22.8
R3	4.3	1.6	13.8	8.6
R4	7.8	3.3	19.7	19.3
R5	3.9	1.6	11.0	10.8
R6	4.0	1.3	11.7	10.1
R7	4.8	2.6	10.8	11.8
R8	5.4	2.2	15.0	15.3
R9	4.1	1.2	14.3	10.9
R10	6.4	3.7	14.4	16.6
R11	1.5	0.0	6.6	5.7
R12	6.2	4.3	11.4	14.1
R13	5.0	1.5	15.6	12.9
R14	6.2	3.6	12.4	16.3
R15	4.7	2.5	9.1	12.8
R16	5.0	1.2	20.0	12.4
R17	3.7	1.9	8.6	9.7

Figure 43: Detailed replicability statistics of the 17 replays in use case 2.

3.2.2 Results summary

The results obtained show that the digital twin can replicate cooperative perception scenario with a deviation in the periodicity that is acceptable.

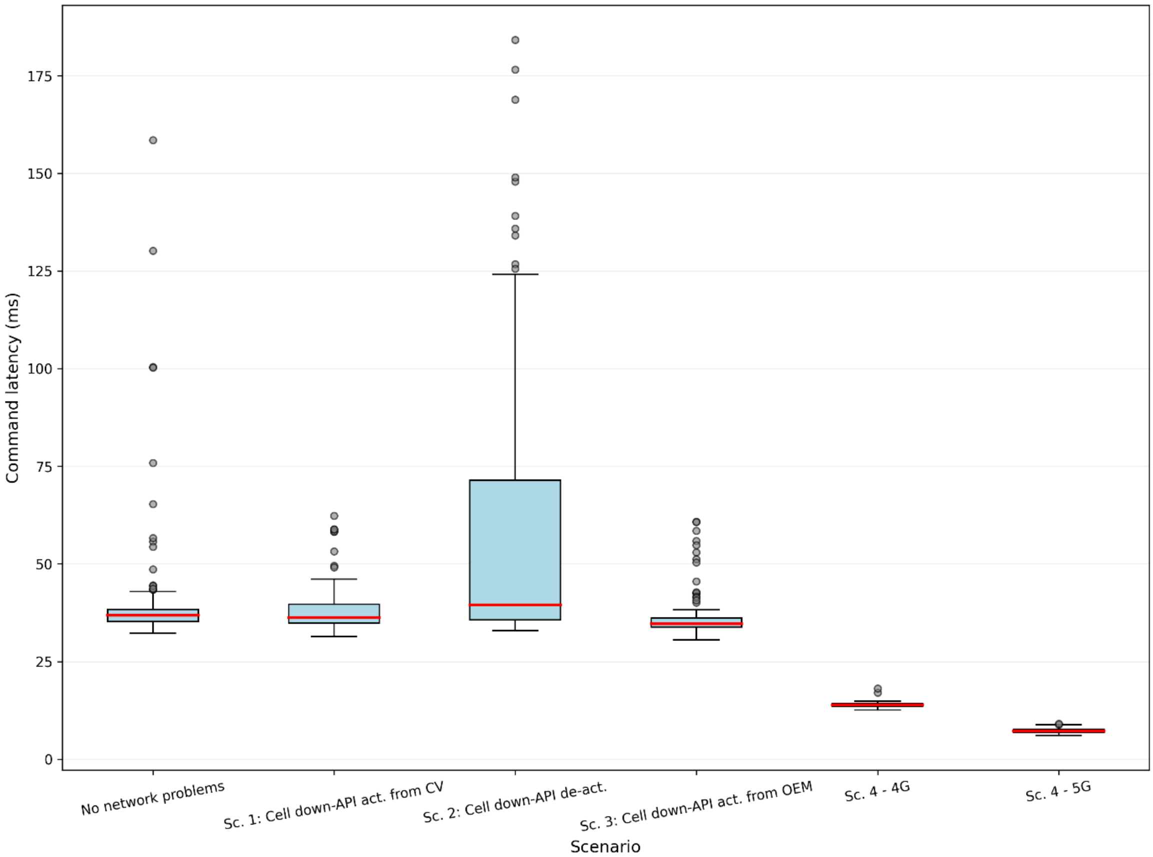
3.3 Use case 3: Predictive QoS for Tele-operated driving

This section presents the performance evaluation results of UC3 in terms of latency, reliability, and throughput for the predictive QoS for tele-operated driving tele when driving the CAV in the Urban area track as described in Section 2.3.4. The scenarios in the figure represent the ones described in Section 2.3.4.

3.3.1 Latency

This section presents the results of latency obtained in tests for all scenarios of UC3. In Figure 44, the boxplots of the downlink command latencies in the six scenarios are depicted (Three outliers of Sc.2 are removed – 224.4 ms, 434.7 ms, and 609.6 ms– to make the figure clearer). In addition to showing enhanced performance when using the predictive QoS, the figure shows difference between 5G and 4G latency. In addition, the table shows the average, median, standard deviation, 95th percentile, minimum, and maximum.

The first result we can observe in the figure is that the significant decrease in average latency when using 5G compared to when using 4G by 47%, which confirms the results obtained in UC 1 about the RTT.



	AVG	MEDIAN	STD	95%-TILE	MIN	MAX
NO NETWORK PROBLEMS	40.5	36.9	16.2	56.3	32.3	158.5
SC. 1: CELL DOWN-API ACTIVATED FROM CV	38.1	36.3	5.5	49.3	31.4	62.4
SC. 2: CELL DOWN-API DE-ACTIVATED	65.4	39.9	69.1	148.5	32.9	609.6

SC. 3: CELL DOWN-API ACTIVATED FROM OEM	36.3	34.6	5.6	50.8	30.6	60.8
SC. 4 - 4G	13.9	14.0	0.7	14.7	12.6	18.1
SC. 4 - 5G	7.3	7.2	0.6	8.3	6.1	9.1

Figure 44: Latency (ms) of the command messages in the downlink for all scenarios of UC3.

