5G for Connected and Automated Road Mobility in the European Union

Deliverable D2.3
5G-CARMEN Final System Architecture and Interfaces Specifications
### Project Details

<table>
<thead>
<tr>
<th>Call</th>
<th>H2020-ICT-18-2018</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type of Action</td>
<td>IA</td>
</tr>
<tr>
<td>Project start date</td>
<td>01/11/2018</td>
</tr>
<tr>
<td>Duration</td>
<td>45 months</td>
</tr>
<tr>
<td>GA No</td>
<td>825012</td>
</tr>
</tbody>
</table>

### Deliverable Details

<table>
<thead>
<tr>
<th>Deliverable WP:</th>
<th>WP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deliverable Identifier:</td>
<td>D2.3</td>
</tr>
<tr>
<td>Deliverable Title:</td>
<td>5G-CARMEN Final System Architecture and Interfaces Specifications</td>
</tr>
<tr>
<td>Editor(s):</td>
<td>Roberto Fantini (TIM), Ana Cantarero (BMW), Ilaria De Biasi (BRE), Dimitri Marandin (CMA), Mauro Femminella, Claudio Casetti, Paolo Giaccone, Marco Ajmone (CNIT), Filippo Visintainer (CRF), Edwin Fischer (DTAG), Antonio Francescon (FBK), Zarrar Yousaf, Marco Liebsch (NEC), Alberto Bellini, Thomas Novak (SWARCO), Roberto Fantini (TIM), Andreas Georgakopoulos (WINGS)</td>
</tr>
<tr>
<td>Author(s):</td>
<td>Victor Garrido Martinez (BMW), Edwin Fischer (DTAG), Jochen Hoidis (QCM), Giuseppe Avino (TIM)</td>
</tr>
<tr>
<td>Reviewer(s):</td>
<td>31/03/2022</td>
</tr>
<tr>
<td>Contractual Date of Delivery:</td>
<td>24/05/2022</td>
</tr>
<tr>
<td>Submission Date:</td>
<td>PU</td>
</tr>
</tbody>
</table>

### Disclaimer

The information and views set out in this deliverable are those of the author(s) and do not necessarily reflect the official opinion of the European Union. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.
Table of Contents

LIST OF ACRONYMS AND ABBREVIATIONS .................................................................................. 4
LIST OF FIGURES .......................................................................................................................... 7
EXECUTIVE SUMMARY ............................................................................................................... 9

1 INTRODUCTION .......................................................................................................................... 11

2 USE CASES FUNCTIONAL ANALYSIS AND REQUIREMENTS .................................................. 12
  2.1 COOPERATIVE AND AUTOMATED LANE-CHANGE MANOEUVRES ........................................ 12
    2.1.1 Centralized Cooperative and Automated Lane Change Manoeuvre ..................................... 12
    2.1.1.1 Functional Analysis ........................................................................................................... 12
    2.1.1.2 Cross-Border Management ............................................................................................... 19
    2.1.2 Decentralized Cooperative and Automated Lane Change Manoeuvre .................................. 25
    2.1.2.1 Functional Analysis ........................................................................................................... 25
    2.1.2.2 Crossborder management ................................................................................................... 28
  2.2 COOPERATIVE AND AUTOMATED IN-LANE MANOEUVRES .................................................. 28
    2.2.1 Functional Analysis ............................................................................................................... 29
    2.2.2 Crossborder management ..................................................................................................... 33
  2.3 LANE CLEARANCE FOR EMERGENCY VEHICLES .................................................................. 35
    2.3.1 Functional Analysis ............................................................................................................... 35
    2.3.2 Crossborder management ..................................................................................................... 38

3 FINAL 5G-CARMEN SYSTEM ARCHITECTURE ......................................................................... 41
  3.1 OVERALL ARCHITECTURE DESCRIPTION ............................................................................ 41
  3.2 REFERENCE POINTS AND INTERFACES ................................................................................. 41
    3.2.1 Interfaces between vehicles and infrastructure ................................................................. 42
      3.2.1.1 Device-Device interface: Uses PC5 between vehicles .................................................... 42
      3.2.1.2 Uu interface: Interface between Device and Mobile Network Base Station which can be 4G/5G ................................................................................................................................. 42
      3.2.1.3 RSU-Vehicle interface ....................................................................................................... 44
    3.2.2 Orchestrated Edge Platform for CCAM services – components and interfaces .................... 44
      3.2.2.1 Functional Architecture and Key Enablers ....................................................................... 44
      3.2.2.2 Key components of the orchestrated edge platform ........................................................... 45
      3.2.2.3 Reference points ............................................................................................................... 46
    3.2.3 Interfaces in the MEC platform ............................................................................................. 48
      3.2.3.1 Interfaces between functional elements in the MECs to handle cooperative manoeuvring use cases ................................................................................................................................. 48
      3.2.3.2 Radio Network Information Service (RNIS) interface ....................................................... 55
    3.2.4 Cloud backend interfaces ..................................................................................................... 57
      3.2.4.1 Interface with road-side sensors and Cloud Analytics platform ........................................ 57
      3.2.4.2 Autobrennero Road Operator (C-ITS) backend ............................................................... 57
      3.2.4.3 Predictive Quality of Service (pQoS) API ......................................................................... 58
      3.2.4.4 Precise Positioning .......................................................................................................... 59

4 WAY FORWARD .......................................................................................................................... 61
  4.1 EVOLUTION OF 5G TO SUPPORT URLLC ......................................................................... 61
  4.2 S-LDM SCALABILITY ANALYSIS ............................................................................................ 66

5 CONCLUSION ............................................................................................................................. 67

REFERENCES .................................................................................................................................... 68
## List of Acronyms and Abbreviations

<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3GPP</td>
<td>Third Generation Partnership Project</td>
</tr>
<tr>
<td>5G</td>
<td>Fifth Generation (mobile network)</td>
</tr>
<tr>
<td>5QI</td>
<td>5G QoS Indicator</td>
</tr>
<tr>
<td>AF</td>
<td>Application Function (5G)</td>
</tr>
<tr>
<td>AI</td>
<td>Artificial Intelligence</td>
</tr>
<tr>
<td>AMF</td>
<td>Core Access and Mobility Management Function (5G)</td>
</tr>
<tr>
<td>AMQP</td>
<td>Advanced Message Queuing Protocol</td>
</tr>
<tr>
<td>APN</td>
<td>Access Point Name</td>
</tr>
<tr>
<td>BSA</td>
<td>Back Situation Awareness</td>
</tr>
<tr>
<td>BSAF</td>
<td>Back Situation Awareness Functionality</td>
</tr>
<tr>
<td>BSR</td>
<td>Buffer Status Report</td>
</tr>
<tr>
<td>CAM</td>
<td>Cooperative Awareness Message</td>
</tr>
<tr>
<td>CCAM</td>
<td>Cooperative, Connected and Automated Mobility</td>
</tr>
<tr>
<td>CG</td>
<td>Configured Grant</td>
</tr>
<tr>
<td>C-ITS</td>
<td>Cooperative Intelligent Transport System</td>
</tr>
<tr>
<td>C-ITS-S</td>
<td>Cooperative Intelligent Transport System Server</td>
</tr>
<tr>
<td>CLC</td>
<td>Cooperative Lane Change</td>
</tr>
<tr>
<td>CPM</td>
<td>Cooperative Perception Message</td>
</tr>
<tr>
<td>CSI</td>
<td>Channel State Information</td>
</tr>
<tr>
<td>C-V2X</td>
<td>Cellular Vehicle-to-Everything</td>
</tr>
<tr>
<td>DA</td>
<td>Dissemination Area</td>
</tr>
<tr>
<td>DATEX</td>
<td>Electronic language used in Europe for the exchange of traffic information</td>
</tr>
<tr>
<td></td>
<td>and traffic data (between traffic centres and motorway operators)</td>
</tr>
<tr>
<td>DC</td>
<td>Dual Connectivity</td>
</tr>
<tr>
<td>DCI</td>
<td>Downlink Control Information</td>
</tr>
<tr>
<td>DENM</td>
<td>Decentralized Environmental Notification Message</td>
</tr>
<tr>
<td>DL</td>
<td>Downlink</td>
</tr>
<tr>
<td>ECU</td>
<td>Electronic Control Unit</td>
</tr>
<tr>
<td>eMBB</td>
<td>Enhanced Mobile BroadBand</td>
</tr>
<tr>
<td>eNB</td>
<td>Evolved Node B (LTE)</td>
</tr>
<tr>
<td>EPC</td>
<td>Evolved Packet Core</td>
</tr>
<tr>
<td>EPS</td>
<td>Evolved Packet System</td>
</tr>
<tr>
<td>ETA</td>
<td>Expected Time of Arrival</td>
</tr>
<tr>
<td>ETSI</td>
<td>European Telecommunications Standards Institute</td>
</tr>
<tr>
<td>FDD</td>
<td>Frequency Division Duplexing</td>
</tr>
<tr>
<td>FWA</td>
<td>Fixed Wireless Access</td>
</tr>
<tr>
<td>GBR</td>
<td>Guaranteed Bit Rate</td>
</tr>
<tr>
<td>GD</td>
<td>Green Driving</td>
</tr>
<tr>
<td>gNB</td>
<td>Next Generation Node B (5G)</td>
</tr>
<tr>
<td>HARQ</td>
<td>Hybrid Automatic Repeat reQuest</td>
</tr>
<tr>
<td>HSS</td>
<td>Home Subscriber System</td>
</tr>
<tr>
<td>HSVW</td>
<td>Hybrid stationary Vehicle Warning</td>
</tr>
<tr>
<td>HTTP</td>
<td>HyperText Transfer Protocol</td>
</tr>
<tr>
<td>I2V</td>
<td>Infrastructure-to-Vehicle</td>
</tr>
<tr>
<td>IDCS</td>
<td>Intrusion Detection and Classification System</td>
</tr>
<tr>
<td>IDS</td>
<td>Intrusion Detection System</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>IQN</td>
<td>In-advance QoS Notification</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transport System</td>
</tr>
<tr>
<td>IVIM</td>
<td>In-Vehicle Information Message</td>
</tr>
<tr>
<td>KPI</td>
<td>Key Performance Indicator</td>
</tr>
<tr>
<td>LBO</td>
<td>Local Breakout</td>
</tr>
<tr>
<td>LCM</td>
<td>Lifecycle Management</td>
</tr>
<tr>
<td>LDM</td>
<td>Local Dynamic Map</td>
</tr>
<tr>
<td>LTE</td>
<td>Long-Term Evolution</td>
</tr>
<tr>
<td>MAC</td>
<td>Medium Access Control</td>
</tr>
<tr>
<td>MANO</td>
<td>Management and Orchestration</td>
</tr>
<tr>
<td>MEAO</td>
<td>Mobile Edge Application Orchestrator</td>
</tr>
<tr>
<td>MEC</td>
<td>Multi-access Edge Computing</td>
</tr>
<tr>
<td>MME</td>
<td>Mobility Management Entity</td>
</tr>
<tr>
<td>mMTC</td>
<td>Massive Machine Type Communication</td>
</tr>
<tr>
<td>MNO</td>
<td>Mobile Network Operator</td>
</tr>
<tr>
<td>NEF</td>
<td>Network Exposure Function (5G)</td>
</tr>
<tr>
<td>NF</td>
<td>Network Function</td>
</tr>
<tr>
<td>NFV</td>
<td>Network Functions Virtualization</td>
</tr>
<tr>
<td>NFV-LO</td>
<td>NFV Local Orchestrator</td>
</tr>
<tr>
<td>NFV-SO</td>
<td>NFV Service Orchestrator</td>
</tr>
<tr>
<td>NFVI</td>
<td>Network Function Virtualization Infrastructure</td>
</tr>
<tr>
<td>NR</td>
<td>New Radio</td>
</tr>
<tr>
<td>NS</td>
<td>Network Service</td>
</tr>
<tr>
<td>NSA</td>
<td>Non-Standalone (5G network)</td>
</tr>
<tr>
<td>OBU</td>
<td>On-Board Unit</td>
</tr>
<tr>
<td>ODD</td>
<td>Operational Design Domain</td>
</tr>
<tr>
<td>OEM</td>
<td>Original Equipment Manufacturer</td>
</tr>
<tr>
<td>OFDM</td>
<td>Orthogonal Frequency Division Multiplexing</td>
</tr>
<tr>
<td>PCF</td>
<td>Policy Control Function (5G)</td>
</tr>
<tr>
<td>PDU</td>
<td>Protocol Data Unit</td>
</tr>
<tr>
<td>PDCCCH</td>
<td>Physical Downlink Control Channel</td>
</tr>
<tr>
<td>PDSCH</td>
<td>Physical Downlink Shared Channel</td>
</tr>
<tr>
<td>PGW</td>
<td>Packet Data Network Gateway</td>
</tr>
<tr>
<td>PI</td>
<td>Puncturing Indication</td>
</tr>
<tr>
<td>PLMN</td>
<td>Public Land Mobile Network</td>
</tr>
<tr>
<td>pQoS</td>
<td>Predictive Quality of Service</td>
</tr>
<tr>
<td>PUCCH</td>
<td>Physical Uplink Control Channel</td>
</tr>
<tr>
<td>PUSCH</td>
<td>Physical Uplink Shared Channel</td>
</tr>
<tr>
<td>QCI</td>
<td>QoS Class Identifier</td>
</tr>
<tr>
<td>QFI</td>
<td>QoS Flow ID</td>
</tr>
<tr>
<td>QoE</td>
<td>Quality of Experience</td>
</tr>
<tr>
<td>QoS</td>
<td>Quality of Service</td>
</tr>
<tr>
<td>RAN</td>
<td>Radio Access Network</td>
</tr>
<tr>
<td>RNIS</td>
<td>Radio Network Information Service</td>
</tr>
<tr>
<td>RQoS</td>
<td>Reflective QoS</td>
</tr>
<tr>
<td>RRC</td>
<td>Radio Resource Control</td>
</tr>
<tr>
<td>RSU</td>
<td>Roadside Unit</td>
</tr>
<tr>
<td>SA</td>
<td>Situation Awareness</td>
</tr>
<tr>
<td>SA</td>
<td>Standalone</td>
</tr>
<tr>
<td>SAE</td>
<td>Society of Automotive Engineers</td>
</tr>
<tr>
<td>SCS</td>
<td>Sub-Carrier Spacing</td>
</tr>
<tr>
<td>SFI</td>
<td>Slot Format Indication</td>
</tr>
<tr>
<td>SG</td>
<td>Scheduling Grant</td>
</tr>
<tr>
<td>Acronym</td>
<td>Full Form</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
</tr>
<tr>
<td>SGW</td>
<td>Serving Gateway</td>
</tr>
<tr>
<td>SIM</td>
<td>Subscriber Identity Module</td>
</tr>
<tr>
<td>S-LDM</td>
<td>Server LDM</td>
</tr>
<tr>
<td>SMF</td>
<td>Session Management Function (5G)</td>
</tr>
<tr>
<td>SR</td>
<td>Scheduling Request</td>
</tr>
<tr>
<td>TCC</td>
<td>Traffic Control Centre</td>
</tr>
<tr>
<td>TCP</td>
<td>Transport Control Protocol</td>
</tr>
<tr>
<td>TDD</td>
<td>Time Division Duplexing</td>
</tr>
<tr>
<td>TEID</td>
<td>Tunnel Endpoint Identifier</td>
</tr>
<tr>
<td>UCI</td>
<td>Uplink Control Information</td>
</tr>
<tr>
<td>UDM</td>
<td>Unified Data Management (5G)</td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
</tr>
<tr>
<td>UE</td>
<td>User Equipment</td>
</tr>
<tr>
<td>UL</td>
<td>Uplink</td>
</tr>
<tr>
<td>UPF</td>
<td>User Plane Function (5G)</td>
</tr>
<tr>
<td>URLLC</td>
<td>Ultra-Reliable Low Latency Communication</td>
</tr>
<tr>
<td>USIM</td>
<td>Universal SIM</td>
</tr>
<tr>
<td>Uu</td>
<td>Interface between UE and Mobile Network Base station (Air Interface)</td>
</tr>
<tr>
<td>V2I</td>
<td>Vehicle-to-Infrastructure</td>
</tr>
<tr>
<td>V2N2V</td>
<td>Vehicle-to-Network-to-Vehicle</td>
</tr>
<tr>
<td>V2V</td>
<td>Vehicle-to-Vehicle</td>
</tr>
<tr>
<td>VAS</td>
<td>Value Added Service</td>
</tr>
<tr>
<td>VIM</td>
<td>Virtual Infrastructure Manager</td>
</tr>
<tr>
<td>VNF</td>
<td>Virtual Network Function</td>
</tr>
<tr>
<td>V-LDM</td>
<td>Vehicle LDM</td>
</tr>
<tr>
<td>VS</td>
<td>Video Streaming</td>
</tr>
<tr>
<td>VSSS</td>
<td>Vehicle Sensors and State Sharing</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1 Centralized Cooperative Lane Change: Functional Components (In-Country) ................13
Figure 2 CLC: End-to-end message flow including interfaces. ........................................13
Figure 3 CLC: On-board Architecture ........................................................................14
Figure 4 Manoeuvring Service – system diagram. ..........................................................15
Figure 5 S-LDM high-level architecture in the framework of the CLC service ..................16
Figure 6 High-level architecture of the S-LDM in a more general service scenario ..........16
Figure 7 Message flow for the Centralized CLC: processing details inside MEC ..........18
Figure 8: A. The basic architecture for a single domain installation of the centralized CLC. B. A close-up to the internal organization of topic and selectors in the Message Broker ..................19
Figure 9 Centralized Approach: cross-border without dynamic endpoint notification ....20
Figure 10: Single Domain scenario extended with dynamic AMQP Broker endpoint management ..22
Figure 11 Centralized Approach: cross-border with dynamic endpoint notification ........25
Figure 12 Decentralized Cooperative and Automated Lane Change Manoeuvre in the sub-case of lane clearance for emergency vehicle ......................................................26
Figure 13 Decentralized Lane Change Manoeuvre: CAMs and CPMs are transferred via Uu at 20Hz. PC5 transfers 10Hz CAM. Neighbour vehicles (C, B, D) are stored in the V-LDM ..........27
Figure 14 Sequence diagram of decentralized lane change manoeuvres .......................27
Figure 15 In-lane manoeuvres based on lateral detection (prevented lane change) (scenario 2b., from D5.1) .............................................................29
Figure 16 Message flow for Cooperative and automated in-lane manoeuvres and for the decentralized lane change manoeuvres, in country ..................................................31
Figure 17 Sequence diagram for V2V communication, with focus on the messages triggering the changes of manoeuvre .................................................................32
Figure 18 Sequence diagram for LDM communication. The S-LDM, triggered by vehicle A lane change indication becomes very responsive and gives A fast updates on the surrounding environment ...............................................................33
Figure 19 Cross-border management of in-lane manoeuvres in the V2V option (valid also for decentralized lane-change) .........................................................33
Figure 20 in-lane manoeuvres in a simple case of forward detection .............................34
Figure 21 Lane clearance for emergency vehicle: the host vehicle (A) triggered by the emergency vehicle notification, performs a decentralized lane change ..........................................35
Figure 22 Overview of BSA functional components. ......................................................36
Figure 23 Sequence diagram highlighting BSA interaction in lane clearance for emergency vehicle ........................................................38
Figure 24 Back Situation Awareness cross-border (architecture view) .........................39
Figure 25 Back Situation Awareness cross-border (scenario view) ............................40
Figure 26 5G-CARMEN System Architecture incl. key reference points ..................41
Figure 27 EN-DC overall architecture (as shown in 3GPP TS 37.340 version 15.3.0 Release 15). 43
Figure 28 Functional architecture and key references points for Orchestrated 5G edges platform. 45
Figure 29 Graphical visualization of the area corresponding to the Quadkeys calculated by the S-LDM when the range is "46.0201280: 11.0279790-47.307088: 11.686929"...

Figure 30 Portion of the proposed JSON message format sent by the S-LDM towards the Manoeuvring Service...

Figure 31 RNIS interfaces in software architecture...

Figure 32 A22 C-ITS system infrastructure...

Figure 33 pQoS overview...

Figure 34 Precise Positioning architecture...

Figure 35 FDD timing relations for (a) LTE; (b) NR as shown in [23].
Executive Summary

This document provides the final version of the 5G-CARMEN system architecture and its interfaces. To this end, it builds on top of the functional analysis of the use cases that has been performed in the previous Deliverable D2.2, which presented a preliminary view, and extends it to further elaborate on how these use cases can be handled in the cross-border scenario, that occurs when a user moves from one country to the next one, switching from its national Mobile Network Operator (MNO) to a visited MNO.

As explained in D2.2, 5G-CARMEN has reassessed the four originally selected use cases (i.e., Cooperative Manoeuvring, Situation Awareness, Video Streaming and Green Driving), to reshape and prioritize those with stronger requirements for analysis and trialling in the last part of the project. The result was the following set of cooperative manoeuvring use cases:

- Cooperative and automated lane-change manoeuvres, which is divided in two sub-use cases based on whether the manoeuvre is handled with a centralized or decentralized approach:
  - Centralized lane change, with V2V and low-latency V2N2V communication, with network recommendation;
  - Decentralized lane change with V2V and low-latency V2N2V communication, without network recommendation.

- Cooperative and automated in-lane manoeuvres, with longitudinal control and lane keeping/centring based on forward and lateral detection of event (e.g., communicating cars + sensed cars).

Both use cases can be triggered by the arrival of an emergency vehicle (EmV), that can be seen as an additional sub-case:

- Lane clearance for an Emergency Vehicle.

These use cases leverage on different components that have been deployed in the domains of the three MNOs involved in the project. The present document describes the interactions that occur between these components to support service continuity also when vehicles are crossing the border, and different vehicles might be served by different MNOs networks. Different cross-border approaches are described, depending on the characteristics of the use case considered. For the centralized Cooperative and automated lane-change manoeuvre, cross-border management is granted by allowing certain functional entities (namely the Server Local Dynamic Map and the Response Router) to subscribe to the message brokers of different MNOs. Additional entities, the Main Manager and the Local Manager, are then added to handle Dynamic AMQP Broker endpoint management, in particular in Shared Areas along the border. For the decentralized Cooperative and automated lane-change manoeuvre and the Cooperative and automated in-lane manoeuvre, cross-border is managed through specific interfaces that are available on the GeoService entity that acts as message broker for these use cases. Finally, the Lane Clearance for an Emergency Vehicle sub-use case leverages on the service continuity capabilities granted by the Orchestrated Edge platform for CCAM services defined in Work Package 4.

Following this extended functional analysis for these prioritized 5G-CARMEN use cases, this deliverable depicts the final 5G-CARMEN architecture, which basically confirms the preliminary version that has been described in Deliverable D2.2. The architecture has been divided into several layers: first, the street layer with vehicles and RSUs connected directly via PC5 and to the other components via the Uu interface. Second, the access layer represented by the eNBs/gNBs from the local MNOs, as well as the existing Roadside infrastructure connections to the respective Backends. Third, the Edge Clouds with MEC platforms in each country which are integrated into the Access/Core networks of the MNOs. This layer is further structured over 2 sub-layers: the MEC service orchestration components that orchestrates functions within the MEC and between different MECs and CCAM services that group application level services for specific use cases and value added services enabling communication at application level, like sharing of ITS messages. The last layer
consists of various Cloud components, including existing systems such as OEM backends and dedicated Services supporting the use cases. The document provides more details about the reference points and interfaces on these different layers.

The last part of the document discusses the current limits of nowadays commercial mobile networks, and how the evolution of 5G defined in the 3GPP standard will enable new features to support Ultra-Reliable Low Latency Communication (URLLC) services, which could be integrated in the proposed architecture. Finally, it shows how the approach followed in the definition of 5G-CARMEN architecture can ensure scalability for the different components necessary to the realization of the project use cases, taking one of the most demanding components as example.
1 Introduction

This is the third public deliverable from Work Package 2 (WP2) of the 5G-CARMEN project. This document provides the final version of the 5G-CARMEN system architecture including the specifications of its sub-components, their interfaces, and protocols to be used for the data exchange. To this end, the functional analysis of the use cases presented in Deliverable D2.2 [2] is summarized in Chapter 2 and extended to detail the solutions that have been deployed to handle the cross-border scenario, that occurs when a user moves from one country to the next one, switching from its national Mobile Network Operator (MNO) to a visited MNO.

This functional analysis is performed for the two main use cases that are trialled in the project: the Cooperative and Automated Lane Change Manoeuvre (in its centralized and decentralized version) and the Cooperative and Automated In-Lane Manoeuvre, and also for the sub-use case of Lane clearance for an Emergency Vehicle (EmV), which describes a specific service that has been deployed in the Edge Orchestration platform developed in the project, and that can trigger the first use case by either requesting to free the lane with a Cooperative and Automated Lane Change Manoeuvre or to keep it free by maintaining vehicles in the other lanes with Cooperative and Automated In Lane Manoeuvres.

These use cases leverage on different components that have been deployed in the domains of the MNOs in the three countries involved in the project, Germany, Austria, and Italy. The present document describes the interactions that occur between these components to support service continuity also when vehicles are crossing the border, and different vehicles might be served by different MNOs networks. Different cross-border approaches are described, depending on the characteristics of the use case considered.

Chapter 3 provides a brief overview of the final system architecture, recalling its structure, which however has not been modified compared to what was illustrated with higher detail in the previous Deliverable D2.2 [2]. The chapter then focuses on the interfaces defined between the functional elements that compose the architecture, providing a detailed description of: (i) the interfaces between vehicles and the other elements in the architecture, (ii) the interfaces to handle the Orchestrated Edge Platform for CCAM services defined by the project in Work Package 4, (iii) the interfaces between specific services running on the MEC platforms of the MNOs and that have been deployed specifically to support the trialled used cases, and (iv) the interfaces towards additional services that are running in the Cloud.

The last chapter discusses the current limits of nowadays commercial mobile networks, and how the evolution of 5G defined in the 3GPP standard will enable new features to support Ultra-Reliable Low Latency Communication (URLLC) services, which could be integrated in the proposed architecture. Finally, it shows how the approach followed in the definition of 5G-CARMEN architecture can ensure scalability for the different components necessary to the realization of the project use cases, taking one of the most demanding components as example.
2 Use Cases Functional Analysis and Requirements

In this section we recall the functional analysis of 5G-CARMEN use cases described in D2.2 [2] and we extend it to detail how these use cases can be handled in a cross-border scenario.

As explained in D2.2, 5G-CARMEN has reassessed the four originally selected use cases (i.e., Cooperative Manoeuvring, Situation Awareness, Video Streaming and Green Driving), to reshape and prioritize those with stronger requirements for analysis and trialling in the last part of the project.

The result is the following set of cooperative manoeuvring use cases:

- Cooperative and automated lane-change manoeuvres, which is divided in two sub-use cases based on whether the manoeuvre is handled with a centralized or decentralized approach:
  - Centralized lane change, with V2V and low-latency V2N2V communication, with network recommendation;
  - Decentralized lane change with V2V and low-latency V2N2V communication, without network recommendation.
- Cooperative and automated in-lane manoeuvres, with longitudinal control and lane keeping/centring based on forward and lateral detection of event (e.g., communicating cars + sensed cars).

Both use cases can be triggered by the arrival of an emergency vehicle (EmV), that can be seen as an additional sub-case, and will be therefore analysed hereafter:

- Lane clearance for an Emergency Vehicle.

5G has arrived and is already shaping the future of connected vehicles, with the goal of ensuring a safer, more efficient, and enjoyable driving experience. As mobility between different states and countries increases, the need to improve traffic efficiency does as well. By leveraging on 5G’s low-latency communication, the network can now play a big role for cooperative manoeuvring scenarios.

2.1 Cooperative and automated lane-change manoeuvres

2.1.1 Centralized Cooperative and Automated Lane Change Manoeuvre

5G-CARMEN focuses on lane-change manoeuvres as main use case for centralized vehicle coordination, due to its importance in traffic oscillation and its high requirements on positioning accuracy and network reliability. More details on architecture and functional analysis are found in the following sections.

2.1.1.1 Functional Analysis

This subsection summarizes the interactions based on the functional representation of the centralized cooperative and automated lane-change manoeuvres use case. A more detailed description is found on Deliverable D2.2 [2].

Figure 1 shows the main functional elements involved in the realization of the use case, as they were described in Deliverable D2.2 [2]. The role of the elements hosted in the MEC is also summarized hereafter, while the next section discusses how elements hosted in different MECs should interact to handle cross-border scenarios, and the additional elements that have been implemented to this aim.

Besides defining the main functional components of the overall system, the interfaces between each one of them have been defined and implemented. A general overview is presented in Figure 2, a more detailed description of each interface is found in Section 3.2.3.1.
Figure 1 Centralized Cooperative Lane Change: Functional Components (In-Country).

Figure 2 CLC: End-to-end message flow including interfaces.
Vehicle On-Board Architecture

The main functionalities of the on-board components and its corresponding applications are:

- Read vehicle data (Speed, positioning, heading, blinker) from CAN bus;
- Read positioning data from high precision GNSS (if available);
- Connectivity to AMQP broker;
- Encoding/Decoding and sending/receiving CAM/DENM messages to the manoeuvring service;
- Processing of recommendations from the manoeuvring service (display, automated vehicle control).

A more detailed view of each vehicle component and its interactions is presented in Figure 3:

![Figure 3 CLC: On-board Architecture.](image)

Manoeuvring Service

The Manoeuvring Service’s task is to monitor the current state of the road traffic and the intentions of vehicles; determine the most suitable moment to execute a lane change if requested; manage the
gaps between vehicles, such that the manoeuvre can be done safely and efficiently; and generate recommendations for vehicles to follow if possible.

To allow flexible recommendations and be as close to the State of the Art as possible, a Model-Based Control Approach was chosen [16]. The proposed algorithm consists of a Tactical- and Operational-Controller. The Tactical-Controller computes the most efficient merging sequence and time instant. The Operational-Controller generates the optimal control output for the car. The initial algorithm implementation utilizes the Tactical-Controller only, while the Operational Controller will be left for a later development.

The following Figure 4 shows a complete system diagram.

![Figure 4 Manoeuvring Service – system diagram.](image)

**Server Local Dynamic Map (S-LDM)**

The Server Local Dynamic Map (S-LDM) is a 5G-enabled MEC service acting as a facility for the Centralized Lane Change, providing the Manoeuvring Service with all the data needed to successfully manage the use case.

Its main target is to store and process the data of objects on the roads (both connected, i.e., vehicles sending CAM messages, and non-connected, i.e., vehicles not sending CAM messages and other obstacles), and, when a certain triggering situation is detected (e.g., a vehicle initiating a lane merge), to provide all the relevant data, plus a precise context and relative positioning of objects, to other services, which will use them to manage L4 automated driving in the best possible way. Only the data...
relevant to a specific situation, both in a raw and pre-processed format, is provided, enabling an optimal usage of the underlying computational and memory resources.

Figure 5 presents the high-level architecture of the S-LDM service in the framework of the Centralized Lane Change service. Despite being developed as part of the Cooperative Lane Change scenario, the S-LDM can provide its information to any service and it is developed with a modular and easily extensible paradigm. Figure 6 provides a more general view of the S-LDM, where it provides its information to any service.

**Figure 5** S-LDM high-level architecture in the framework of the CLC service.

**Figure 6** High-level architecture of the S-LDM in a more general service scenario.
**Message Broker**

The message broker is a software component that implements the publish/subscribe communication paradigm. The broker decouples senders (publishers) and receivers (subscribers), and acts as a message router. Messages are discrete chunk of data, opaque to the broker – this means that the broker is data agnostic, unaware of the information contained in the messages. The publish/subscribe mechanism enables clients to communicate one-to-one, one-to-many and many-to-one.

Specifically for the centralized CLC scenario, the Message Broker is providing two dedicated topics:

- **fromCARS**: meant for the messages coming from vehicles on the monitored road, which feed the service;
- **toCARS**: meant for the service messages (e.g., recommendation messages) sent to the cars.

Moreover, the AMQP Broker provides a mechanism (Selectors) that allows selecting the final receiver(s) for a given AMQP message according to the value of some header fields of the message itself. The selection rule is specified in a selector clause that can be optionally specified by any AMQP Broker subscriber in its subscribing request. Thus, if a client subscribed to a topic without providing any selector rule, it will receive all the messages produced on that topic. Otherwise, only messages matching the selector clause will be forwarded to such a client. It is worth underlying that the selector clause may leverage ad-hoc defined key-value pairs in the header of the AMQP message.

**Response Router**

The Response Router module is in charge of receiving messages that are due to be delivered to a specific car via Message Broker.

The Response Router relies on the properties' definition feature offered by the Message Broker to achieve a successful aimed delivery. This feature allows Message Broker to filter the receivers for a given AMQP message according to a set of values specified in the header of that message.

The Message Broker allows each car to subscribe to a topic (toCARS) to receive messages from the service. In the subscription process, a vehicle shall also provide a value for the property "carID": this value identifies the car and the messages that are associated with it.