The average values in the two scenarios with exposure APIs (Scenario 1: notification via connected vehicle, and Scenario 3: notification via Ericsson Dashboard) are both in the range 20-50 ms (38.1 ms for scenario 1 and 36.3 ms for scenario 3) specified in Deliverable D4.1 [1] as a target. Even the 95th percentile is within this range (49.3 for scenario 1 and 50 ms for scenario 3). This is not the case in scenario 3 where no exposure APIs are implemented (average 65.4 ms and 95th percentile 148.5 ms). Another key finding is the significantly more stable command latency performance when using predictive QoS, in both implementations, compared to Scenario 2, where no predictive mechanism was used during cell deactivation. Specifically, the standard deviation of latency dropped by 92% in both scenarios, demonstrating far less performance fluctuation when proactive notifications are in place.

To better understand this, Figure 47 illustrates the command latency evolution during cell deactivation. In Scenario 2, where the remote driver continued driving into the coverage hole, latency increased drastically (more than 100 ms in average and up to 600 ms in some cases), resulting in loss of control on the vehicle, and an emergency stop by the safety driver. In contrast, in Scenarios 1 and 3, the remote driver received timely notifications via the network exposure API, allowing him to stop the vehicle before entering the coverage hole. As a result, latency remained stable, and the user experience was uninterrupted, enabling a safe, controlled parking maneuver without service degradation.

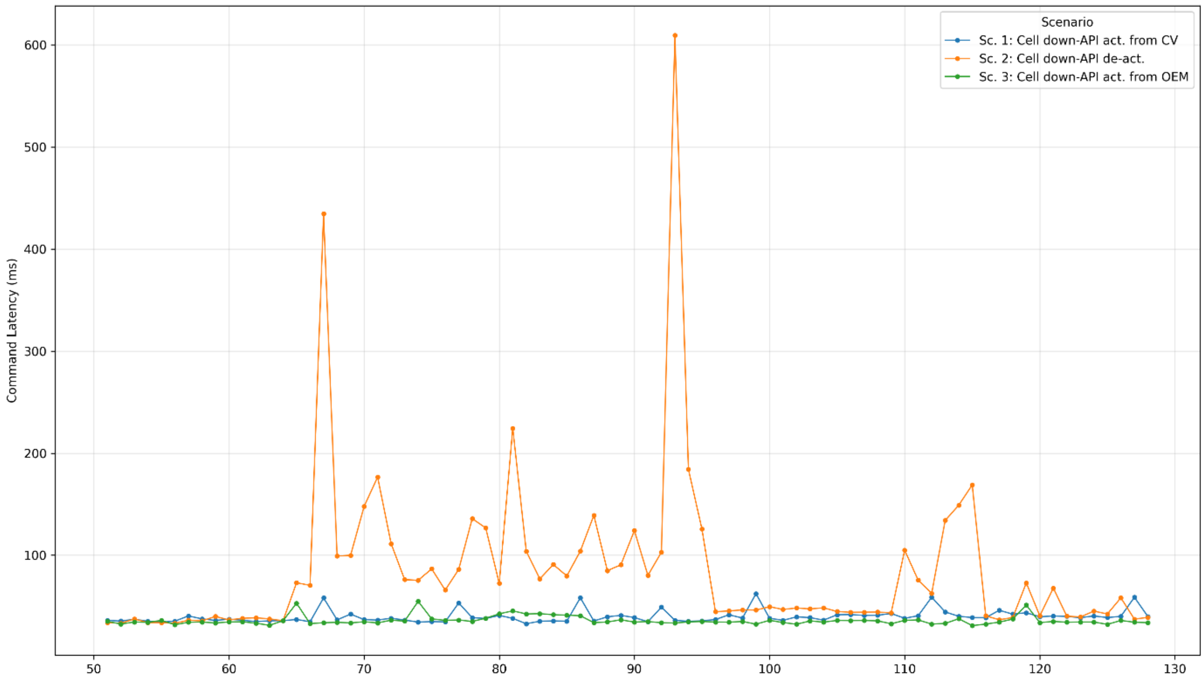


Figure 45: Evolution of the command latency over time in UC3 when the cell is deactivated.

It should be noted that the video uplink average latency was much lower than the threshold of 100 ms in all scenarios: the highest was in scenario 2 with 58.3 ms.

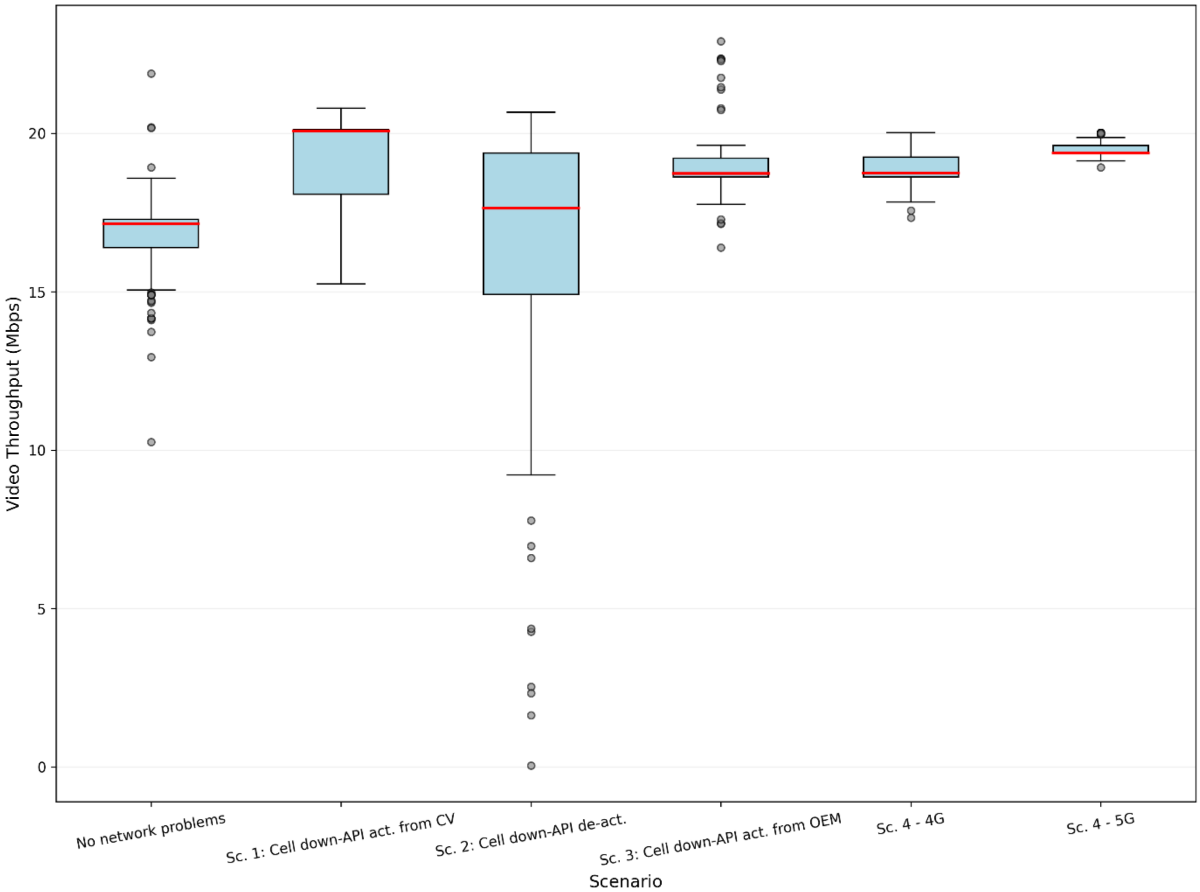
3.3.2 Reliability

The reliability in both links was always higher than 99% and this is because, when the cell was turned off, the remote driver had to stop the vehicle either before reaching the zone without coverage or directly when reaching that zone.

3.3.3 Throughput

This section presents the results of the uplink throughput of the three Cameras used in UC3 and obtained in tests for all scenarios. In Figure 46, the boxplots of the throughputs in the six scenarios are depicted. In addition to showing enhanced performance when using the predictive QoS, the figure shows difference between 5G and 4G throughput. In addition, the table shows the average, median, standard deviation, 5th percentile, minimum, and maximum.

The first result



	AVG	MEDIAN	STD	5%-TILE	MIN	MAX
NO NETWORK PROBLEMS	16.7	17.1	1.4	17.1	10.3	21.9
SC. 1: CELL DOWN-API ACTIVATED FROM CV	19.2	20.1	1.4	20.1	15.3	20.8

SC. 2: CELL DOWN-API DE-ACTIVATED	16.4	17.6	4.3	17.6	0.0	20.7
SC. 3: CELL DOWN-API ACTIVATED FROM OEM	19.0	18.7	1.0	18.7	16.4	22.9
SC. 4 - 4G	18.9	18.7	0.5	18.7	17.3	20.0
SC. 4 - 5G	19.5	19.4	0.2	19.4	18.9	20.0

Figure 46: Uplink video Throughput in Mbps of the three Cameras for all scenarios of UC3.

The average values in the two scenarios with exposure APIs are in all scenarios in the range 10 – 50 Mbps (19.2 Mbps for scenario 1, 16.4 Mbps for scenario 2, and 19 Mbps for scenario 3) specified in Deliverable D4.1 as a target. However, the minimum values in scenarios 1 and 3 are 15.3 Mbps and 16.4 Mbps, whereas it can reach 0 in the case of scenario 2. Another key finding is the significantly more stable video throughput performance when using predictive QoS, in both its implementations, compared to Scenario 2. Specifically, the standard deviation of the throughput dropped by 67% in Scenario 1 and 76% in Scenario 3, demonstrating far less performance fluctuation when proactive notifications are in place.

To better understand this, Figure 47 illustrates the throughput evolution during cell deactivation. At the end of the series where the vehicle approached the low-coverage zone, the throughput dropped to less than 5 Mbps, which is way below the threshold of 10 Mbps. The remote driver did not receive alerts and continued driving into the low coverage area, and throughput collapsed abruptly, resulting in video glitches, eventual loss of feed, and an emergency stop by the safety driver. In contrast, in Scenarios 1 and 3, the remote driver received timely notifications via the network exposure API, allowing them to stop the vehicle before entering the low coverage area. As a result, throughput remained stable, and the user experience was uninterrupted, enabling a safe, controlled parking maneuver without service degradation.

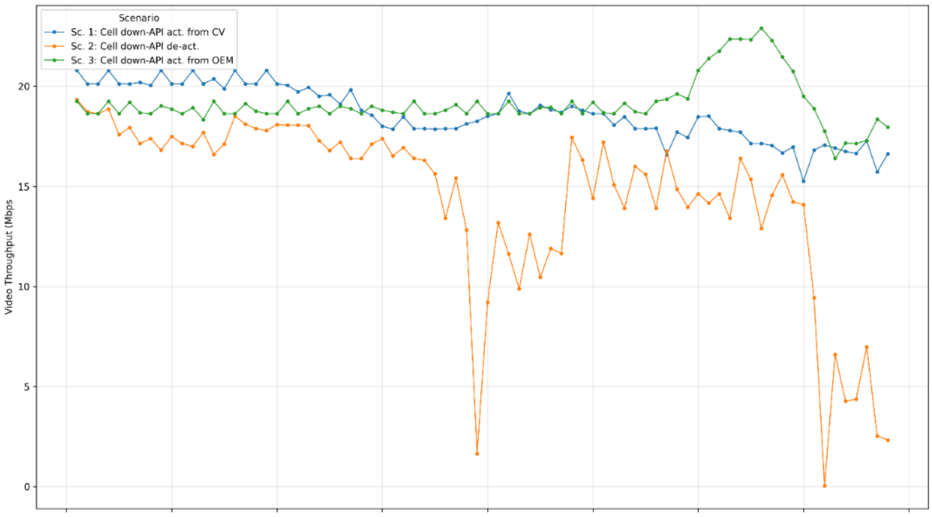


Figure 47: Evolution of the video throughput over time in UC3 when the cell is deactivated.

We also evaluated the network-level throughput for this use case by collecting traces using tcpdump. The results are shown in Figure 48 confirm the results of the service-level throughput. As it can be seen, the throughput in scenario 2 is decreased significantly compared to the other two use cases, where the pQoS was activated. It should be noted that the network throughput is higher than the service-level throughput as it is measured without extracting the headers.