On the other side, the Response Router publishes each recommendation message encapsulated in a dedicated AMQP message on the toCARS topic, specifying in the header of each AMQP message the car ID property.

Thanks to the binding provided by the carID property, the Message Broker can select the receivers associated with each message and then send the message only to them. Since each car has its univocal car ID, each recommendation message is delivered only to its destination car.

The message flow between all these MEC components is shown below in Figure 7:
Figure 7 Message flow for the Centralized CLC: processing details inside MEC.

Figure 8 (A) shows a basic setup for a single domain instance of centralized CLC service. Some cars are traveling on a road segment which is supposed to be monitored by the service. Each vehicle is connected to the Message Broker, both in uplink (to provide the service with its generated CAM messages) and downlink (to receive the recommendation messages generated by the service when triggering condition occurs).

On the other side of the Message Broker, the S-LDM module receives and processes the flow of CAM messages arriving from the cars. In particular, the S-LDM looks for those situations where a vehicle is going to change the lane in which it is currently traveling. When one of such situations is detected (for example, as shown in the figure, when the cyan car is initiating an overtake manoeuvre over the white car), this triggers a notification from the S-LDM to the Manoeuvring Service module. Such a notification consists in sending to the Manoeuvring Service the context of the event, i.e., a collection of the most recent CAM messages received by cars in a meaningful surrounding (i.e., an area of adequate radius) of the vehicle triggering the notification event. For example, with reference to the situation shown in Figure 8 (A), the context will include the CAM messages produced by the cyan and white car and the latest generated by the red one, which is reaching the other two cars at very high speed.

Once the Manoeuvring Service has performed its evaluation and generated a set of recommendation messages for each car in the context, it sends down all these messages to the Response Router, the module in charge of dispatching each message to the proper destination. To this purpose, the Response Router is connected to the Message Broker, and it leverages the downlink channel of the latter towards the cars to send down the recommendation messages to the proper destination. The arrival at the vehicles of the recommendation message closes the loop, with the car client software in charge of actuating the suggested solution or translating it into a human intelligible notification to the car driver.
Figure 8: A. The basic architecture for a single domain installation of the centralized CLC. B. A close-up to the internal organization of topic and selectors in the Message Broker.

Figure 8 (B) provides a zoom on the AMQP Broker connections with other architectural components and cars. It highlights the two topics (i.e., fromCARS and toCARS) defined to support the centralized CLC service. In addition, it emphasizes the selectors of these topics used by subscribers to allow the AMQP Broker to select the messages delivered to each subscriber automatically.

Since the S-LDM subscription to the fromCARS topic is finalized only to the CAM messages coming from the covered area, it subscribes to the remote Message broker by leveraging a geofencing selector clause, restricting the message delivery only to messages originated in the covered area. Such a selector clause is defined through a quadtree-based description of the geographical space where the road spans. Quadtree model allows to divide an area in a set of non-overlapping squares: a code univocally identifies each of these squares, and each point inside a square is also associated with that square code. A position value equal to a given quadtree code means that the car is inside the squared area related to that code. The quadtree model offers the chance for the implementer to tailor the desired service's quadtree resolution level: indeed, a higher resolution (smaller area square, i.e., shorter square edge length) simply translates into a longer code string.

Each AMQP message by vehicles should include its position encoded as quadtree code in the message header to have the mechanism work. Thus, in the subscription request to the Message Broker the S-LDM specifies the list of quadtree codes of interest so that only those messages matching one in the list are forwarded.

Similarly, each vehicle subscribes to the toCARS topic to enable notifications from the centralized CLC service. The precise delivery to the destination vehicle is achieved by the “carID”, i.e., a univocal identifying code provided to each subscribing vehicle. Suppose that code is present in the header of the AMQP message injected by the Response Router. In that case, that message is delivered only to the destinations matching such an identifier in their selector settings. Each vehicle uses the same carID both in subscribing to the toCARS topic and in its CAM messages sent to the S-LDM (with all the service modules referring that code for identifying the originating car). Thus, the consistency in using that identifier is preserved all along the process.

### 2.1.1.2 Cross-Border Management

Figure 9 extends the basic scenario shown in the previous Section 2.1.1.1 to the more complex situation of a multidomain scenario, with a border situation between two different operators as shown.
Two operators, A and B, are providing the centralized CLC service on two consecutive sections of a given road. On both sides of the border, each operator has its MEC, where the components of the centralized CLC are assumed to be deployed and running properly. Since radio network borders are not actually fixed, but subject of fluctuations that may depend on several environmental factors (fixed or time-varying), it is not possible to identify a clear border where all the cars are commuting from one operator to the other. To model such a situation, we identified an area close to the border to have it considered as a shared one: inside it, it is possible to have vehicles associated to one or the other operator. Instead, outside the shared area each vehicle is assumed to be connected to the operator officially covering it.

![Diagram of Centralized Approach: cross-border without dynamic endpoint notification.](image)

Such a shared area is the main difference from the previous scenario. It forces the S-LDM to rely on an additional external Message Broker to cover part of its coverage area (i.e., the portion of the shared area on the other side of the border), an area otherwise excluded from the local Message Broker monitoring. By extending the previously described approach based on quadtree selection clause, S-LDM subscribes also to the remote Message Broker providing the list of quadtrees covering that specific portion of the Shared Area.

In addition to this, the Response Router shall register as producer both to the local and remote Message Broker and send any message meant for a car to both Brokers, just to be sure it is going to be delivered to the destination car. Otherwise, if a car referred in a context switches from the original Message Broker to the other during the evaluation of a recommendation message, the moment that recommendation message would be ready it couldn’t be delivered to its destination.

With the introduction of the Shared Area and the double subscription to local and remote Message Broker for both S-LDM and Response Router, the centralized CLC is now capable of managing the uncertainty regarding the actual reference endpoint used by the vehicle. Indeed, independently from the associated Message Broker, each CAM message forwarded by a vehicle in the shared area is delivered to both the S-LDM instances covering that area, thus allowing each of them to retrieve all the context related info required in case of lane-change detection. And, similarly, each generated recommendation messages will be delivered to its destination, independently from vehicle’s message broker association.

A final consideration regarding the width of the shared area, with the help of an example. With reference to Figure 9, let a car being inside domain B, very close (closer than a context radius) to the right limit of the shared area but still connected to the Message Broker of Operator A. If, for any reason, a lane-change situation is triggered by that car, the context will be generated by Operator A’s S-LDM. Thus, the context will collect the CAMs of vehicles inside a context radius, but, by construction, this will span beyond the right border of the shared area, in an area that is totally
unknown to the evaluating S-LDM. So, to avoid an incomplete description of the context, the shared area should span a bit more (at least a context radius on both sides) than the actual area of potentially promiscuous connectivity.

**Dynamic AMQP Broker endpoint management**

The solutions described in the previous sections 2.1.1.1 and 2.1.1.2 are relying on the assumption that the vehicles are aware of the AMQP Broker endpoints to connect with, in every instant. This means that each vehicle has to retrieve and upkeep a list of the available endpoints for each area and which is the AMQP broker to connect with, according to vehicle position and network operator in use.

Maintaining an updated list of the broker endpoints and their associated coverage in addition to the continuous check of the AMQP broker to connect with, can be a time and resource consuming activity at car client side. Thus, enabling a notification system that provides such a subservice to the connected vehicles will be a remarkable added value, also considering the improvement in terms of maintainability and flexibility of the overall service.

The proposed solution is extending the previously described architecture by introducing two new components: the Local Manager and the Main Manager. They integrate with the other architecture components and enable a dynamic, MEC-hosted AMQP Broker endpoint management service toward each connected vehicle.

Let assume to have a vehicle that would like to join the centralized CLC service, but it is completely unaware regarding the AMQP broker(s) covering its position. Thus, an always reachable service component that provides that information based on car position would be a valuable solution. Moreover, such complementary service would be useful not only for accessing the service, but also for re-joining it in case of unexpected loss of connectivity. The Main Manager is the module that covers this sub-service. It is worth underlining that the service provided by the Main Manager can be deployed in the cloud, but in case of service connectivity recovery this one could be faster if performed with an instance at MEC side.

After having queried the Main Manager, the vehicle knows which broker endpoint to connect with, so it can actually join the service. The vehicle is then moving along the road and most likely at a certain point it will reach an area where it will be suggested to switch to another, closer endpoint (to continue benefitting from lower latency of the MEC solution). The entity which is going to inform the vehicle about the more convenient transition is the Local Manager, a module running at the same MEC of the AMQP Broker at which the car is currently connected with. Local Manager stores a list of candidate endpoints associated to the several sections of the road covered by that AMQP Broker. Indeed, it is possible that in some areas only one exclusive AMQP Broker is allowed (e.g., outside shared areas) whereas in other areas more instances are available (e.g., inside shared areas). Furthermore, endpoints may also depend on the network operator selected by the vehicle, thus the candidate list shall also take in account such a condition. Finally, still depending on the position, in a set of candidate endpoints some can be considered preferable to the others, thus defining a priority list among the candidates that the vehicle should/has to follow when trying to connect to the broker.

The Main and Local Manager combined activities provide the support to the connected vehicle for managing the not easy task of joining (or reconnecting to) the service and identify the most convenient endpoint, in every situation. Moreover, given the wider knowledge of the several AMQP broker coverage, we envisage the Main manager also as the component in charge of configuring (and reconfiguring, if needed) the Local Manager’s georeferenced endpoint list.
Figure 10: Single Domain scenario extended with dynamic AMQP Broker endpoint management.

Figure 10 shows how the new modules integrate into the architecture originally portrayed in Figure 8 (A). The Main Manager shall interact with vehicles and one or more Local Managers, while each Local Manager, in addition to the communication channel toward the Main Manager, is also subscribing the AMQP Broker and sending messages to the Response Router (to be dispatched to the vehicles via AMQP Broker).

In the following sub-sections we provide further details regarding the two additional modules

**Main Manager**

As previously said, the Main Manager is a reference component (at least, at Operator domain scale) for all the vehicles leveraging the centralized CLC service. It provides each requesting vehicle with the best centralized CLC and AMQP broker, according to vehicle position. This information can be used both when joining the service for the very first time and when a vehicle already part of the service gets disconnected and has no hint regarding the next Message Broker to connect with. In both cases, the Main Manager is supposed to receive a request from a querying vehicle containing a reference to the position of the vehicle.

The Main Manager is also supporting a set of Local Managers, by providing each of them with a dedicated list of AMQP brokers. The list has to be notified by the Local Manager to the vehicles, to keep them aligned with the covering access points to the service. After initialization, the Main Manager is supposed to retrieve (from a configuration file or directly form a database) the information about:

- Its served Local Managers;
- The list of georeferenced AMQP Broker candidates for each served Local Manager.

Then, the Main Manager will contact each Local Manager and provide a list of georeferenced AMQP broker endpoints. Moreover, whenever the Main Manager detects or is notified of a change in configuration affecting one or more of its served Local Managers, it shall contact each of them providing the new configuration.

Given the pivotal role, the wide coverage and the nature of the covered tasks, there is no real need for the Main Manager to be deployed at MEC side. Deploying it in the cloud could be a better
alternative considering the larger number of available computational resources. The only latency sensitive activity is when a vehicle queries the Main Manager after losing alignment with the service, but one of the goals of the suggested solution is also minimizing the occurrence of such events.

Local Manager

The Local Manager is a module instantiated and running at each MEC site. Its purpose is to provide the centralized CLC service-leveraging vehicles with information regarding the best-performing Message Brokers providing the service at the vehicle position.

The information regarding the Message Broker endpoints is provided and maintained by the Main Manager module. Instead, The Local Manager retrieves the current position of each vehicle by subscribing the fromCARS topic at the Message Broker (i.e., the same topic subscribed by the S-LDM) and checking each time the carID and position property of the message in the header of each AMQP message. According to the configuration passed by the Main Manager and the current position of a vehicle, the Local Manager may decide to send to that vehicle an updated list of Message Broker endpoints (support message). If so, the Local Manager, similarly to what done by the Manoeuvring Service module for the recommendation messages, sends the support message to the Response Router, leaving to it the responsibility for a fast dispatching to the right target vehicle.

Several policies can be implemented at the Local Manager side to trigger the generation of a support message. The simplest one sees the Local Manager generating a support message every time a message is published on the fromCARS topic. This solution is highly inefficient, since it finally leads to the injection of a number of support messages equal to the number of forwarded CAMs. This will burden the Local Manager, the Response Router as well as the vehicles, proportionally to the number of generated CAM messages. As a result, this first solution is unacceptable. A much better solution involves the generation of the support message only when the vehicle is detected moving into an area with a different support message content. Basically, the support message generation is triggered only on the transition from an area to the other, with the burden for Response Router and vehicle minimized and the Local Manager optimizing the message generation. Sadly, also this solution is flawed, since if the message is not correctly delivered, the vehicle has no more chance from being updated.

To overcome such an issue, we can assume to have the AMQP message from cars carrying information about the last support message received: if this one refers a message associated to an obsolete list of endpoints, the Local Module generates a new update message and sends it to the vehicle via Response Router. As a trade-off alternative, if the vehicles are not providing such a piece of information, the Local Manager may send the update only if a given amount of time (e.g., 5 seconds) elapsed since the previous update message sent, or if a transition between two areas with different candidate endpoints occurred. Even if some redundancy is present, this is much less impacting.

As mentioned, a support message may contain a prioritized list of endpoints. Indeed, the endpoints are listed with a (i) priority number (the lower the value, the higher the priority), the (ii) network operator name, the (iii)main manager endpoint, and the (iv) message broker endpoint.

Among message broker endpoints under the same network, the vehicle shall try to connect with those having the highest priority and, in case of failed connection, to proceed to the one(s) with the second highest priority and so on. When a new support message arrives, the vehicle receiving it shall check if its connected Message Broker endpoint has the highest priority in the list:

- If the selected endpoint is still among those with the highest priority, the vehicle shall keep it.
- If there is another endpoint with a higher priority under the same operator currently in use, the vehicle must try to connect to that higher priority endpoint.
- If there is a higher priority endpoint, but it is provided by an operator different from the one currently selected by the vehicle, then the next steps depend on the connectivity management capabilities of the vehicle. If the vehicle supports autonomous operator switch, it shall perform
the switch and try to connect to the recommended endpoint. Otherwise, it shall keep its current connection, which is considered suboptimal but still acceptable by the system.

- If no endpoints are supported for the current selected network, it means that the car is outside the coverage of the service for that operator. If it can switch to another supported operator, it shall do it. If the vehicle cannot perform such an operation, it shall contact the Main Manager, to check in depth for alternative endpoints. If even this response is negative, the vehicle is no longer covered by the service. It shall be remarked that the previously selected endpoint could be still reachable, but the service no longer supports it because of its potential service disruptive effects if used in the current vehicle position (e.g., unacceptable latency or connection instability).

The last point depicts a situation that should be tolerated only at the borders of the service-provisioned area and should be avoided elsewhere.

**An example of dynamic Message Broker endpoint notification at cross-border**

Figure 11 shows an example of the dynamic Message Broker endpoint notification in a cross-border scenario. To simplify the reading of the figure, some modules (i.e., S-LDM and Manoeuvring Service) are hidden, but they are assumed to be there and connected to the other modules (as shown in Figure 10).

Similarly to the situation described in Figure 9, two adjacent Operator domains (A and B) are providing the centralized CLC service over two consecutive sections of a road. According to the coverage given by the two operators, the road is divided into subsections that identify the weight (i.e., priority) of each operator over that given subsection. In the example there are two main areas associated exclusively to a given operator (exclusive domain A and B) and two shared areas (A* and B*), where operators can coexist, but with a priority assigned to one or the other according to the “covering” domain. The union of the shared areas in Figure 11 gives the shared area of Figure 9. For each one of these (shared or exclusive) areas, a support message is defined. It is composed of a set of entries structured as follows:

```plaintext
[priority] <network operator ID>, <message broker endpoint>, <main manager endpoint>
```

Each support message is associated with an area, i.e., the same support message is provided by both operators to any car inside the same area.

If we assume a car traveling from left to right, first it will connect to the Main Manager A to retrieve its very first support message according to its position. Since it is inside the exclusive domain A area, the vehicle will receive the support message S1 as response. In it, it is specified that the only accepted operator is A and the reference Message Broker is Message Broker A. Following the instructions in the support message, the car connects to the Message Broker A. This way, the car joins the service and starts sharing its CAM messages with the system. The car messages are also received by the Local Manager module, which refreshes the car with the support messages according to the policy implemented.