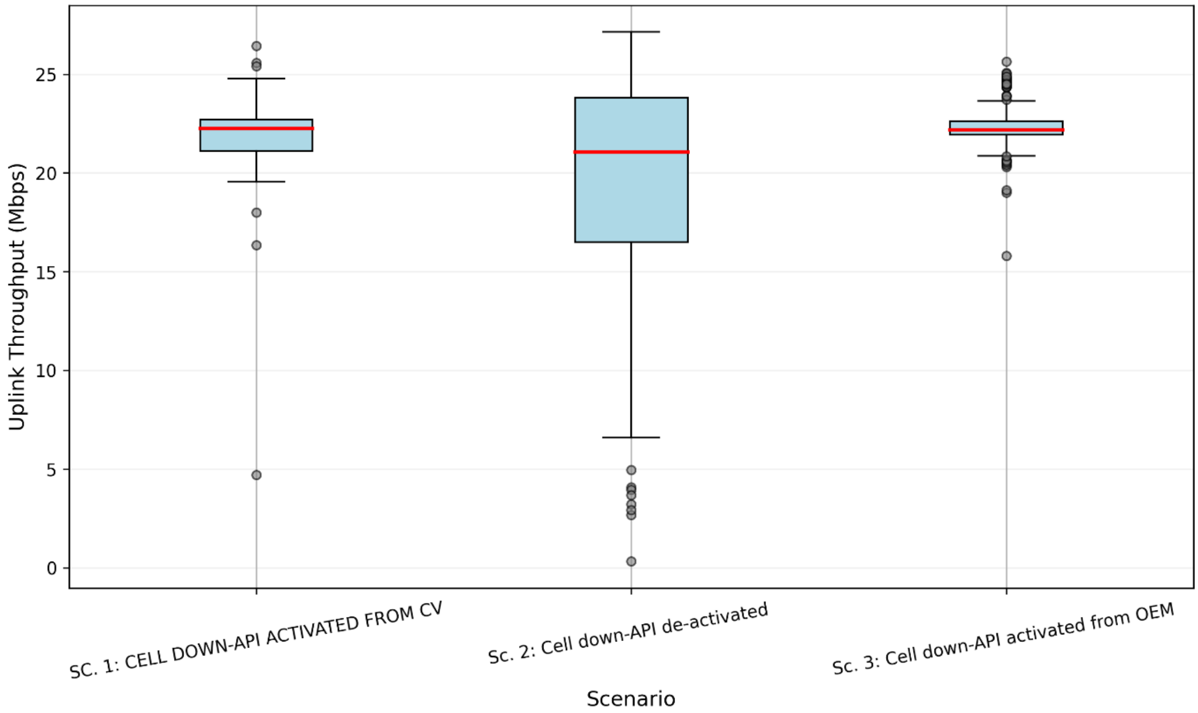


Figure 48: Network-level throughput in UC3.

3.3.4 Results summary

The results in this use case showed that using 5G combined with predictive QoS functionality allows safer driving of ToV. It was shown that using this method, all KPIs required for safe tele-operation can always meet their targets through the interaction with the network and getting notification when the conditions are not suitable for ToD. This highlights how predictive QoS, powered by network exposure, transforms a disruptive failure into a managed, user-transparent event. It should be noted that this use case was also evaluated in a large-scale simulation to test the possibility of re-routing in a dense urban area using real-life network KPIs obtained from a European operator in this city. The results show the stability of the solution and its ability to provide a reliable route for the ToD [8].

3.4 Use case 4: VISTA

This section depicts relevant statistics from KPIs (i.e., SNR, signal levels, RTT) that VISTA tool collected from the UE during the 75 minutes of mission travel in IDIADA test road. In addition, we also analyze the service adaptation delay that measures the time needed by the DMF to switch from one container to another.

3.4.1 SNR

The SNR is considered the metric to trigger the orchestration, and 9 dB is considered as the threshold as explained in Section 2.4.3. The reported diagram in Figure 49 illustrates the cumulative distribution function (cdf) of the SNR. It shows that the SNR was higher than 9 dB in 56% of the test time. During the experience, Container 1 was active and operating for almost 43% of the time while container 2 was operating for 57% of the time. Besides, in 9.5% of the time, the AEF was waiting 30 seconds to switch between containers. It was also recorded 14 events where, even though the SNR had reached the threshold, the DMF did not trigger the AEF because the duration was less than 30 seconds. This situation happened in 8.3% of the events where the threshold was crossed.

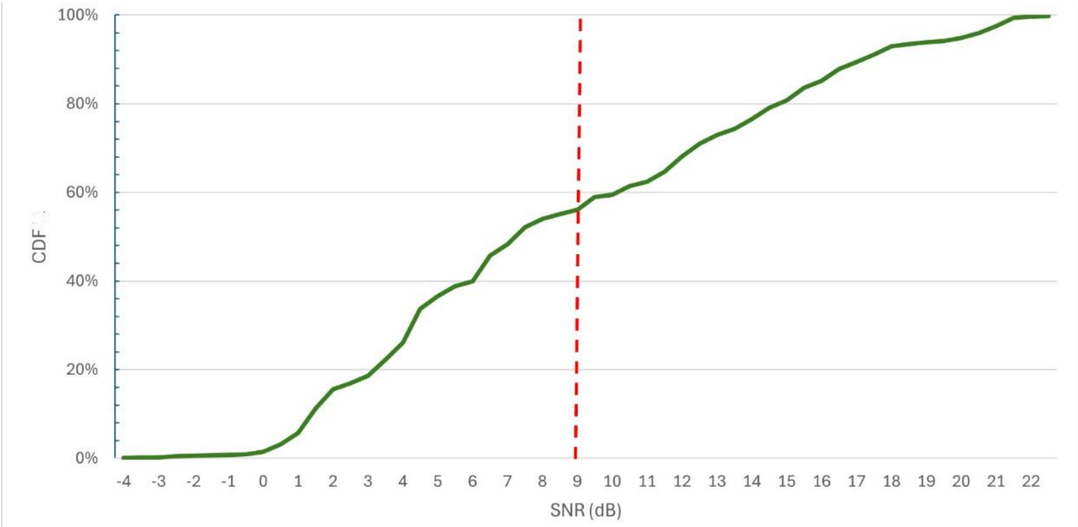


Figure 49: CDF for SNR and marked threshold in UC4.