When the car enters shared area A*, the local Manager detects the area change and triggers a new support message S2 notification to the car. The new support message informs the car that two Message Brokers are available (each one for a different operator), as well as the one associated with operator A has the higher priority. Since the car is already connected to it, the car keeps its settings and proceeds. It could happen that inside shared area A* the car is forced for some reason to switch to operator B. From the previously received support message S2 it knows that there is a Message Broker available for operator B, thus it tries to join it. Thanks to the support message, the car do not get lost in the change of operator and can fast re-join the service after the operator switch.

The car then enters shared area B*: assuming it is still connected to Operator A, Local Manager at A’s side sends immediately a new support message S3, where the two brokers presented in the previous support message S2 have now switched their priorities. If the car has autonomous operator
switching capabilities, it shall switch to Operator B and connect to Message Broker B. Otherwise, it can keep its current settings, but being aware of leveraging on a suboptimal Message Broker.

Finally, the car enters **exclusive domain B** area: the car is assumed to be already connected to Operator B, or at least capable of switching forcibly to that Operator. Indeed, the support message S4 associated with that area states that only Operator B endpoint(s) are available, thus the car should join Operator B network (if not already inside it) and connect to the first available broker. This means that if a car cannot join network B, it cannot be served by the centralized CLC service.

![Centralized Approach: cross-border with dynamic endpoint notification.](image)

Figure 11 Centralized Approach: cross-border with dynamic endpoint notification.

As explicitly stated, this is just an example of a possible scenario.

### 2.1.2 Decentralized Cooperative and Automated Lane Change Manoeuvre

In the decentralized use case, the Manoeuvre Management is hosted inside each vehicle and is based on information exchanged by means of V2V communication via the network (V2N2V). Through the Uu interface, Collective Perception Messages\(^1\) (CPMs) and Cooperative Awareness Messages (CAMS) allow the mutual awareness and perception share between vehicles.

This section reflects the D2.2 functional analysis with some modifications and updates, including: the role of PC5 for redundancy (10 Hz CAM); updates in the pictures; revised requirement table; WP5 reference in the connected and automated statements; cross-border management of lane-change, similar to the in-lane manoeuvres (left to section 2.2.2).

#### 2.1.2.1 Functional Analysis

Vehicles periodically exchange CAMs, i.e., messages containing information about position and dynamics, and CPMs, reporting non-connected vehicles and detected objects. The Uu interface allows to exchange such information at 20Hz through the GeoService, as shown in Figure 12. The GeoService southbound interface allows for low latency connection. Received CAMs and CPMs in the host vehicle are thus an input to the Automated Driving module similar to the vehicle on-board

\(^1\) The current document reports the final choice of Collective Perception Message, although the pilot activity will continue with the Virtual CAM approach for practical reasons.
Deliverable D2.3 “5G-CARMEN Final System Architecture and Interfaces Specifications”

sensors (i.e., camera, radar work with similar refresh rate). The V2V communication, enabled by the the PC5 interface, is used as redundant source. It is characterized by a rate between 1 and 10Hz, set by the ETSI Decentralized Congestion Control of ad-hoc communication networks. As shown in Figure 12, the GeoService is also used through its northbound interface to interact with the Back Situation Awareness function, in order to disseminate the Estimated Time of Arrival for the sub-use case of lane clearance for emergency vehicle (see Section 2.3).

![Diagram of vehicle communication and services](image)

**Figure 12 Decentralized Cooperative and Automated Lane Change Manoeuvre in the sub-case of lane clearance for emergency vehicle.**

Referring to the scheme reported in Figure 13 and the flow chart of Figure 14,

1. Car A is aware of the surrounding cooperative vehicles (in this case based on received CAMs by car B and C) and evaluates the conditions to change lanes.

2. Car A sends its intentions, by changing the state of the right turn indicator (“off” to “on”) on its CAM transmitted.

3. Car C slows down based on A’s intention.

4. Car B constantly monitors the front situation and share it with A, through a CPM. In this way:
   - A is aware of any “non cooperative” car in front of B (in this case, D);
   - Any emergency situation (e.g., braking of D) is promptly communicated to A (via CPM) so that A can change plan.

5. Car A needs to change lane based on the conditions, set by
   - The free space between B (front) and C (rear);
A safe condition in front of car B.

Both PC5 and Uu connections are required at the same time for redundancy. PC5 is used just for CAM exchange (awareness of connected vehicles with 1 to 10 Hz refresh) while the complete flow chart of Figure 14 is enabled by Uu. Although PC5 is redundant, it is an important baseline: the lack of either Uu/PC5 connection affects the level of automation allowed. WP5 is evaluating the actual L4 Operational Design Domain (ODD) for this use case, as well as the resilience of Automated Driving functions with respect to the connectivity availability.

It should also be noted that both A and C can separately measure the spacing thanks to the same information source, namely the exchanged CAMs of the three vehicles. Therefore, the algorithms, although decentralised, have the same basis and allow the correct sequence (in particular the aforementioned steps 1, 3 and 5 which are primarily based on the spacing, also called “headway”).

---

**Figure 13** Decentralized Lane Change Manoeuvre: CAMs and CPMs are transferred via Uu at 20Hz. PC5 transfers 10Hz CAM. Neighbour vehicles (C, B, D) are stored in the V-LDM.

**Figure 14** Sequence diagram of decentralized lane change manoeuvres.
On-board perception for de-centralized manoeuvres: the V-LDM

The on-board Local Dynamic Map (V-LDM) illustrated in D2.2 considers relevance areas around the host vehicle (the ego-vehicle) and maps the relevant neighbour vehicles around it. These vehicles around can be connected or not connected. The latter are sensed by the host vehicle itself (through its own sensing system) or notified by other vehicles through the CPM messages. All relevant neighbour vehicles are sent as multiple “virtual” objects, to the AD unit.

Functional requirements of Decentralized Cooperative Lane Change use case were reported in the previous WP2 deliverable D2.2 and are still valid.

2.1.2.2 Cross-border management

The decentralized lane change manoeuvre manages cross-border in the same way as the in-lane manoeuvre (see Section 2.2.2).

2.2 Cooperative and automated in-lane manoeuvres

With respect to D2.2, the general architecture has been updated. The usage of the GeoService southbound interface has been pursued for the extended perception, instead of the V-LDM update by the S-LDM. Therefore, the S-LDM is reported in the functional analysis but is not treated in the cross-border management. The latter is specified for the GeoService only, in section 2.2.2.

In Cooperative and Automated in-lane manoeuvres the vehicle performs longitudinal control and keeps the in-lane lateral control (lane centring). Therefore, 5G-CARMEN will address:

- In-lane manoeuvres based on forward detection (cruise control);
- In-lane manoeuvres based on lateral detection (prevented lane change);
- In-lane manoeuvres based on backward detection;
- In-lane manoeuvres based on emergency vehicle approach.

The focus is on unexpected events, as required by L4 driving. An example is scenario b as shown in Figure 15. The vehicle is on the first lane and plans to exit the motorway in moderate-high traffic situation, with vehicles in front obstructing the view. A queue or obstacle on the exit lane would require the driver to take over. Thanks to 5G, the on-board automated system has sufficient data and sufficient time in advance to re-evaluate an original lane change manoeuvres to an in-lane manoeuvre, and thus keep in the L4 Operational Design Domain. The extended perception has been developed to prove the concept of cooperative perception just with forward looking sensors. Therefore, the piloted sub-cases are a., b., and d.
As explained in the decentralized lane-change manoeuvres, 5G allows to exchange cooperative awareness and collective perception messages at the same refresh rate as the vehicle on-board sensors and to have the needed responsiveness to emergency situations, a key aspect for SAE L4 automation.

2.2.1 Functional Analysis

The following picture (Figure 16) highlights the message flow for the Cooperative and automated in-lane manoeuvre use case. For simplicity, we assume that:

- The vehicle ahead (Vehicle B) shares its state and detections;
- The vehicle behind (Vehicle A) receives state, detections and performs L4 manoeuvres².

In a real-word deployment, the architecture, as well as the state sharing, will be symmetric, and the AD control will be performed whenever needed and possible.

1. Vehicle B constantly senses the cars around.

---

² This is reflected in the different modules of the vehicle box. Actually, all vehicles can communicate their state detections. Similarly, the flow of the V-LDM is unidirectional for better readability (detections by A, LDM updates to B) but the LDM potentially gets update from every communicating node (vehicle/infrastructure) and can update all vehicles with a V-LDM
2. Through the on-board ECU (Electronic Control Unit) and the precise positioning, the Cooperative Sensor and State Sharing module converts relative coordinates (with respect to its own reference system) to the absolute reference system.

3. The “Cooperative Sensor and State Sharing module” calls the facility layer to encode messages as Vehicle B own state (CAMs) and detections (CPMs).

4. The “Cooperative Sensor and State Sharing modem” uses three interfaces to dispatch messages: AMQP client “publish”, GeoService interface and PC5.

5. The PC5 modem Rel.14 will exchange CAMs at the highest possible rate at 5.9GHz, as allowed by ETSI Decentralized Congestion Control.

6. The 5G Uu modem will exchange CAMs and CPMs at 20Hz with the GeoService (pilot focus) and the AMQP broker.

7. The GeoService will cast the CAM and CPM messages to the vehicles in the interested area. If Vehicle A is in the area, it receives the message.

8. When Vehicle A intends to exit, two options are provided:
   - Option with S-LDM: partially tested. The S-LDM updates vehicle B with context information via the AMQP publish interface. The trigger for this could be the turn indicator or also a predictive algorithm on the S-LDM. The V-LDM gets updated context by S-LDM to evaluate lane change conditions.
   - Option with V2V (V2N2V): tested and demonstrated in the corridor. The V-LDM of the Vehicle A gets updated from PC5 direct link and GeoService. The turn indicator of vehicle A triggers the on-board evaluation of lane change conditions.

9. The V-LDM filters relevant events based on the received messages, according to the vehicle motion and intentions. It provides a highly dynamic picture of the relevant objects around to the Electronic Control Unit of the vehicle.

10. In the Electronic Control Unit the perception component combines the on-board sensors input (such as radar, camera) with the V-LDM input.

11. The decision component of the vehicle evaluates the best manoeuvres options, considering both the trip plan but also the target of keeping within L4 ODD. The decision component decides for an in-lane manoeuvre (step 4 of Figure 15) instead of the planned lane change.

12. The actuation of the vehicle keeps the cruise control and lane, adjusting following the road geometry and adjusting its speed automatically, according to the surrounding vehicles/obstacles constantly monitored by on-board sensors and 5G communication.
Figure 16 Message flow for Cooperative and automated in-lane manoeuvres and for the decentralized lane change manoeuvres, in country.

Figure 16 shows the message flow for in-lane manoeuvre, whereby:

- Green lines correspond to the V2V solution, the one piloted thoroughly: from 5G modem, through the GeoService to the other vehicle;
- Red lines correspond to the LDM solution, piloted in uplink: data is provided to the AMQP, then to the S-LDM, which pushes context updates when the vehicle is about to perform the manoeuvre.

The following sequence diagrams report both options, highlighting the main messages that trigger the main manoeuvring steps, or the changes in decision/AD functionality. It is worth pointing out that messages are continuously exchanged and potentially every single message can affect manoeuvres. For instance, the constant CAM exchange at 20Hz is used for data fusion and distance keeping, but it is not depicted here.

**V2V approach** (used in the pilot)

The following Figure 17 reports a sequence diagram for the case in V2V. When close to the motorway exit, the Autonomous Vehicle A initially pursues the exit plan (Figure 15). Based on Vehicle B state (CAM) and detections (CPM) Vehicle A becomes aware that the exit lane has a queue. Since the queue (as perceived from B and communicated to A) is almost still, Vehicle A decision module evaluates that taking that exit lane would imply a severe slow down, and a sudden exit of the L4 Operational Design Domain. Therefore, Vehicle A changes its mind (automatically) keeps in-lane in car following situation. Car following is supported by both the radar and the CAM messages as well as from Vehicle A’s own sensors.
Figure 17 Sequence diagram for V2V communication, with focus on the messages triggering the changes of manoeuvre.

**LDM approach** (for future implementation; only the uplink is demonstrated in the pilot).

The sequence diagram of Figure 18 reports the same use case done via LDM update message. Here it is worth pointing out that the S-LDM instance gives periodic updates at a default frequency in a steady scenario (like ordinary cruise keeping in-lane manoeuvre) while it raises its update frequency when the context demands: in this case, a fast update to Vehicle A is triggered by the intention of Vehicle A to exit the motorway.

The drawing of Figure 18 focuses on the key messages: Vehicle A communicates its intention, i.e., changing the S-LDM state, and Vehicle B updates the LDM with its detections. In reality, there is a constant stream of data from all connected vehicles to the S-LDM. Furthermore, the turn indicator trigger symbolizes (and simplifies) a generic triggering condition. In a future perspective, Artificial Intelligence within the S-LDM could trigger the fast update.
2.2.2 Cross-border management

Figure 19 gives the functional view both for cooperative lane change and in-lane manoeuvre (decentralized) in a cross-border scenario, using the GeoService interface.

Figure 18 Sequence diagram for LDM communication. The S-LDM, triggered by vehicle A lane change indication becomes very responsive and gives A fast updates on the surrounding environment.
As general case, a group of vehicles could be performing such manoeuvre while crossing the border. For simplicity, we can consider two cars leading and following (B and A, respectively) at 130km/h and crossing the border (see Figure 20).

![Figure 20 in-lane manoeuvres in a simple case of forward detection.](image)

The goal is that A is constantly in condition to perceive B’s position (CAM) and detections (CPM), 20 times per second. When close to the border, two conditions could happen:

**Cross-border condition (1):** B and A connect to different MEC instances of GeoService. Indeed, a vehicle could start transitioning to the next country road operator while the other could still be attached to the country it is coming from.

**Cross-border condition (2):** when B and A reselect the communication network (always true at the border). In this case outages of communication may happen in both vehicles, potentially at different times.

Both are expected to be transient conditions lasting of less than 10 seconds. As a reference, inter-vehicle distances between A and B go up to 300-400m (GeoService configuration in the pilot is 350m). Although MNO transition does not correspond to geographic border, we can state the following:

- for distant vehicles, the leading vehicle (B) is more likely to change operator first. The following vehicle (A) goes along the path of the leading 10 seconds later, in the 350m case, so indicatively, time duration of mixed MNO/mixed MEC should have this order of magnitude (lower vehicle speeds at such high distance are not relevant).
- for close vehicles (<300m) the mixed scenario of vehicles connecting to different instances will last shorter and it is expected that the communication outage (to be counted for each vehicle) is predominant.

In condition (1) (two instances for two vehicles) CAMs and CPMs are routed between GeoService instances over the MEC-MEC interface. In this case, transition is properly managed as:

- MEC-MEC connection between GeoService instances (which maintains the low latency requirements);
- Local Break Out ensures that each vehicle connects to the service always through the shortest path, independently on the country in which it finds itself.

Condition (2) is more critical. It has indeed two main effects: the absence of high data rate extended perception, and the aging of GPS corrections which are needed for sub-meter accuracy.

In 5G-CARMEN, two solutions at network and application layers, are aimed at minimizing this outage in terms of timing:

- ePLMN + Release with Redirect at the network infrastructure, which ensures the reselection below 3-4 sec. Position accuracy KPI is <1 sec, but 3-4 sec might be sufficient (to be evaluated in WP5).
A software component on-board the vehicle constantly checks the Uu connectivity and the operator change (at radio access level), and quickly triggers the IP level-reconnection (well below 3 sec).

In addition, in order to mitigate the cross-border effect and not jeopardize the user experience, the following countermeasures have been designed:

- The usage of QoS on the automated driving component of the car, to constantly monitor the 5G/C-V2X related KPI, such as positioning accuracy, latency, jitter, 5G connectivity (full UC-related KPI's are provided in D5.1).
- The usage of QoS on the automated driving component (though only partially piloted, due to the NSA deployment) to smoothly move to lower layers of autonomous driving.
- The on-board object tracking, which combines 5G with PC5 and sensor fusion, to keep active the cooperative cruise control (even without extended perception) and spot, e.g., any cut-in manoeuvre by non-connected vehicles.

### 2.3 Lane clearance for emergency vehicles

This sub-case regards a safe lane-change to clear the way for emergency vehicle, keeping a vehicle in the L4 operational design domain. The main factors are a continuous awareness of the emergency vehicle, by utilizing the back-situation awareness feature, across the corridor and exploiting the highly performant communication system established amongst the vehicles. Only by deploying 5G along the corridors these factors can be met. This section presents the various scenarios in this respect.

This scenario is depicted in Figure 21, where the ETA information from BSA service function in the infrastructure triggers the lane change manoeuvre. Lane change is performed in decentralized way, thanks to the cooperation of vehicles A, B, C. The presence of non-connected vehicles (D) (detected by on-board sensors) is shared, so that vehicle A can have a full picture of its surroundings.

![Figure 21](image)

**Figure 21** Lane clearance for emergency vehicle: the host vehicle (A) triggered by the emergency vehicle notification, performs a decentralized lane change.

Lane clearance for emergency vehicles is the same as specified in D2.2 [2]. Here the high-level description is provided, with an updated sequence diagram to highlight the difference between BSA service and GeoService. Furthermore, the cross-border section has been added.

### 2.3.1 Functional Analysis

A cooperative lane change decision is triggered, for instance, when the overtaking lane needs to be cleared because an emergency vehicle (EmV) is approaching from the back. The decision is governed by the Estimated Time of Arrival (ETA) value of the approaching EmV. This functionality is realised
using the redundant Uu/PC5 link between the vehicle and EmV, when the EmV is within the PC5 range, and over the Uu interface (between the vehicle and 5G RAN) via the MEC system to notify vehicles that are outside the Audio and Visual range of the EmV.

![Diagram](image)

**Figure 22 Overview of BSA functional components.**

This use case is illustrated in Figure 22, which provides an overview of the interactions between the various actors involved in realizing this use case in a C-V2X environment, including CCAM components and interfaces.

1. The Emergency Management Authority (EMA) on receiving the emergency information will send an emergency notification message to the MEC system, which is used as a trigger to instantiate the Back-Situation Awareness (BSA) function. It is realized as a virtual application function.
2. The EMA will inform the EmV and provide the BSA service function id (e.g., IP address) of the newly instantiated BSA service function. The EmV will periodically send a CAM notifying its status of Emergency vehicle (EmV Notification message) over the PC5 interface as well as the Uu interface towards the RAN.
3. The RAN will route these EmV Notification messages towards the designated MEC system where the BSA service is instantiated.
4. Based on the route plan and the destination information of the EmV, the instantiated BSA service function shall trigger additional BSA service function instances in other MEC systems, even cross-border.
5. Based on the periodically received CAM messages notifying the status of the EmV, the BSA service function calculates the Estimated Time of Arrival (ETA) with reference to the multiple dissemination areas (DAs), which are geo-sectorized portions of the route-path of the EmV.
6. The periodically derived ETA values are encoded in an EmV Warning Notification messages (DENM), which is then pushed towards the associated RAN BSs via the GeoService.
7. Optionally, the Back Situation Awareness Functionality (BSAF) can leverage latest information from external traffic management system, for example about latest traffic situation in respective DAs, to derive accurate ETA values.
8. The ETA is disseminated to the relevant DAs using DENM messages via a message dissemination service, such as the GeoService.
9. The host vehicle (ego-vehicle/recipient) periodically receives the DENM message valid for the DA where it is driving and extracts the ETA. Both EmV and host vehicle are moving, so the sequence of received ETA will vary in time.
10. When the EmV comes closer (or if the emergency operation/siren starts when the EmV and host vehicle are close to each other), the CAMs of the EmV are used by the host vehicle to calculate the ETA value and the relative distance (<500 meters). They are calculated with a higher precision since both the host vehicle and the EmV speeds are considered.
11. Based on the ETA values and the internal logic, the host vehicle can make appropriate decisions to perform a cooperative and automated lane change manoeuvre and to clear the lane in due time enabling the free passage of the EmV.

The main functional components are:

1. **OBU for EmV**: The on-board unit (OBU) of the EmV is required to collect information from the on-board sensors/monitoring modules like speed, geo-location, route plan etc., and is able to encode them on CAM frames to be broadcasted over the Uu interface towards the cellular network base station (e.g., eNB, gNB) in a periodic fashion, till the EmV arrives at the event destination.

2. **Cellular network infrastructure (4G/5G)**: The RAN component of the cellular network should be able to redirect the EmV Notification messages towards the User Plane Function (UPF), which will forward it to the MEC platform implementing the CCAM platform.

3. **BSA service function container**: The BSA service function instance instantiated and managed in the CCAM system platform is able to process the periodically received EmV Notification messages to derive the ETA, and encode it in the EmV Warning Notification message, which is then pushed towards the vehicles by means of GeoService. BSAF instance shall also be able to forward the received EmV Notification messages towards other BSA service function instances hosted in other MEC platform across borders.

4. **MEC System**: This MEC system deploys the CCAM platform that manages the BSAF container and interconnects with the cellular infrastructure. The BSA service function container may also leverage the services offered by the AMQP Broker and the GeoService entities.

5. **Vehicle’s OBU**: The OBU in the vehicles shall be able to receive and process the EmV Warning messages forwarded by the BSAF function and received via the cellular base stations and display the ETA on the dashboard.

The sequence diagram is reported in Figure 23. When the EmV is still far away from the host vehicle A, the latter vehicle is aware of the incoming EmV from the estimated time to arrival (ETA) by the BSA service (through the GeoService).
2.3.2 Cross-border management

A cross-border case can happen when, for instance, an accident occurs on one country (Country B in Figure 24) close to the border but the closest emergency vehicle comes from the other country (Country A). Therefore, the goal is to ensure that the stepwise process as described in 2.3.1 is carried out with continuity. In this way, all vehicles along the path can be informed and it is possible to ensure right of way to the emergency Vehicle while it is travelling from Country A towards Country B, until it reaches the rescue destination.
As Figure 24 shows, cross-border management of BSA is performed through the Orchestrated Edge platform.

The first cross-border phase is managed while the emergency vehicle is still in country A. In this phase:

- The closer vehicles are informed through the services in the same MEC. The BSA of instance in country A receives information from the EmV, and dispatches ETA to the vehicles via GeoService (e.g., dissemination Area 1 and 2).
- The farther vehicles, which already past the border (Figure 25), should be informed too. This is done via the additional BSA service function instance in the other MEC of country B. This instance receives information about the approaching EmV by the BSA instance of country A, and can in turn generate ETA values for the interested dissemination areas (Dissemination area 3).
The second phase is the transition of EmV between borders. As soon as the EmV changes country, it should reach the BSA instance of the visited country. The IP addresses of the BSA instances shall be given by the EMA to the EmV, while the actual change of IP is performed thanks to a dedicated software component in the on-board system (see also 2.2.2).

Figure 25 Back Situation Awareness cross-border (scenario view).
3 Final 5G-CARMEN system architecture

3.1 Overall architecture description

Below, Figure 26 reports the overall architecture mapped to the field deployment as it was already presented in Deliverable D2.2 [2]. The final architecture confirms the structure that has been detailed in the preliminary study of D2.2. As described in more detail in that deliverable, the architecture has been divided into several layers: first, the street layer with vehicles and RSUs connected directly via PC5 and to the other components via the Uu interface. Second, the access layer represented by the eNBs/gNBs from the local MNOs, as well as the existing Roadside infrastructure connections to the respective Backends. Third, the Edge Clouds with MEC platforms deployed in each country which are integrated into the Access/Core networks of the MNOs. This layer is further structured over 2 sub-layers: the MEC service orchestration components that orchestrates functions within the MEC and between different MECs and CCAM services that group application-level services for specific use cases, and value-added services enabling communication at application level, (e.g., sharing of ITS messages). The last layer consists of various Cloud components, including existing systems such as OEM backends and dedicated Service supporting the use cases.

3.2 Reference points and interfaces

In the previous picture (Figure 26), three main types of communication can be identified:

- Interfaces between vehicles and other elements in the overall infrastructure, including other vehicles, Mobile Networks and Road Side Units (RSUs).
- CCAM platform orchestration and virtualisation related ETSI MEC, NFV and MANO reference points (not exhaustive, and adding the new local orchestrator reference point).
• Interfaces towards services hosted in the Cloud.

The next sub-sections will describe these three families of interfaces, as well as some additional use case-specific interfaces that have been defined between functional elements inside the MECs.

### 3.2.1 Interfaces between vehicles and infrastructure

This section presents a detailed description of the interfaces which will be used between vehicles and infrastructure in each of the project’s use cases.

#### 3.2.1.1 Device-Device interface: Uses PC5 between vehicles

C-V2X, as specified at the radio access layer by 3GPP in Rel.14 and 15 and by ETSI TC ITS at network through application layers, supports various communication links and interfaces including:

- PC5 interface for V2V, V2I, and V2P communication;
- Uu interface for V2N communication.

The PC5 interface, also called “Sidelink”, supports direct communication between devices in the vicinity. Being a direct link, it is ideally suited to support use cases requiring low latency like basic safety and a host of Day 1 and Day 2 Cooperative Intelligent Transport Systems (C-ITS) services.

The PC5 interface is designed for vehicular communications in 5.9 GHz ITS band with 10 or 20 MHz bandwidth (in Europe, 10 MHz is used). It effectively addresses the challenges of high relative speed (up to 500 km/h), high device density and low latency to transfer ITS messages. A long list of features like physical layer modifications improving detection and high-speed tolerance, HARQ transmission, and receive diversity ensures the above requirements are fulfilled.

The PC5 direct link Mode 4 is the commercially implemented mode in Europe, US, and China. It does not depend on the utilization or signal quality of cellular network and ensures continuity of the use cases when there is no network coverage or even across political and network borders. The 5G-CARMEN partners integrated PC5 Mode 4 in their vehicles for direct short-range communication.

Vehicles share their critical state and kinematics information in broadcast manner as part of the periodic CAMs (Cooperative Awareness Message) and ad-hoc, event triggered messages in DENM (Decentralized Environment Notification Message). The neighbouring ITS station receiving these messages processes and filters them, and triggers use cases according to their relevance to the driver or autonomous driving system.

The 5G-CARMEN use cases leverage the PC5 connectivity utilizing it as a complementary link to the Uu V2N connectivity.

#### 3.2.1.2 Uu interface: Interface between Device and Mobile Network Base Station which can be 4G/5G

The baseline in 5G-Carmen is the 5G non-standalone (NSA) network as specified in 3GPP Rel.15, where both existing 4G LTE base stations (eNodeB) and 5G New Radio (NR) base stations (gNodeB) are attached to a 4G core network (EPC). The Uu interface is the interface between the user equipment (UE) and the radio access nodes. In a 5G NSA setting, the Uu interface is based on the principle of Multi Radio Access Technology Dual Connectivity (MR-DC), i.e., EUTRA (LTE) and New Radio (NR) combined for Dual Connectivity, also referred to as ENDC (see Figure 27).
With ENDC, the Uu interface is split into the control plane (CP), served exclusively via E-UTRA (LTE) and the user plane, being either served only via 5G NR, or, as split bearer, both via LTE and NR, the latter also being referred to as deployment option 3.x, as it is the case with the three networks involved in serving 5G-Carmen.

The control plane architecture for EN-DC (as shown in 3GPP TS 37.340 version 15.3.0 Release 15).

The user plane radio protocol architecture for MCG, SCG and split bearers from a UE perspective in EN-DC (as shown in 3GPP TS 37.340 version 15.3.0 Release 15).

Note: MCG is the Master Cell Group, representing the E-UTRA (LTE)-only user plane, while SCG stands for Secondary Cell Group, representing the NR-only user plane, while split bearer uses both E-UTRA and NR bearers.
This Uu setting enables the flexible inclusion of 5G NR-based bearers in areas with 5G coverage, as it will be the case for the two cross-border sections, while maintaining seamless connectivity in 4G-only areas where no 5G coverage is available yet.

3.2.1.3 RSU-Vehicle interface
The RSUs distributed along the motorway axis manage the transmission and reception of the Infrastructure to Vehicle (I2V) and Vehicle to Infrastructure (V2I) messages on the DSRC ETSI ITS-G5/802.11p wireless network, which however is not used in the 5G-CARMEN project.
It will also include the transmission via PC5. However, the hardware is not yet available at the reference retailer and will become so indicatively during the first half of 2022.

3.2.2 Orchestrated Edge Platform for CCAM services – components and interfaces

3.2.2.1 Functional Architecture and Key Enablers
This section summarizes the key results from the design, development, and deployment phases [4] of Work Package 4 (WP4) and highlights the main enabler and associated features to support edge operation and orchestration of CCAM services. The summary is structured into three sections:

- Functional architecture with main functional blocks and associated internal and external interfaces;
- Main reference points that apply between MNOs’ edges cross-border and how they can be used;
- Feature description – Main features which can support different aspects of cross-border CCAM service continuity.

Features are summarized in the context of the previously analysed use cases and requirements, which have been evaluated and assessed in the final phase of WP4 for delivery to the final trial phases of the project. These include the following main features:

- Fast on-boarding and lifecycle management of CCAM services at mobile operator edges.
- Instantiation of CCAM service components in defined slices on the orchestrated edge infrastructure, ensuring dedicated and isolated resources as required for the delivery of agreed service levels. Orchestrated edges are herewith prepared to integrate well into a future end-to-end as well as edge-to-edge slicing enabled 5G ecosystem.
- Multi-domain CCAM service lifecycle management and service continuity, leveraging three layers of interfaces between orchestrated edges of different MNOs: Federation interfaces between MNOs’ top-level NFV Service Orchestrator (NFV-SO), localized federation interfaces between MNOs’ edge local orchestrators (NFV-LO), and data plane inter-connect between orchestrated edges and CCAM services.
- Policy-based Local Orchestration and federation for balancing orchestration load and accelerating orchestration decisions, following and extending the ETSI MEC and ETSI NFV standard.
- Inter-edge connectivity and programmatic data plane traffic steering for smoother service relocation and continuity.
- Authenticated access to edge CCAM services and fast re-authentication during cross-border mobility.
- **Protected operation of CCAM services** at orchestrated edges and vehicles against intrusion attacks, in alignment with rev. 3 of UN Regulation No. 155 and complementing 5G security as a step towards end-to-end security in an automotive 5G ecosystem.

- **Smart edges -- Integral support of data analytics and machine learning** at edge application- and edge value-added-services (VAS) level to boost orchestration decisions and improve the experienced CCAM service quality during cross-border mobility.

Figure 28 depicts the functional architecture and key reference points of the secure orchestrated and federated edges:

![Functional architecture and key references points for Orchestrated 5G edges platform.](image)

### 3.2.2.2 Key components of the orchestrated edge platform

**The key components of the depicted architecture**

- **NFV-SO**: Represents the top-level orchestrator of the multi-tier orchestration system design of the orchestrated platform for CCAM. The operational scope of this orchestrator includes the management of the entire virtualized infrastructure of an operator’s domain. It is responsible for the management and orchestration of application services from multiple tenants. It has the additional feature of enabling federation with the NFV-SOs of other administrative domains. It maintains a global repository of the application packages and software images received from the OSS upon an on-boarding request.

- **NFV-LO/MEAO**: The combination of the NFV-LO and the MEAO forms the local edge orchestrator, which represents the second tier of the multi-tier orchestration system design of the orchestrated platform for CCAM. The operational scope of this orchestrator includes the designated clusters of MEC sites. There is a 1:N relationship between the NFV-SO and the Local Orchestrator. The pair of the local orchestrator decouples the management operations,
such that the MEAO is responsible for the lifecycle management (LCM) CCAM services hosted on the MEC hosts, while the NFV-LO is responsible for the generic management of the Virtual Network Functions (VNFs) hosted on the Network Function Virtualization Infrastructure (NFVI).

- **CCAM Services**: Any CCAM service function or micro-service instance is then instantiated on the MEC Platform. It can be distinguished between ad-hoc (on-demand) services, such as situation-aware or dynamic mission-critical applications, and persistent services, such as the project’s Manoeuvering Service. Main on-demand service for deployment in the pilot is the BSA service.

- **MEC Value-Added Services (VAS)**: VAS services are MEC services, which can be leveraged by any CCAM Service. This includes services per the ETSI ISG MEC, such as the Radio Network Information Service (RNIS) and the Location Service, as well as the 5G-CARMEN project’s AMQP broker and the GeoService.

- **MEC PF**: It represents a collection of essential functionalities required to run MEC applications on top of a virtualization infrastructure, while these applications can deliver and consume various services.

- **Edge Controller**: It combines functions per the ETSI ISG MEC, such as the MEC platform manager and VNF management, with additional functions, such as VNF connectivity and service mesh control, data plane control and 5G system coupling enablers. The Edge Controller represents an abstraction layer between virtualization technology specific infrastructure management and edge level orchestration, hence supports the enforcement of network slicing and associated policies at edge system level. The project extends the Edge Controller with an Open API towards orchestration functions.

- **VIM**: Virtual Infrastructure Manager for the resource management and providing connectivity between the various VNFs of a network service. It interacts with the Edge Controller and NFVI for example for LCM enforcement.

- **NFVI**: The Network Function Virtualization Infrastructure provides the necessary resources (compute, storage, network) for the MEC application services running on top of the MEC platform.