3.4.2 Signals levels

Signal level metrics including RSRP, RSRQ and RSSI that are collected by VISTA in the CPE while traveling for 75 minutes are reported in Figure 50.

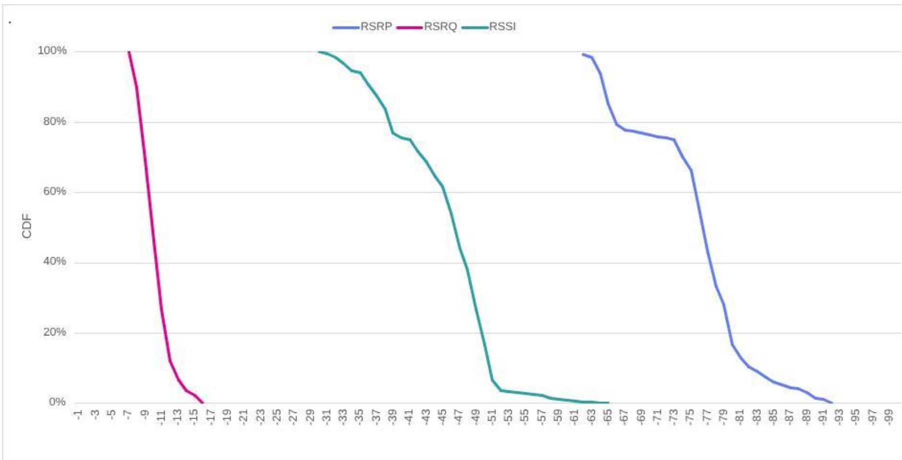


Figure 50: CDF of RSRP, RSRQ and RSSI that are collected with VISTA in UC4.

3.4.3 RTT

RTT to edge and to a cloud destination that are collected with VISTA tool are reported in Figure 51.

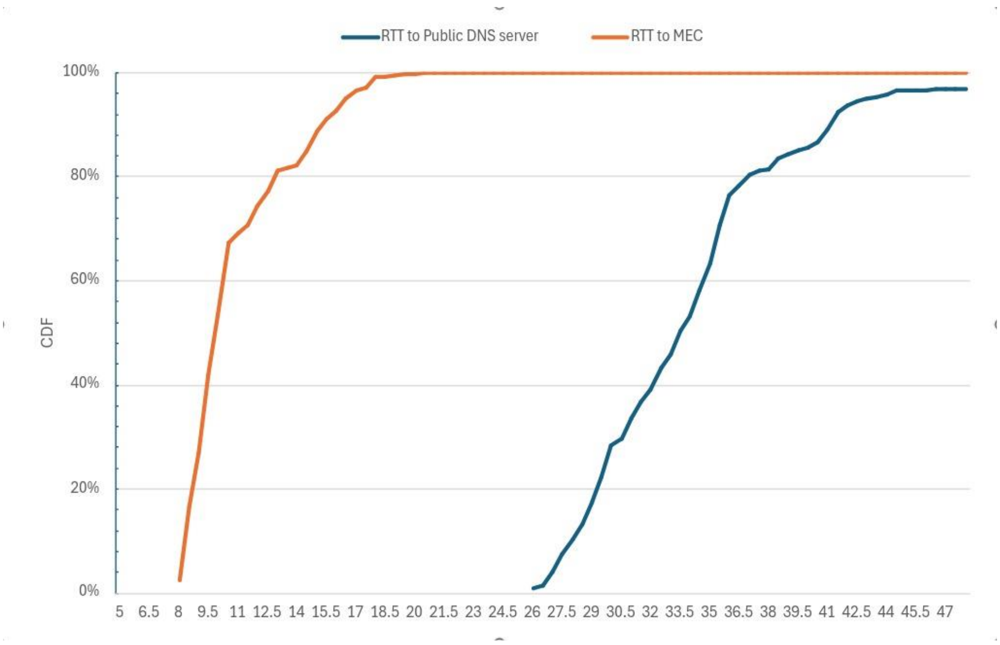


Figure 51: CDF of RTT to a public DNS server and MEC destination that are collected with VISTA in UC4.

3.4.4 Service adaptation delay

As is mentioned in Section 2.4.3, a buffer of 30 seconds is considered from when the threshold is exceeded until when the MDF triggers the orchestration. This duration is configurable and can be changed based on the setup requirements. Figure 52 illustrates the 30 seconds buffer when the SNR increased above 9 dB at 15:56:45 until when metrics started to appear in the Grafana at 15:57:15 (30 seconds later)

During the test, it happened 16 times that DMF triggered the orchestrator and switched from one container to another. Also 14 times the SNR crossed the threshold, but the orchestrator had not been triggered since the duration was less than 30 seconds.

Figure 52: Orchestration execution delay in UC4.

3.4.5 Results summary

Based on the experimental results, VISTA tool can successfully record experienced KPIs from the user point of view. The delay in the orchestration activation to avoid the ping-pong situation is very well validated in Figure 52.

3.5 Use case 5: Remote power consumption monitoring tool

3.5.1 Service interruption (Reliability)

As it is reported in the Figure 32, the orchestrator triggered 15 times during 75 minutes and the container3 migrated from either field device to MEC, or from MEC to the field device. However, Grafana dashboard shows that results are received without interruption. It confirms that the container3 migration from field device to the MEC, and vice versa, has not affected on the application layer functionality.

3.5.2 Throughput

The main idea of UC 5 has been saving network resources when there are not enough available. Hence, Figure 53 shows considerable saving in the bandwidth when the container3 runs in the field device. There is an average of 22.8 Kbps throughput when the container 3 is running in the field device while this average reaches 3.2 Mbps when container3 runs in the MEC. This throughput is when the analogue sample rate is set to 10k. The required throughput will increase if the sample rates increases.