- The **5G Data Plane, 5G Control Plane and Transport Network** blocks represent the mobile core network abstractions, whereas the Programmable Data Plane represents a data plane overlay of the 5G system’s N6 reference point in support of policy routing and traffic steering.

- **IMM** comprises the functional components for **Identity Management and authenticated access to CCAM edge services**. It is used to authenticate edge components such as users and services to the services provided by 5G-CARMEN. It is a “passwordless” authentication solution that makes use of Social Logins for users and certificates stored in secure elements to reduce the attack surface of the system.

- **IDCS** for **Edge Intrusion Detection and Classification**. We developed two types of Intrusion Detection Systems (IDSs). One IDS is an Intrusion Detection and Classification Module that is developed to protect the orchestrated edge and deployed on the MEC infrastructure. The second IDS is developed to protect vehicles from cyber-attacks. In the context of the project, the two IDSs have been integrated to provide a secure solution that aims to mitigate cyber-attacks in vehicle fleets.

### 3.2.2.3 Reference points

The following lists the management plane reference points
1. The **Or-Or reference point** is specified between the top-level orchestrators, NFV-SOs, that may belong to different operators' administrative domains. According to D4.1 [4], this reference point is *primarily* designed to:
   a. Enable the federation between administrative domains belonging to different organizations, where the organizations may transcend international borders.
   b. Enable the interaction between multiple decentralized orchestration functions. This implies North-South and East-West interactions in case of orchestrators' hierarchy.
   c. Support the lifecycle management (LCM) of services in a multi-domain environment.

The Or-Or reference point design is based on the by ETSI GS NFV-IFA030 specification [10], which describes 7 management interfaces and operations on them respectively. Those interfaces are *Policy management interface*, *Fault management interface*, *Performance management interface*, *NS Lifecycle Operation Granting interface*, *NS Lifecycle Management interface*, *NSD management interface*, and *NS instance Usage notification interface*.

It should be noted that the implementation scope of the Or-Or reference point is limited within the context of the 5G-CARMEN project and focuses on the trial requirements of the cross-border use cases. In addition, due to the proposed two-tier orchestration layer, the operation features of the standard Or-Or reference point is also extended. The 5G-CARMEN project extends the ETSI NFV Or-Or reference point in order to support federation and delegation of orchestration tasks using management level agreements.

2. The **Lo-Lo reference point** is instantiated between the NFV-LOs of different MEC sites that may belong to the same and/or different operator's administrative domains. In terms of interfaces and operations, it is similar to Or-Or reference point but the main rational for introducing this reference point is to enable direct and low-latency management of multi-domain and multi-site services, bypassing the top level NFV-SO orchestrators. The interfaces exposed between the NFV-LOs of the peering MEC sites will be determined by the Management Level Agreement (MLA) negotiated between the top level NFV-SOs, thereby enabling the federating domains to control the level of the management autonomy that is to be delegated to the peering NFV-LOs. The concept and details of the MLA is provided in D4.1 [4]. The Lo-Lo reference point is also used to ensure synchronization between the NS/VNFM package repositories in the peering domains.

3. The **Mv1 reference point** is instantiated between each NFV-LO and its pairing MEAO from the same edge domain, and it is related to Os-ma-nfvo reference point defined by ETSI NFV. As per D4.1 [4], D4.2 [13], and ETSI NFV [15], this reference point allows MEAO to invoke operations towards the NFV-LO to manage and orchestrate CCAM service deployments. Thus, this reference point enables interaction between these two edge-level orchestrators, which further allows MEAO to enhance LCM operations performed by NFV-LO by making decisions on service placement, scaling, relocation, and termination.

4. The **Mv1’ reference point** is instantiated between each NFV-LO and its 'parent orchestrator', i.e., the NFV-SO, and its scope is limited within the same operator's domain. According to D4.1 [4], this reference point is *primarily* designed to:
   a. Enable the interaction between multiple decentralized orchestration functions. This implies North-South and East-West interactions in case of orchestrators' hierarchy.
   b. Support the provisioning of management autonomy to lower orchestration domains of an orchestration hierarchy.
   c. Shall support the lifecycle management (LCM) of services in a multi-domain environment.

The Mv1’ reference point is unique to the orchestrated platform for CCAM, and this is the interface over which the MLA parameters are negotiated to determine the scope of the management autonomy that the NFV-SO can delegate to the NFV-LO. Moreover, this reference point also exposes interfaces that will enable the NFV-SO to not only monitor the performance and fault events of the resources within the NFV-LO domain but will also monitor
the compliance of the MLA agreement. It should be noted that the NFV-SO has full administrative access of the entire management domain and can support and/or overrule the management decisions of the NFV-LO. The Mv1’s reference point is also used for the Network Service (NS)/VNF package management between the global package repositories and the local package repositories.

Beyond the specified management reference points, the orchestrated edges platform supports reference points on the data plane to connect edge services for collaboration (sharing of application states), to perform context transfer (during edge service relocation when a vehicle performs cross-border movements and a change in the service MNO), and to perform data plane relay, which connects a vehicle to a handover target MNO’s edge service before the actually cross-border movement and handover to a target MNO happens. This helps to reduce the service interruption and associated data plane packet loss. Documentation and validation of these mechanisms can be found in [13] and [17] respectively.

### 3.2.3 Interfaces in the MEC platform

#### 3.2.3.1 Interfaces between functional elements in the MECs to handle cooperative manoeuvring use cases

**Message broker – S-LDM:**

The S-LDM receives CAM messages published by vehicles through the AMQP broker. In order to receive only those messages sent by the vehicles currently in the geographical area of its interest, the S-LDM needs to subscribe to the broker. To this aim, the S-LDM uses the topic “topic://5gcarmen”, being the same topic that is set in the OBU’s vehicles. Thus, since it is quite generic, to correctly receive from the broker only the messages relevant to the S-LDM service area, it is necessary to use a further filter on the Quadkeys. However, this filter is not constant for all S-LDM and is calculated at the instantiation of the S-LDM based on a range of latitudes and longitudes specified by the S-LDM configuration. This range is of the type "<min lat>: <min lon> - <max lat>: <max lon>". In practice, the S-LDM calculates the minimum set of Quadkeys needed to cover the smallest area greater than or equal to that specified in its configuration file. Then, since this process is computationally demanding, the obtained Quadkeys are stored in a cache file, so, in case of restart of the S-LDM with the same settings, it can simply reload the file from the cache, instead of recalculating all the Quadkeys to generate the filter.

For example, a fairly large area corresponding to the range of coordinates "46.0201280: 11.0279790-47.307088: 11.686929", is translated into a filter which has a form like:

quadkeys LIKE '1202302000%' OR quadkeys LIKE '1202302022%' OR ...

In Figure 29 it is possible to see the geographical area corresponding to the Italy-Austria border that is specified by the above range of coordinates.
The broker and Quadkeys settings are all set via command line options (for example, with "-A 46.0201280:11.0279790-47.307088:11.686929 --broker-url 213.162.90.227:5672 --broker-queue topic://5gcarmen" for a subscription to the Austrian broker), which are passed to the containerized S-LDM via Docker environment variables, in turn set within the .yaml files for the deployment on the edge platform, made available during the on-boarding phase.

The environment variables supported by the S-LDM container are the following:

- `ENV SLDM_INTERNAL_AREA`: area covered by the S-LDM.
- `ENV SLDM_EXTERNAL_AREA_LAT_FACTOR`: extension factor to define the external area covered by the S-LDM (extension of the Latitude in degrees).
- `ENV SLDM_EXTERNAL_AREA_LON_FACTOR`: extension factor to define the external area covered by the S-LDM (extension of the Longitude in degrees).
- `ENV BROKER_URL`: URL/IP of the main broker.
- `ENV AMQP_TOPIC`: topic.
- `ENV MS_REST_ADDRESS`: address or IP of the REST server of the Manoeuvring Service or other services to which to send the data.
- `ENV MS_REST_PORT`: port of the REST server of the Manoeuvring Service or other services to which to send data.
- `ENV VEHVIZ_UDP_ADDRESS, ENV VEHVIZ_UDP_PORT`: advanced options to manage the communication between the main process of the S-LDM and the web-based viewer (normally it is not necessary to set them).
- `ENV VEHVIZ_WEB_PORT`: port used to access the web-based view.
- `SLDM OTHER_OPTIONS`: any other option (for example username and password for the AMQP broker, frequency of sending data to other services via REST, ...).
S-LDM – Manoeuvring Service

The interface towards the Manoeuvring Service is based on a REST API. In more detail, upon the occurrence of a triggering condition, the S-LDM will push the context, formatted as JSON data, to the Manoeuvring Service through a HTTP POST message. Since both sender and recipient will be hosted in the same MEC cluster, the latency associated to this transaction will be quite low. The REST endpoint of the Manoeuvring Service will be specified in the configuration of each S-LDM, as well as the configuration parameters to connect to the AMQP broker. A possible structure for the POST message to be sent to the Manoeuvring Service is reported in Figure 30. The involved vehicles (i.e., the context) are provided as a vector, in which the reference vehicle is differentiated from the other ones through the field “relative_distance_to_reference_m”. In fact, for the reference vehicle, this value is set to 0, whereas for the others it reports the distance from it. In addition, for each vehicle it is specified the absolute position through latitude and longitude, as well as the “station ID”, which will be used by the Response Router to generate a proper AMQP unicast downlink message towards each of them. Also, the “turnindicator” allows identifying the vehicle that signalled the need to turn, as in the example below.

```
POST <URL> HTTP/1.1
Host: CLIC-MNA
Content-Type: application/json
Accept: application/json

{
   "error": "OK",
   "event": "CLIC",
   "eventID": 0,
   "generation_timestamp": 16293602560102296,
   "reference_VEHICLE_ID": 4,
   "vehicles": [
      {
         "CM_timestamp": 30743,
         "GN_timestamp": 3504379590,
         "FH_points_lat": [
            "46.128769",
            "46.125852",
            "46.126365",
            "46.128195",
            "46.125019",
            "46.124837",
            "46.124653",
            "46.124469",
            "46.124205",
            "46.124101",
            "46.123917",
            "46.123735",
            "46.123569"
         ],
         "FH_points_lon": [
            "11.083418",
            "11.083525",
            "11.083637",
            "11.083722",
            "11.083821",
            "11.083903",
            "11.083996",
            "11.084089",
            "11.084181",
            "11.084274",
            "11.084367",
            "11.084459",
            "11.084552"
         ],
         "car_len_mm": 5000,
         "car_width_mm": 1800,
         "heading": 335.1,
         "lat": 46.1225103,
         "lon": 11.0837543,
         "relative_distance_to_reference_m": 0,
         "sourceCountry": "IT",
         "speed_m": 30.1,
         "stationID": 5,
         "time_since_generation_timestamp": 404223,
         "turnIndicator": "rightTurnSignalOn"
      }
   ],
   "time_since_generation_timestamp": 353501,
   "turnIndicator": "off"
}
```

Figure 30 Portion of the proposed JSON message format sent by the S-LDM towards the Manoeuvring Service.
**Manoeuvring Service – Response Router:**

Response Router exposes a single API REST interface to collect all the messages it is requested to forward to the car clients, via AMQP Broker.

Any entity interested in sending messages to the cars has to perform a HTTP POST request on the following URL:

<response-router-IP>:<response-router-port>/api/v1/to_cars

The body of the request must fit the structure shown in Code 1.

```json
{
  "messages": [
    <rr_message_entry>,
    <rr_message_entry>,
    ...
  ]
}
```

**Code 1: body of HTTP POST request on Response Router.**

The body of the message is just made by a "messages" entry associated with a list of items, identified as Response Router Message Entry, structured as shown in Code 2.

```json
{
  "car_ID": <string:carID>,
  "position": <string:quadtree>,
  "message": <string:message>
}
```

**Code 2: Response Router Message Entry (list element in HTTP request body).**

The "car_ID" key is associated with the value used by the car client to identify univocally itself in the CAM message it has forwarded to the AMQP Broker. That value should also have been used by the car as selector when the car client subscribed to the "toCARS" topic on the AMQP Broker. The Response Router will use the "car_ID" value in the header of the associated AMQP message to let the selector trigger at the AMQP side. Thus, the consistency in the usage of this value is pivotal for the proper functioning of the selector feature, i.e., the correct dispatch of the message to the destination car.

The (optional) "position" key contains the quadtree code associated with the last notified position of the car at the moment of the context transmission from S-LDM to Manoeuvring Service.

The "message" key is associated with the message to be delivered to the destination car. The Response Router will use this field value as the body of the AMQP message to be sent down to the broker.

The "message" value is opaque to the Response Router: it never tries to read it to derive some information.

Thus, in the end, the Response Router just copies the message value as it is into the AMQP message, uses the other information in the Response Router Message Entry to configure the header of that AMQP message and then sends it to the AMQP Broker.

An example of HTTP POST request body is provided in Code 3.

```json
{
  "messages": [
    {
      "car_ID": "ITAA123AA",
      "position": "123456789123456789",
      ...
    ]
  ]
}
```

**Code 3: Example of HTTP POST request body.**
In the specific case of the request sent by the Manoeuvring Service, the “messages” list will contain a list of several Recommendation Messages, potentially one for each car involved in the context shared by S-LDM with the Manoeuvring Service.

**Response Router – Message Broker:**

At initialization time, the Response Router registers as a producer on the topic “toCARS” of the AMQP Broker, the same topic cars subscribe for messages from the service infrastructure.

As said in the previous section, every time the Response Router receives an HTTP POST Request, it processes the request body and for each Response Router Entry Message it creates an AMQP message having the value of “message” entry as body and the value of “car_ID” as a header parameter of the same message. Then it posts the message into the AMQP Broker on topic “toCARS”, letting the broker to deliver the message to the car subscribed with the selector set to the same value of “car_ID”.

**Additional Cross-Border Interfaces**

**Vehicle – Main Manager**

As explained in Section 2.1.1.2, the Main Manager exposes a REST interface that provides to each requesting car client with the Support Message associated with the car position.

A Support Message is a JSON document structured as shown in Code 4.

```json
{
  "id": <string>,
  "endpoints": [
    <ep_entry>,
    <ep_entry>,
    ...
  ]
}
```

**Code 4: Support Message structure.**

Below, a brief description of each key in Code 4:

- **id**: unique identifier of the Support Message. It is used internally in Main Manager and Local Manager to rapidly refer/compare messages.
- **endpoints**: list of Endpoint Entries.

An Endpoint Entry (ep_entry) is structured as described in Code 5.

```json
{
  "priority": <integer>,
  "netop": <string>,
  "mmep": <ip:port>,
}
```

**Code 5: Example of body of HTTP POST request on Response Router.**
Below, we report a brief description of each key of the Support Message Endpoint Entry:

- **priority**: it allows to rank the several endpoints in the list according to their preferred usage. It is a non-negative integer: the lower the value, the higher the priority. We propose some policies related to priority management later in this description.
- **netop**: the network operator associated with the endpoint.
- **mmep**: Main Manager endpoint. It identifies the Main Manager endpoint to connect with inside the netop network.
- **amqpep**: AMQP Broker endpoint supporting centralized CLC in the netop network and assigned priority.

An example of Support Message is shown in Code 6.

```json
{
  "id": "sm2",
  "endpoints": [
    {
      "priority": "10",
      "netop": "DTAG",
      "mmep": "80.159.227.46:4000",
      "amqpep": "80.159.227.2:5672"
    },
    {
      "priority": "20",
      "netop": "MTA",
      "mmep": "188.125.17.78:4000",
      "amqpep": "213.162.90.227:5672"
    }
  ]
}
```

**Code 6: Support Message Example.**

Each car client may contact the Main Manager with a GET HTTP Request on the following URL:

```
<main-manager-IP>:<main-manager-port>/api/v1/sm/<quadtree>
```

with the final part of the URL being the quadtree code identifying car position. The Main Manager answers with the Support Message associated with that quadtree code.

**Main-Manager – Local Manager**

The Main Manager configures the Local Managers associated with it by providing for each of them a list of Support Messages, each of the latter associated with a collection of quadtree codes identifying the area where that support message is valid.

The structure of the Local Manager Configuration Message is shown in Code 7.

```json
{
  id: <string>,
  ref_amqpep: <ip:port>,
  local_config:
  
  <lm_area_entry>,
  <lm_area_entry>,
  ...
}
```

**Code 7: Local Manager Configuration Message.**
The keys in this message have the following meaning:

- **id**: unique identifier of the Local Manager. It is used internally in the Main Manager, to refer to the Local Manager.
- **ref_amqpep**: the endpoint of the AMQP Broker at which the Local Manager shall connect to. In this way it can retrieve the AMQP wrapped CAM messages it uses in its analysis.
- **local_config**: the list of Local Manager Area Entries, which are the data structures storing the information of each area covered by the Local Manager and the associated Support message.