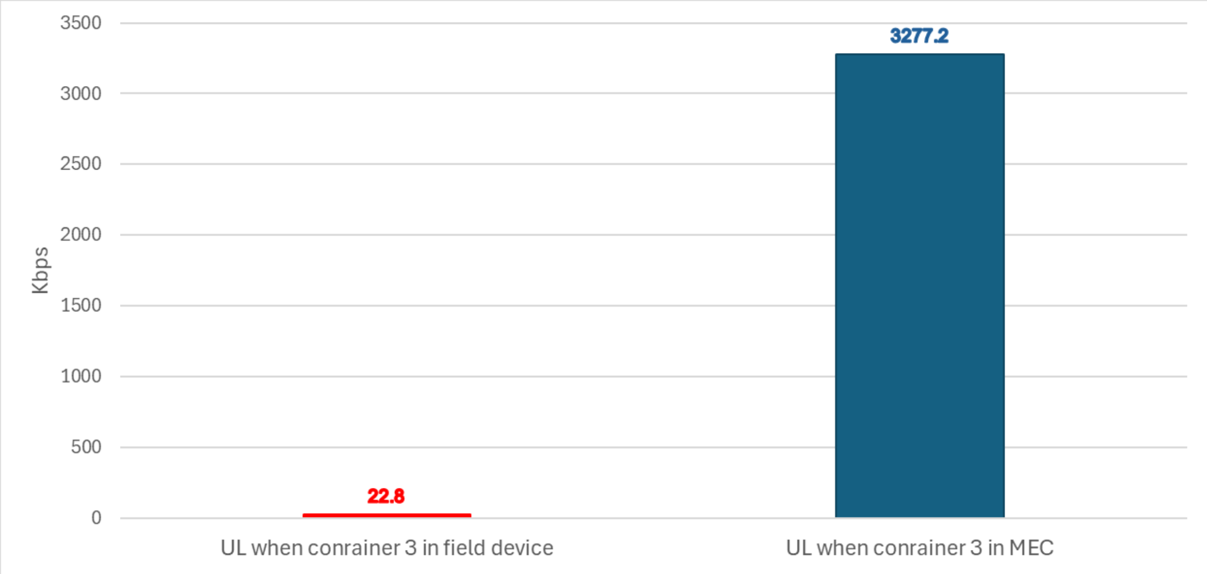


Figure 53: Comparing network resource consumption when container 3 is running in the edge versos running in the field device.

3.5.3 Computational resource

Figure 54 shows the computational resource of the VILLASNode2. When the DFT algorithm is running in the MEC, the CPU consumption in the field device (Vehicle) is negligible. Once the orchestrator triggered to run the DFT algorithm in the field device, the CPU consumption jumps to 21% while the average is 25% and pick CPU consumption is reported 53%.

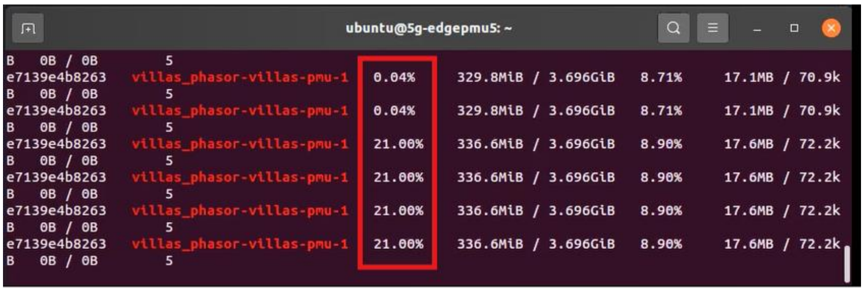


Figure 54: Comparing field device computational resource consumption when container 3 is running on the edge versus running on the field device.

3.5.4 Results summary

Based on the experimental results, the proposed dynamic orchestration solution doesn't interrupt the energy consumption minoring in the displayer. It means that the process for computational resources migration from MEC to the field device is completely transparent for application layer and end user.

It is evaluated that almost 3.2 Mbps in uplink channel will be consumed to relay raw collected data from container1 to container3 in the MEC. It is possible to decrease this throughput to some Kbps by moving container3 to the fired device. This migration happens at the cost of increasing computational resource consumption in the field device. It is also captured that an average of 25% of computational resources of the field device will be dedicated to the DFT algorithm if it is run in the field device. This fact results in increasing energy consumption in the vehicle, which is critical in electric vehicles.

The conclusion is that there should be a tradeoff for network resources consumption and the field device computational resource consumption where we put the orchestrator trigger on it.

3.6 Use case 6: Remote environment monitoring tool for automated vehicles

The main parameter for this use case to validate is how the service adapts the uplink traffic considering connectivity quality. To this end, it is essential to map the SNR value, which is selected to trigger the orchestrator, and the throughput during testing timeline.

Figure 55 shows collected SNR during 60 minutes of test experience. The sample rate is 15 seconds, and the decision time window is 30 seconds (i.e. the time which the SNR should stay beyond the threshold, so the orchestrator be triggered). The threshold has been considered to be 9 dB in this test.

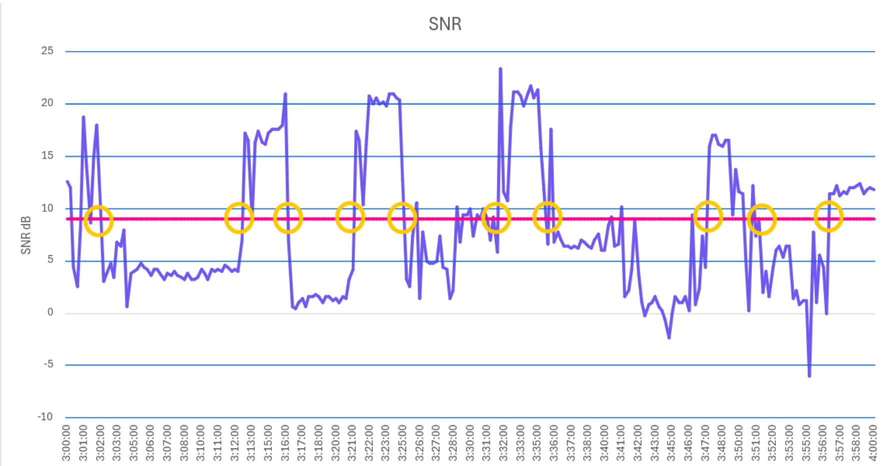


Figure 55: Captured SNR during UC6 validation test.

Orange circles show the time when the orchestrator had been triggered because of reaching to threshold. There are other occasions that even through the SNR had reached the threshold, the orchestrator had not been triggered. That is because the duration when the SNR passed the threshold was less than 30 seconds.

Considering the trigger points, Figure 56 illustrates how the throughput has been affected by scaling down the video streaming based on orchestrator decision.

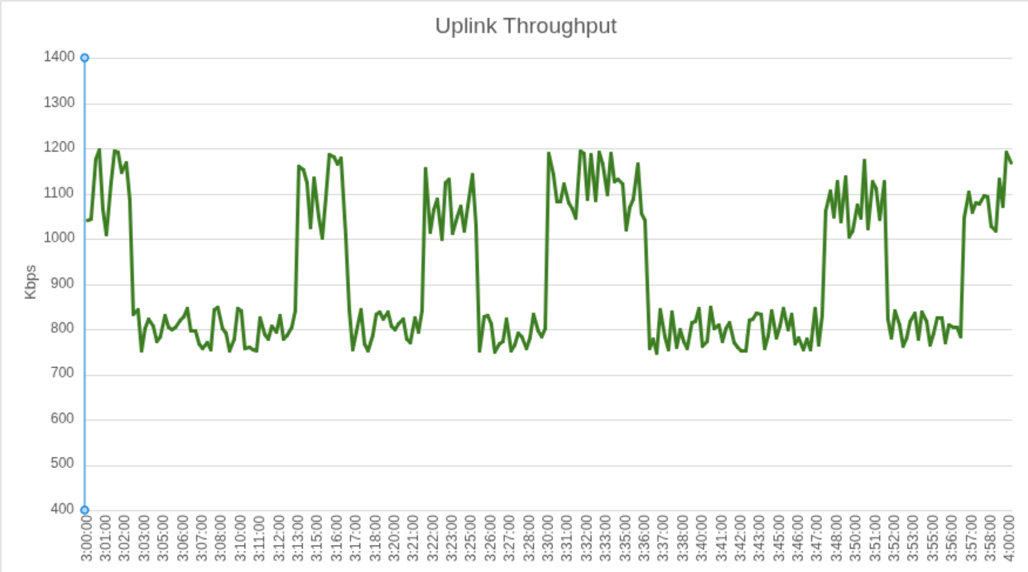


Figure 56: Throughput during UC6 validation and test.

The normal throughput with the default video quality setup goes between 987 Kbps and 1.21 Mbps with the average value of 1.108 Mbps. Once the orchestrator triggers, the vehicle device scales down the video quality, and the throughput decreases to an average of 798.96 Kbps.

3.6.1 Service level delay

Service level delay, which we named it as Service-jitter, in the application layer delay from when a video packet is scheduled to be sent from the vehicle until reception in the server side, placed in the MEC. Figure 57 shows the service-jitter that is reported by the UC application during 60 minutes of testing experience

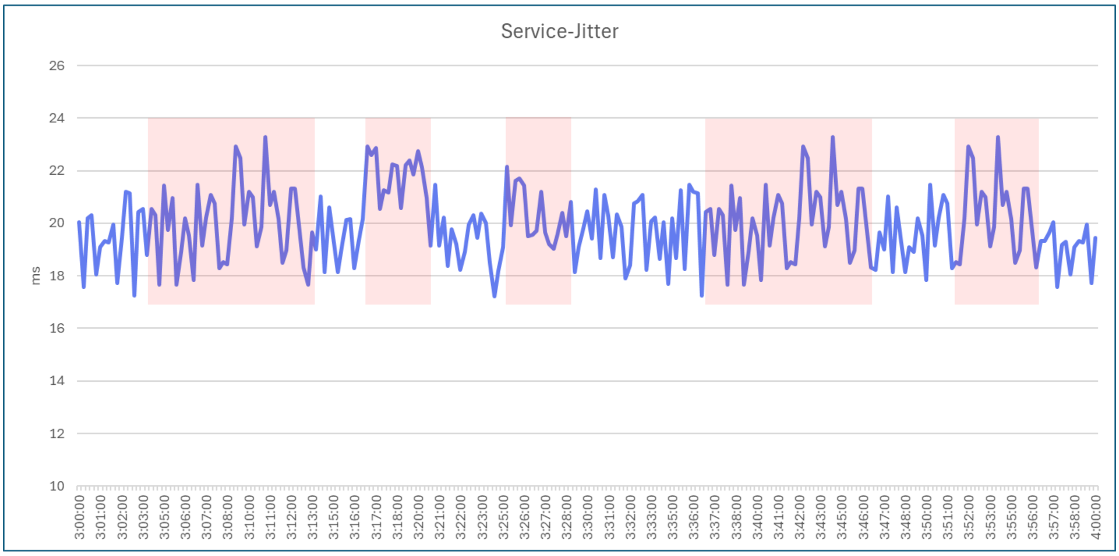


Figure 57: Service-jitter during test experience.

Red background timeslots are the duration when the orchestrator had been triggered to low quality connection and video streaming had been scaled down. The service-jitter average during whole 60 minutes is 19.86 ms while the average for the slots when there is good connection quality and slots when the orchestrator is triggered because of low connection quality is 21.45 ms and 23.28 ms respectively.

Considering the only 1.82 ms of difference in video packet sending, it is concluded the dynamic orchestrator helps to keep the service-jitter change negligible during low connection quality. Hence, the user experience, as a KVI, will not be affected.

3.6.2 Results summary

The use case results show that the metrics extraction successfully performed from the network where further has been used to trigger the orchestrator. It has been validated that the dynamic orchestrator could efferently scale down the video streaming quality to adapt with connection quality.

As the service level, it is also concluded that the dynamic orchestrator helps to keep application-level experience very smooth, that gets very negligible effect from low connection quality.