The structure of the Local Manager Area Entry is shown in Code 8.

```json
{
  qtcode_list:[
    <qtcode>,
    <qtcode>,
    ...
  ],
  message: <support_message>
}
```

**Code 8: Local Manager Area Entry**

The keys in this message have the following meaning:

- **qtcode_list**: a list of all the quadtree codes identifying the area.
- **message**: the Support Message associated with the area (see Code 4)

The Main Manager and each Local Manager interacts through a dedicated REST API interface, which allows the Main Manager to send updates (via POST/PUT HTTP requests) to the Local Manager and the Local Manager to require its configuration again (via GET HTTP requests), if for some reason it has been lost (e.g., the Local Manager is restarted).

**Local Manager – Message Broker**

As described in section 2.1.1.2 (Local Manager subsection), the Local Manager is subscribed to the AMQP Broker receiving all the CAMS forwarded by connected cars. It subscribes the same topic “fromCARS” which the S-LDM subscribes for the CAM messages and on which each connected car registers as a producer (injecting its CAM messages on it). The CAM messages are received as body of an AMQP broker message.

The header of every AMQP message received is parsed, looking for the parameters “carID” and “position”, the former being the identifier associated with the car which generated the message, the latter the quadtree code associated with the car position. These two parameters are used in identifying the Support Message associated to the car: according to the previously transmitted message, the Local Manager decides whether sending that Support Message to the Response Router or not.

**Local Manager – Response Router**

When the Local Manager detects a situation where a Support Message has to be sent down to a car, it uses the same REST API described in the previous section dedicated to Manoeuvring Service – Response Router and structures the body of the message as described in Code 1.

In this specific case, the “messages” list will contain only one single Response Router Message Entry (see Code 2), configured to have the Support Message stored in the “message” entry and the
destination car identifier (recovered from the header of the received AMQP message from the broker) stored as the “car_ID” value.

3.2.3.2 Radio Network Information Service (RNIS) interface

The Radio Network Information Service (RNIS) is a service that runs in the MEC and provides possibility to MEC applications to extract radio channel information and execute RAN control per vehicle/user equipment (UE).

RAN Control includes possibility to command from the MEC applications to RAN to:

- block handover operation for short time to avoid service interruption during time-critical manoeuvres;
- reduce scheduling latency of specific UEs (vehicles) during time-critical manoeuvres;
- improve robustness against transmission errors (e.g., through lower rate channel coding and lower-order modulation) during time-critical manoeuvres.

As an example, if a MEC application detects a risk that the radio conditions for some vehicles could result into service interruption (e.g., predicted handover) or unacceptable high delay (e.g., due to a high load in the cell or a high number of retransmissions), it may instruct the RAN to block the handover operation, request for high-priority handling and higher robustness against transmission errors till the end of the manoeuvre. By using the RNIS, MEC applications could postpone handover in some situations or improve handover robustness. This is essential for cross-border handovers enabling continuous autonomous driving experiences cross countries. Also, RNIS can inform MEC applications about the loss of 5G radio coverage, and MEC applications can adjust gracefully to the 5G link loss.

The general software architecture is presented in Figure 31.

It includes the following interfaces:

- **RNIS API**: This is an interface to RNIS from MEC applications. RNIS API is also suggested by ETSI. In 5G CARMEN, we selected and implemented a subset of RNIS API useful for the project use cases and extended functionalities ETSI’s RNIS API to include features for RAN control by MEC applications.
**RNIS Abstraction Layer API:** The role of an abstraction layer is to encapsulate external layer API and map them to the equivalent interface defined by RAN vendors (interface of vendor-specific RAN library). The RNIS Abstraction Layer API is an API to interact with vendor-specific RNIS library over common frontend. It does not change when switching from one vendor to another.

**Vendor-specific RNIS library API:** This is an interface that uses the RNIS abstraction layer to interact with vendor-specific RNIS library.

**Vendor-specific EPC’s API:** This interface is used to interact with 5G Core (e.g., get the IP address of a base station and Tunnel Endpoint Identifier of the connected UEs). This makes possible to find out the gNodeB that the vehicle with a given IP address is connected to.

**Vendor-specific base station’s API:** This interface is used to interact with the base station and get RAN information for specific UEs.

Typical interface actions enabled by RNIS API are listed as follows:

- **Subscribing to the RNIS information**
  MEC applications can request RNIS service to provide RAN information for specific vehicles. To receive notifications on selected RNIS events, the application sends a subscription to certain specific RAN information. The subscription information request includes an IP address of the vehicle, a KPI subscription type (CQI, RRC measurements, handover status change), a reporting configuration. The reporting configuration supports 3 modes: (i) periodically, (ii) one time only as data is available, or (iii) on a value update.

  After receiving the subscription request, the RNIS contacts the core network (via Vendor-specific EPC’s API), identifies base stations serving UEs (the vehicles) and UE identifiers within base stations, and configures base stations (via vendor-specific base station’s API) to report subscribed RNIS information about these UEs.

  In the case of 5G NSA, the additional useful functionality is to make aware MEC applications about the loss of 5G radio coverage.

- **Unsubscribing to the RNIS information**
  MEC applications can unsubscribe the RNIS information for specific vehicles (e.g., if no further actions are planned with these vehicles).

- **Receiving RNIS information**
  Base stations report subscribed RNIS information (such as CQI, RRC measurements, handover status change) to the RNIS that in its turn passes this information to the MEC applications.

- **Sending a request for RAN control**
  MEC applications evaluate received RAN information and the API allows MEC application to send a request to the RAN to block the handover operation, request for high-priority handling and higher robustness against transmission errors till the end of manoeuvring for the involved vehicles. This assuming that there is sufficient overlap between cells to allow postponing the handover. If the handover cannot be postponed, it is executed with increased robustness for UEs involved in current manoeuvring. After completion of the manoeuvre, the handover blocking and high-priority handling and higher robustness for UEs involved in the past manoeuvre is no longer needed, so the MEC application unblocks the handover and resets the scheduling priority of vehicles to the default settings.

The RNIS component is implemented in WP3 and its integration, demonstration and evaluation in the lab environment is done in WP5.
3.2.4 Cloud backend interfaces

3.2.4.1 Interface with road-side sensors and Cloud Analytics platform

The basic functioning of the infrastructure requires the information concerning potentially dangerous situations on the motorway axis to be detected by sensors, cameras and reports from professional operators responsible for traffic control. They are then validated by the Traffic Control Centre (TCC) of Autostrada del Brennero (called CAU) and included in its management system.

The C-ITS Server (C-ITS-S) is also interfaced with the AMQP broker, exploiting a serializer that writes messages to the AMQP broker. The solution used, based on mobile network technologies (i.e., cloud), permits to increase the availability of the service. This approach has the advantage of maximizing coverage, but also the disadvantage of a higher communication latency compared to direct short-range communication between vehicles and infrastructure.

The hybrid communication architecture foresees three players:

- The producer (typically a virtual RSU), delivering messages;
- The consumer (typically the OBU), subscribing and receiving messages;
- The broker: acting as a mediator and redistributing to consumers the messages obtained by the producers.

The connection between producers / consumers and the broker is persistent and the forwarding of messages is asynchronous. This means having a real-time notification which does not requires continuous interrogations by users as the paradigm foresees an initial subscription mechanism.

Using this infrastructure there is no need to put in direct communication Autobrennero C-ITS in Italy with other C-ITS-S in other countries. The interoperability between systems is done at broker level.

3.2.4.2 Autobrennero Road Operator (C-ITS) backend

The A22 C-ITS system is an existing infrastructure (see Figure 32) that has been adapted and configured for use within the 5G-CARMEN project and demonstrated in the first phase of the project to support the Situation Awareness and Green Driving use cases. The information collected with this infrastructure could be used also in the current selected use cases to improve information collected in the S-LDM or to refine Estimated Time of Arrival computed by the Back Situation Awareness function.

Events are notified to the C-ITS Server (C-ITS-S) with push methodology, using a DATEX II-based protocol without extensions. This structure allows the easy realization of a possible interconnection with a generic TCC.

The C-ITS-S integrates information of current events coming from the TCC with detailed geographical data in the area concerned by the events and pre-codes information in a suitable format (pre-coding of DENM, IVIM, CAM structures) to air transmission for the Road Side Units (RSUs).

The C-ITS server also manages the dissemination policies of C-ITS messages along the motorway axis, activating the transmission of I2V communications in an area of interest according to the event, its location and the RSUs next to it. The DENM and IVIM messages are thus distributed geographically on the RSUs based on the position of the event, involving only the RSUs that are near the event itself.

The computer protocol used for the communication between the server C-ITS and the RSUs is the HTTP protocol and the Websocket protocol (allowing both server and client to push messages at any time).

The C-ITS server integrates a management web interface dedicated to the management and visualization of the events in progress on an interactive map and to the management of the C-ITS infrastructure.
3.2.4.3 Predictive Quality of Service (pQoS) API

The predictive QoS API consists of a client (either in a vehicular on-board unit or as part of a vehicle backend cloud systems) and the predictive QoS server running in an MNO cloud environment.

For 5G-CARMEN, only the variant with a vehicular pQoS client and an MNO cloud-based server will be used.

The pQoS API request initially comprises two major data elements:

1. The UE capabilities (unique ID, 3GPP UE category, supported frequency bands, number of antennas);
2. A string of waypoints of the route for which QoS predictions are being requested (latitude 1, longitude 1, altitude 1, estimated time of arrival 1…, latitude n, longitude n, altitude n, estimated time of arrival n).

The pQOS API response provides a set of information for the requested waypoints containing the following data elements:

1. Area (lat/lon/alt);
2. Parameters (uplink and downlink minimum data rate, uplink and downlink maximum latency, uplink and downlink packet error rate, network type (i.e., none, 2G, 3G, 4G, 5G)).

Further data elements are being considered to be added to the pQoS API depending on the results and experiences gained from the respective 5G-CARMEN trial and pilot rounds.

Figure 32 A22 C-ITS system infrastructure.
The general architecture of pQoS is depicted in Figure 33 (see also [18] and [19]).

### 3.2.4.4 Precise Positioning

Precise Positioning is a service which provides correction information to GNSS receivers via mobile networks, thus contributing on average to centimetre-level or low decimetre-level positioning accuracy.

A network of reference stations spanning already large parts of Europe and other regions across the globe, providing regional positioning information and the perceived positioning error to a centralised service, which computes regionalised correction information and delivers it to the clients (e.g., vehicles) via mobile networks, where a positioning software applies the correction information to the positioning information it receives from GNSS.

This data fusion, which can also be combined with local positioning capabilities such as inertial sensors or wheel odometry, results in the level of positioning accuracy required for mission-critical vehicle manoeuvres with ADAS level 3 and beyond.

The architecture of Precise Positioning is illustrated hereafter in Figure 34:
Figure 34 Precise Positioning architecture.
4 Way forward

The architecture presented in this document has been defined to be generally enough to handle CCAM services that can leverage on the capabilities of 5G networks and MEC platforms even in cross-border scenarios, as in the challenging use cases selected by the project.

The current mobile network setup is based on commercially available technologies, which still have some limitations in terms of support of ultra-reliable low latency communication services, has will be discussed hereafter. However, these solutions are part of an evolutionary path defined by the 3GPP standard for the rollout of 5G networks, that will allow to transition smoothly to future deployments where even more impressive network KPIs will be reachable.

This evolution can be easily integrated with the presented architecture, in which certain elements have been defined having already in mind the potentiality of future 5G network architectures, e.g., the Edge Orchestration platform that already supports network slicing capabilities for its components, and that is built through “cloud ready” virtual network functions, with an approach that is proper of the future standalone 5G network.

The path towards “edge” cloudification in 5G networks can greatly benefit the scalability of the solutions here presented. So, this aspect will be discussed hereafter for the S-LDM component, which is one of the more challenging in the overall architecture in this regard, being in charge of processing all the messages being sent by the vehicles it manages.

4.1 Evolution of 5G to support URLLC

The support of Ultra-Reliable Low Latency Communication (URLLC) has been one of the founding pillars of the 5G system since its first definition in the ITU R M.2083, “IMT Vision” document [20], together with the support for enhanced Mobile Broad Band (eMBB), and Massive Machine Type Communication (mMTC). However, the standardization path towards 5G has decided to follow an incremental approach, with new features being added from the first release of 3GPP specifications devoted to 5G, Release 15. In the so called “early drop” of Release 15, the standard introduced a network architecture that it is still strongly relying on 4G to provide the first 5G enabled services. In this architecture, also known as Non-Standalone (NSA), 5G is available only on the radio access as an additional link through the mechanism of Dual Connectivity, thus building on an existing 4G connection that will be in charge to manage the connection and handle the whole control plane of the communication. The 5G radio link to a gNB will be only added to an existing 4G radio link, and the user traffic will be then routed and managed through a traditional LTE evolved Packet Core. The declared intention was to provide a way to support mainly eMBB services with an architecture that could be deployed right from the start leveraging on existing 4G networks, while standard activity was still working to close the specification of the new 5G Core.

Even when 5G Core is not available, the 5G New Radio access theoretically offers some features that could be used to reduce latency compared to existing LTE networks.

A full list of these features has been presented in Deliverable D3.3 [18] and are summarized hereafter, with some considerations on their applicability in nowadays commercial networks:

- **Short slot duration.** A key new feature in 5G is the introduction of flexible sub-carrier spacing (SCS). Whereas in LTE the SCS was fixed to 15kHz, in 5G, values of 15 kHz, 30 kHz, 60 kHz, 120 kHz and 240 kHz are allowed. This is one of the major differences between 5G and LTE that aims to reduce transmission latency by decreasing the time length of Orthogonal Frequency Division Multiplexing (OFDM) symbols, which is inversely proportional to the used SCS. The reference frame duration in both LTE and 5G New Radio (NR) is 10 ms containing 10 subframes of 1 ms each. In LTE the scheduling of user packets is done on a subframe basis, i.e., every 1 ms. In 5G NR the scheduling of resources in time is done every 14 OFDM
symbols, which are defined as one slot, and correspond to 1 ms with an SCS of 15 KHz (unless mini-slot are considered, see hereafter). With an higher SCS it will be possible to have more slots in a 1 ms subframe (2 slots for SCS 30 KHz, 4 for SCS 60 KHz, etc.), therefore having higher scheduling opportunities and being faster in reacting to packet transmission request. Moreover, it provides higher retransmission possibilities, crucial also to increase reliability. On the other hand, a higher SCS implies larger transmission bandwidth given the same number of sub-carriers, and reduces the robustness of transmission to fading effects in the frequency domain, which is particular critical for low to mid bands. For this reason, the standard has linked the SCS to be used to the band used for transmission, and larger SCS, that corresponds to shorter symbol durations, can only be used in mmWave transmission. In the typical 3.4-3.7GHz deployments (band n78) that are currently used for wide area coverages with 5G, 30 KHz subcarrier spacing is used, which reduces the advantages achievable in terms of latency improvements, as the slot duration is only halved.

- **Mini-slot transmission.** Another important feature introduced in Release 15 is the support for sub-slot-based transmission, also called mini-slot transmission. As previously mentioned, in 5G NR a slot is composed of 14 OFDM symbols, and in slot-based transmission the transmission can only start at the beginning of the slot; so, if a packet arrives after the starting of the OFDM symbol, it must wait until the first OFDM symbol of the next available slot to be transmitted. This alignment time adds some extra latency that might be harmful to URLLC services with a low latency requirement. Therefore, in 5G, sub-slot based transmission is introduced, where a packet can be scheduled without waiting the beginning of the slot, and using only 2, 4 or 7 OFDM symbols for its transmission (a mini-slot). In this way a packet transmission has more occasions to start in a slot instead of only one opportunity, and the waiting time before an arriving packet can be transmitted is reduced. In today commercial networks, however, this functionality is usually not present, and also the support in devices chipset is limited.