4 Lesson learned and recommendations

The design, integration, and validation activities in WP4 revealed several important lessons regarding the deployment of advanced V2X (Vehicle-to-Everything) applications and their enabling 5G infrastructures. These lessons provide critical insights for future large-scale deployments:

Limitations of Current Messaging Standards (CAM/DENM) and Suggested Improvements

In Use Case 1 (Cooperative Perception), Scenario 1 (vehicles on collision course at an intersection), accurate collision estimation is only possible when vehicle speeds are constant, and the DENM notification system is limited to alerting only the two involved vehicles. To improve efficiency and safety, two major enhancements are suggested:

1. Broader information sharing from vehicles via richer message types, such as CPM that was used in scenario 2 and its complementary message Sensor Data Sharing Message (SDSM), moving beyond the now largely obsolete CAM/DENM framework, and
2. Establishing a cloud-based intelligent Traffic Management Center (TMC), similar to the C-ITS system at IDIADA, capable of collecting real-time data from vehicles, inferring traffic patterns, and issuing dynamic safety policies to vehicles and roadside units. This centralized intelligence could greatly enhance cooperative perception efficiency and safety in complex scenarios

Mapping Network KPIs to Service KPIs and Driving Conditions

It proved difficult to relate low-level network KPIs (latency, jitter, packet error rate) to higher-level service KPIs (e.g., cooperative perception accuracy, ToD responsiveness) and contextual driving conditions (e.g., vehicle speed, road layout). There is a clear need to establish systematic KPI correlation models linking network behavior to service performance under varying mobility patterns and environmental conditions. Such models are essential for automated Service-Level Agreement (SLA) assurance in V2X scenarios.

Need for Road and Network Digital Twins

The trials demonstrated that assessing V2X applications at scale is challenging using only physical testbeds. Accurate evaluation of mass-deployment scenarios (e.g., thousands of connected vehicles) requires the creation of road digital twins and network digital twins (NDTs). These twins can emulate dense vehicular traffic, diverse network conditions, and dynamic radio environments, enabling the study of system-level behavior, scalability, and resilience without the prohibitive cost and complexity of full physical deployment.

Tight Integration of Predictive QoS and V2X Applications

The results of the predictive QoS (pQoS) for Tele-Operated Driving (ToD) use case clearly show that reliable ToD performance depends on bidirectional data exchange between the network and V2X applications. Network-side information about congestion, radio conditions, and resource allocation must be made accessible to applications, while applications must provide service-level requirements (e.g., latency budgets, bandwidth needs) back to the network. This confirms the need for standardized interfaces enabling real-time pQoS feedback loops, such as CAMARA project [9].

Challenges with Coarse network analytics exposure

Current network analytics functions typically expose aggregated KPIs at coarse time intervals. For instance, in IDIADA, the network status is updated every 15 minutes in the network Dashboard. This relatively long period can be reduced in some network implementation, but it is always in the order of minutes. For current services, there is no urgent need to make it lower. However, future safety-critical V2X applications demand second-level, fine-grained analytics (e.g., instantaneous RSRP, SINR, scheduling delays) to make timely decisions. Future networks must support network functions capable of exposing analytics with lower granularity, accessible through standardized APIs.

Challenges with Spectrum regulation

Current 5G Time Division Duplex (TDD) spectrum regulation focuses on harmonized TDD frame structure in all public network operators to ensure proper synchronization. The recommended TDD frame structures in Europe for the 3.5 GHz are DDDSU or DDDDDDDSUU [10], which means that the ratio of downlink to uplink resources is 3 to 1 or 7 to 2, with clear bias toward downlink. This is justifiable by the fact that most of current applications (e.g., video streaming, application downloads) require more downlink bandwidth than uplink bandwidth. However, this is not the case in many of automotive applications (e.g., cooperative perception using CPM, ToD as shown in this document), where much more bandwidth is required in uplink than downlink.

Challenges in Feedback from OEMs and Network Enabler Identification

Obtaining actionable feedback from automotive OEMs on required network enablers proved difficult, partly due to the absence of widely adopted APIs and reference frameworks. However, several gaps were identified during the project: the need for APIs similar to QoD (Quality of Data), APIs for more accurate positioning than CAMARA's current Location Verification API, and richer connectivity insights exposing more KPIs than currently available. This gap complicates the identification of network features critical for V2X use cases.

5 Summary and conclusions

The integration and validation activities conducted in WP4 of TARGET-X have successfully demonstrated the viability and benefits of 5G-enabled architectures for connected mobility use cases. Across all scenarios, the results confirm that deploying services closer to the edge reduces latency and improves reliability, while dynamic orchestration enables adaptive use of computational and network resources. Finally, the predictive quality of service mechanism showed that leveraging information exposed by network exposure API can make tele-operation experience safer. The measured KPIs consistently met or exceeded the target thresholds set in earlier phases of the project, validating the design decisions and implementation strategies adopted. Moreover, the lessons learned from system integration, field trials, and cross-partner collaboration provide valuable insights for future large-scale deployments. These outcomes reinforce the potential of 5G and beyond technologies to support advanced, safety-critical, and resource-intensive services, contributing to the broader objectives of the TARGET-X project and the EU SNS framework.

References

[1] "TARGET-X D4.1, Integrated pilot setup," Target-X project, 2023.

[2] "TARGET-X D4.2, "Automotive toolset and service implementation"," Target-X project, 2024.

[3] "TARGET-X D4.3, "Enhancement of automotive use cases with 5G and beyond"," Target-X project, 2025.

[4] S.B. Kamtam, et al., "Network Latency in Teleoperation of Connected and Autonomous Vehicles: A Review of Trends, Challenges, and Mitigation Strategies," Sensors 2024, 24, 3957. https://doi.org/10.3390/s24123957

[5] VILLASnode, https://www.fein-aachen.org/en/projects/villas-node/ [last visited17.09.2025]

[6] D. Kafetzis, N. Fotiou, S. Argyropoulos, J. Nasreddine and I. Koutsopoulos, "Video Quality Monitoring for Remote Autonomous Vehicle Control," 2025 IEEE International Mediterranean Conference on Communications and Networking (MeditCom), Nice, France, 2025, pp. 1-7, doi: 10.1109/MeditCom64437.2025.11104446.

[7] R. Rassool, "VMAF reproducibility: Validating a perceptual practical video quality metric," in IEEE Intl. Symp. on Broadband Multimedia Systems and Broadcasting (BMSB), 2017.

[8] R. Roca Oliver, J. Nasreddine, D. Camps-Mur, P. Salvati, "Reducing travel times of Tele-Operated vehicles through Connected Road Maps," The 27th International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 27–31 October, 2025, Barcelona, Spain.

[9] CAMARA project, https://camaraproject.org/ [last visited17.09.2025]

[10] 5GAA Automotive Association, "5g TDD synchronisation, guidelines and recommendations for the coexistence of TDD networks in the 3.5 Ghz range."