- **Self-contained slots.** While LTE supported either Frequency Division Duplexing (FDD) or Time Division Duplexing (TDD) to separate uplink (UL) and downlink (DL) traffic, 5G NR supports a Dynamic TDD approach, and there is theoretically full flexibility as each slot can be independently configured to be DL/UL using the Slot Format Indicator (SFI) carried in the physical layer control channel PDCCH. The SFI selects one out of 57 possible configurations, going from all OFDM symbols in downlink or uplink, or a mix of them. Using a slot composed of a mix of DL and UL symbols allows to have a “self-contained transmission”, where data is scheduled, transmitted and already acknowledged in the same slot (or even mini-slot). Nevertheless, this type of configurations will be more likely in millimeter waves than in midbands. Unfortunately, in nowadays networks in midband range of operation, the TDD frame structure is fixed and it has been defined by national regulators to avoid cross-link interference between transmission of different operators or systems operating on adjacent channels. Cross link interference will occur when simultaneous transmissions in uplink (UL) and downlink (DL) directions take place in different TDD networks. The national regulators in Italy and Austria imposed in band n78 (3.4-3.8 GHz) the same frame structure of DDDDDDDDSUU (although with a 3 ms offset between the twos), where D represents a slot with only OFDM symbols for DL transmission, U represents a slot with only OFDM symbols for UL transmission, and S is a special slot with 6 DL, 4 silent and 4 UL symbols. Germany imposed a frame format of DDDSUUDDSUU. The difference in the frame format between IT/AT on one side and DE on the other side is given by the protection of LTE/WiMAX-based legacy Fixed Wireless Access (FWA) applications existing in IT and AT. Anyway, the selected frame formats follow recommendations from GSMA for this operating band [21][22]. This makes it impossible to support this kind of dynamic TDD configuration, and also prevents other latency reduction mechanism, as will be discussed hereafter.

- **Pre-emption.** In 5G NR, a mini-slot carrying high-priority or delay-sensitive data in DL can pre-empt an already ongoing slot-based transmission on the first available OFDM symbol without waiting until the next free transmission resource. This operation enables ultra-low latency for mini-slot-based transmission, as this feature allows URLLC services to pre-empt...
resources already allocated to eMBB services. The pre-empted resources are communicated to the eMBB UE based on a Puncturing Indication (PI) carried in next slot Downlink Control Information (DCI). Those resources should not be taken into account for the decoding/combining of retransmissions. As mini-slot transmissions are not currently supported in commercial networks, also pre-emption is a mechanism that is theoretically available but not deployed currently.

- **Configured UL grant.** 5G NR supports both grant-based and grant-free access schemes for the uplink. The former is the more traditional mode of operation, which is similar to LTE DL/UL and NR DL. In the UL grant-based scheme, upon data arrival at UE RLC queues, the UE requests an UL grant by sending a Scheduling Request (SR) to the gNB over the physical uplink control channel PUCCH. Then, the gNB sends the UL Scheduling Grant (SG), with a Downlink Control Information (DCI) in the physical downlink control channel PDCCH, to indicate the resources that can be used by the UE to transmit. This first scheduling assignment may not be sufficient for the complete UL data transmission, since the gNB does not know the accurate requirement (e.g., buffer status) at the UE yet, so it usually schedules a limited amount of resources just to start the data exchange. After receiving the UL scheduling grant, the UE performs the data transmission in the allocated resources, which may contain UL data and/or a Buffer Status Report (BSR), indicating the remaining amount of data in the UE buffer. If a BSR is received, the gNB knows the UE buffer status and can proceed with another UL grant to account for the remaining data. Note that, depending on the packet size and the amount of resources granted in the first scheduling assignment, the UL packet transmission may end either after a 3-step process (SR → UL-grant → UL-data) or after a 5-step process (SR → UL-grant → UL-data + BSR → UL-grant → UL-data). This signalling overhead, needed to indicate to the UEs the resources to be used to perform the communication, and exchange information on the Buffer status, leads to undesirable communication delays in the uplink. For this reason, in 5G Release 15 grant-free uplink access was introduced. Grant-free provides a faster access to the channel since the SR and 5G phases are removed based on Semi-Persistent Scheduling (SPS). The gNodeB can configure the UE to have pre-allocated periodic radio resources available for transmissions. More precisely, the gNodeB provides configured UL transmission opportunities to the UEs. N Transmission Occasions (TO) are configured within a period P for repetition and retransmission. Therefore, a device can directly transmit when it has data to send, reducing the average radio access delay for uplink data. However, in nowadays commercial networks, that are more focused on eMBB type of services, this kind of UL access scheme is not implemented.

- **Processing times.** As a further improvement, in 5G NR the requirements on the device and network processing times in terms of scheduling latencies and hybrid ARQ (HARQ) retransmission latencies are tightened significantly compared to LTE. The timing relations in uplink and downlink defined by 3GPP are given by [23]:
  - K1 → Delay in TTI between downlink data (carried by PDSCH) reception and corresponding ACK/NACK transmission on uplink.
  - K2 → Delay in TTI between uplink grant reception in downlink and uplink data (carried by PUSCH) transmission.
  - K3 → Delay in TTI between uplink NACK reception and corresponding retransmission of data (PDSCH) in downlink.
  - K4 → Delay in TTI between uplink data reception and corresponding ACK/NACK.

As shown in Figure 35, in the case of LTE in Frequency Duplexing Division (FDD), the ACK/NACK for downlink data PDSCH or the uplink data corresponding to the uplink grant transmitted to the UE in subframe n is sent in subframe n + 4. In the case of NACK or PUSCH transmitted in n + 4, the eNB retransmits the downlink data or the uplink ACK/NACK in subframe n + 8. So, in LTE there is a fixed timing relation for transmission and acknowledgement. On the other hand, in NR, a flexible timing relation has been introduced, since the acknowledgment can be transmitted in sub-frame n + 1, 2, 3 or 4. K1 and K2 will be configured depending on the UE processing capability and network load.
Figure 35 FDD timing relations for (a) LTE; (b) NR as shown in [23].

This flexibility can be fully exploited also in TDD when Flexible TDD slot structure is used, so that UL or DL transmission can be scheduled according to the need. However, as previously stated, in nowadays networks in midband range of operation, the TDD frame structure is fixed and for band n78 it has been defined by national regulators in Italy and Austria as DDDDDDSUU (with a 3 ms offset), while Germany imposed a frame format of DDDSUDDDSUU. However, these frame structures greatly impact the latency that can be achieved with 5G, in particular for UL traffic with the configurations used in Italy and Austria. Indeed, any uplink transmission will have to wait for the first occurrence of an UL slot in the frame structure, limiting the possibility to use the lowest value of K1 and K2 when an UL slot is not available right after the PDSCH or UL Grant has been received. It should be stressed again that, as previously stated, this fixed frame structure also prevents the possibility to use some of the most promising tools available in 5G NR Release 15 to reduce latency, in particular for small packets: namely the possibility to have self-contained transmission composed of both DL and UL symbols and delivered with a sub-slot based transmission (i.e., self-contained mini-slots that could carry URLLC packets almost as soon as they are generated, with pre-emption on existing eMBB traffic), as this would break the existing imposed TDD frame structure.

- **Low latency support at RLC and MAC layers.** The higher layer protocols MAC and RLC have also been designed with low latency in mind for 5G, with header structures chosen to enable processing without knowing the amount of data to transmit, thus allowing to process data with a pipeline approach to reduce processing times. This contributes to the reduction of latency experienced in nowadays 5G network.

So, while already Release 15 provides several features in the radio access to reduce latency, the majority could not be exploited in the currently used bands, due to the limitation imposed by following a fixed frame structure, or are still not available in commercial equipment, as the focus has been so far mainly on eMBB services. Therefore, while the target latency for the user-plane of URLLC services in Release 15 was 1 ms one-way for both downlink and uplink [25], the achievable latency in nowadays networks is higher, as only some of the available improvements can be exploited.

Nonetheless, further improvements have been defined in the following Releases to fully support URLLC services. Release 16 further enhances the NR support for URLLC services by enabling latency in the range of 0.5 to 1 ms and improved reliability with a target error rate of $10^{-6}$. In particular:

- **Physical Downlink Control Channel (PDCCH) enhancements**, focusing on compact Downlink Control Information (DCI) with new DCI formats for DL and UL scheduling, PDCCH repetition and increased monitoring capability.
• Uplink Control Information (UCI) enhancements, focusing on enhanced HARQ feedback methods and Channel State Information (CSI) feedback enhancements, with the possibility to have more than one PUCCH for HARQ ACK transmission within a slot.

• Physical Uplink Shared (PUSCH) enhancements focusing on mini-slot level frequency hopping and enhancements to retransmission and repetition schemes to increase reliability. In particular, one or more actual PUSCH repetitions in one slot, or two or more actual PUSCH repetitions across slot boundary in consecutive available slots are supported, using one UL grant for dynamic PUSCH, and one configured grant configuration for configured grant PUSCH.

• Specification of enhanced UL Configured Grant (CG) transmission, in particular the case of multiple active configured grants.

• Inter UE transmission prioritization/multiplexing, considering UE generating traffic with different latency and reliability requirements.

Moreover, the transition from an NSA architecture to a SA architecture will allow to benefit of further enhancements enabled by the availability of the 5G Packet Core.

Some of these enhancements include the following, as discussed in [23] and [26]:

• **Flow-based QoS mode with support for reflective QoS and standardized 5QIs.** One of the key requirements for URLLC services is the stringent end-to-end QoS goals that include low latency and high reliability. The QoS differentiation within a Protocol Data Unit (PDU) session is defined by QoS Flow, which is identified by a QoS Flow ID (QFI). Traffic associated with the same QFI receives the same QoS forwarding treatment. Each QoS Flow is associated with a set of QoS characteristics (packet delay budget, packet error rate, priority level). A standardized set of 5G QoS Indicators (5QIs) are defined and point to a set of QoS characteristics. The 5QI is similar to the QoS Class Identifier (QCI) in Evolved Packet System (EPS). A new "resource type, "Delay Critical Guaranteed Bit Rate (GBR)", is also defined. Reflective QoS (RQoS) is used to minimize the need for control-plane signaling. RQoS, instead, is achieved by creating a derived QoS rule in the UE based on the received downlink traffic QoS. The UE creates a "mirror" packet filter and associates the QoS of the downlink packet to uplink packet. The mirror packet filter and the associated QoS constitute the derived QoS rule. The UE uses the derived QoS rule to bind corresponding uplink packets on the same QoS flow.

• **Support for Network slicing.** 5G Core Network natively supports Network Slicing, i.e., the multiplexing of virtualized and independent logical networks on the same physical network infrastructure. Each network slice is an isolated end-to-end network tailored to fulfill diverse requirements requested by a particular application. Running multiple logical networks on a common physical infrastructure allows a UE to access to a portfolio of possible Network Instances offering different Control Plane and User Plane behaviors, and different customized Service Levels (i.e., meeting certain Service Level Agreements which are associated to the concept of Tenancy of a Network Slice) with efficient utilization of the network resources.

• **Native support for MEC,** which enables efficient service delivery through reduced end-to-end latency and load on the transport network. There are several MEC enablers in the 5G system:
  o User Plane Function (UPF) selection to route the user traffic to the local data network. The 5G core network selects the traffic to be routed and steered to the applications in the local data network.
  o Multiple data paths with redundant transmission in the user plane to ensure reliable delivery of application data. This helps in transmission of data with reliability higher than with single user plane.
  o Session and service continuity to enable UE and application mobility. Multi-homed IPv6 PDU sessions to support make-before-break service continuity to support Session and Service Continuity and concurrent access to local services and internet with different IPv6 prefixes.
o Application function influence on UPF (re)selection and traffic routing via Policy Control Function (PCF) or Network Exposure Function (NEF).
o Network capability exposure with 5G core network and application function providing information to each other via NEF.

- **Support for UE and network-controlled, always-on PDU sessions** to enable low-latency transmissions.
- **Enabling a new Radio Resource Control (RRC) state, RRC_INACTIVE**, which allows a UE in connected state when not transmitting or receiving data to achieve power efficiency comparable to that of an idle UE.

### 4.2 S-LDM scalability analysis

The S-LDM is a key component in the overall centralized manoeuvring service, since it is in charge to manage all messages arriving from cars and to raise alarms and relevant context data for the manoeuvring component, thus an analysis of its scalability properties is necessary.

For this reason, a number of scalability tests are ongoing at CNIT premises in Politecnico di Torino, using a testbed built on a local computing cluster. This local testbed is based on virtual machines (VMs) that have the same specifications of those deployed in MEC nodes to run the AMQP broker and orchestrated edge services. The scope of this activity is to identify possible bottlenecks in the S-LDM component under typical usage conditions, spanning from light to severe load conditions. For this reasons, ongoing tests are using varying numbers of emulated cars as well as different rates of messages sent per car, in order to evaluate not only response times, but also memory and CPU usage in different conditions. Thus, this analysis is in charge to evaluate vertical scalability of the service, by analysing the component performance (service time and resource consumption) as a function of the amount of computing resources assigned to it in MEC nodes. Since requests arriving from cars have to be served sequentially by the S-LDM, delay statistics can be predicted as a function of the traffic load by using classic queueing models, if a preliminary estimate of the average service time (function of assigned resources) is available. The relevant results will be presented and discussed in a forthcoming WP5 deliverable.

In addition to the evaluation of vertical scalability, it is worth to mention that a further architectural alternative is horizontal scalability, which can be used in conjunction with the former one. In fact, it is possible to deploy multiple instances of S-LDM in the same MEC node by using different containers orchestrated by the platform described in Section 3.2.2, where each instance is associated to a different tile square of terrestrial projection of the coverage identified by the quadtree key (Quadkey, see D2.2 [2] and D3.3 [18] for further details). However, this solution may be not effective for the management of the overlapping areas that need to be addressed by two neighbouring S-LDM, both in intra and inter-MNO scenarios. A complete discussion about the management of the overlapping coverage between neighbouring instances (also cross-border) is addressed in detail in D2.2 deliverable [2]. A further and simpler alternative is making all different instances of S-LDM in a MEC node listen to all messages coming from the same area (or from a set of common tiles), by filtering out those that are not relevant to the portion of coverage that each instance of the service is in charge to manage. Even if a portion of the service capacity is used for filtering out-of-scope messages, the complete processing is carried out only for a subset of the received messages, thus the solution is effective for implementing horizontal scalability.
5 Conclusion

This document aimed at detailing the 5G-CARMEN technical specifications by providing the final specifications of the 5G-CARMEN system architecture including its sub-components, their interfaces, and the protocols to be used for the data exchange. 5G-CARMEN utilizes 3GPP standardized C-V2X which supports various communication links and interface including:

- PC5 interface for V2V, V2I, and V2P communication;
- Uu interface for V2N communication.

The 5G-CARMEN use cases leverage PC5 connectivity utilizing it as a complementary link to the Uu V2N connectivity.

Section 2 elaborated the functional analysis of the selected use cases providing additional information on the management of cross-border scenario. Different approaches are described, depending on the use case, which leverage on the characteristics of the different functional elements that are involved in the realization of the selected use case.

Section 3 proposed the overall 5G-CARMEN system architecture by summarizing the description provided in the previous deliverable D2.2 and detailing reference points and interfaces. The main interfaces are between vehicles and the infrastructure, on the MEC level between the Applications, system-related orchestration and platform interfaces, as well as interfaces towards Cloud components such as analytics platforms and C-ITS backends. Since the architectural objectives of this deliverable are the use cases and the cross-border approach, the Orchestrated Edge platform architecture is explained in more detail in the deliverables of the Work Package 4 [13][17].

Finally, section 4 discussed the current limits of nowadays commercial mobile networks, and how the evolution of 5G defined in the 3GPP standard will enable new features to support Ultra-Reliable Low Latency Communication (URLLC) services, which could be integrated in the proposed architecture. It also showed how the approach followed in the definition of 5G-CARMEN architecture can ensure scalability for the different components necessary to the realization of the project use cases, taking one of the most demanding components, S-LDM, as example.
References

[5] 3GPP TS 22.129 “Service aspects; Handover requirements between UTRAN and GERAN or other radio systems”
[7] ETSI GS MEC 003 “Multi-access Edge Computing (MEC); Framework and Reference Architecture”
[8] ETSI GS MEC 012 "Mobile Edge Computing (MEC); Radio Network Information API"
[10] ETSI GS NFV-IFA 030: " Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Multiple Administrative Domain Aspect Interfaces Specification”
[12] ETSI TS 102 636-1 V1.1.1, Vehicular Communication, GeoNetworking
[14] 3GPP TR 23.725 “Study on enhancement of Ultra-Reliable Low-Latency Communication (URLLC) support in the 5G Core network (5GC)”
[15] ETSI GS NFV-IFA 013: "Network Functions Virtualisation (NFV); Management and Orchestration; Os-Ma-Nfvo reference point - Interface and Information Model Specification”
[17] 5G-CARMEN, D4.3 “Deliverable D4.3 Results on the validation of the 5G-CARMEN platform”, March 2022
[19] 5G-CARMEN, D3.4 “Deliverable D3.4 Final report on 5G Technological Enablers for CCAM”, April 2022
[23] NGMN, “5G E2E Technology to support verticals URLLC requirements”, February 2020
[24] 3GPP TS 38.214 “NR; Physical layer procedures for data (Release 15)”
[26] 5G Americas Whitepaper “New services and applications with 5G ultra-reliable low latency communications”, November 2018