5GAA Automotive Association

Vehicle to Network to Everything (V2N2X) Communications; Architecture, Solution Blueprint, and Use Case Implementation Examples

5GAA Automotive Association
Technical Report

CONTACT INFORMATION:
Executive Manager – Thomas Linget
Email: liaison@5gaa.org

MAILING ADDRESS:
5GAA c/o MCI Munich
Neumarkter Str. 21
81673 München, Germany
www.5gaa.org

VERSION:	2.0
DATE OF PUBLICATION:	3 June 2025
DOCUMENT TYPE:	Technical Report
EXTERNAL PUBLICATION:	Yes
DATE OF APPROVAL BY 5GAA BOARD:	24 April 2025

Contents

Foreword

This Technical Report has been produced by 5GAA.

The contents of the present document are subject to continuing work within the Working Groups (WG) and may change following formal WG approval. Should the WG modify the contents of the present document, it will be re-released by the WG with an identifying change of the consistent numbering that all WG meeting documents and files should follow (according to 5GAA Rules of Procedure):

x-nnzzzz

(1) This numbering system has six logical elements:

(a) x: a single letter corresponding to the working group:
where x =
T (Use cases and Technical Requirements)
A (System Architecture and Solution Development)
P (Evaluation, Testbed and Pilots)
S (Standards and Spectrum)
B (Business Models and Go-To-Market Strategies)

(b) nn: two digits to indicate the year. i.e. ,17,18 19, etc

(c) zzzz: unique number of the document

(2) No provision is made for the use of revision numbers. Documents which are a revision of a previous version should indicate the document number of that previous version

(3) The file name of documents shall be the document number. For example, document S-160357 will be contained in file S-160357.doc

Contents

Executive summary

This Technical Report (TR) complements the 5GAA whitepaper "Road traffic operation in a digital age" and describes for different stakeholders how to realise various V2X applications and use cases (UCs), using cellular network communications in combination with information sharing structures between backend systems.

The TR provides an application-level reference blueprint architecture and introduces an "information sharing domain" to facilitate a federated, scalable digital data exchange between ecosystem stakeholders, e.g., Vehicle OEMs, Service Providers and Infrastructure Owners and Operators (IOOs)[1]. The TR provides descriptions on how to realise V2X applications of different types utilising cellular network communications and information sharing, with different protocols and deployment options across stakeholder domains, and exemplified with safety- and mobility-enhancing UCs, such as *Traffic event information sharing, Traffic signal information sharing, Traffic signal priority request, Emergency Vehicle Approaching, HD MAP handling, Automated valet parking, Object Detection and Sharing, and Vulnerable Road User protection*. The TR also clarifies the different implementation options of the application in a vehicle and related implications, namely *OEM-controlled App (OEM App), OEM-supported SP App,* and *OEM-independent SP App.* This TR furthermore describes verified solutions and includes references to initial operational deployments that realise the suggested application-level reference architecture, e.g., C-Roads, Talking Traffic, Mobilidata, etc.

Ecosystem stakeholders like vehicle OEMs, Service Providers, and IOOs[1], who are interested in deploying V2X services, are encouraged to use this TR as a handbook of deployment solutions using cellular network and information sharing with examples

[1] IOO is an umbrella term used global wise for different local and regional actors in V2X ecosystems, e.g., road traffic authorities, road operators, cities, parking area providers.

of reference deployments. Solutions described in this TR utilise existing commercial cellular networks and have been proven feasible and effective in accelerating the V2X service penetration by various deployments. Especially for UCs, which require interaction between road infrastructure and other road users, or UCs, where information needs to be delivered over long distance but with less stringent latency requirement, the solutions described in this TR are considered currently viable. With enhanced cellular network coverage, radio capacity and capabilities, and network features such as Mobile Edge Computing (MEC), Quality of Service (QoS) and Network Slicing, it is foreseen that also more demanding UCs can be addressed by cellular communication.

Contents

Introduction

This Technical Report presents the system architecture, blueprint solution for deployment, and end-to-end (E2E) implementation examples of V2N2X[2] communications for V2X services. The architecture and solutions described in this report focus on the E2E application layer data exchange among key ecosystem stakeholders, i.e., Vehicle OEMs, V2X Service Providers (SPs) and Infrastructure Owner and Operators (IOOs)[3]. The blueprint technical solutions developed in this work provide guidance for interoperable V2X service implementations, considering the business interests as well as go-to-market constraints of the ecosystem stakeholders. The described solutions are based on state-of-the-art cellular technologies and networks. Therefore, they can readily be implemented using existing vehicle connectivity supported by commercially operating 4G/5G cellular networks[4].

The intended readers of this Technical Report include the ecosystem stakeholders interested in implementing V2X applications using cellular networks, i.e., V2N2X communication, and anyone looking for deep technical understanding about the V2N2X architecture and implementation solutions. Readers are suggested to use this TR together with the complementary 5GAA White Paper “Road Traffic Operation in

[2] In this report, we use Vehicle-to-Network-to-Everything (V2N2X) as a general term for cellular network-based communications supporting V2X application use cases. In actual implementations, depending on the communicating end-points V2N2X may be realised as Vehicle-to-Network-to-Infrastructure (V2N2I), Infrastructure-to-Network-to-Vehicle (I2N2V), Vehicle-to-Network-to-Vehicle (V2N2V), Vehicle-to-Network-to-Pedestrian (V2N2P), or Pedestrian-to-Network-to-Vehicle (P2N2V), and even Infrastructure-to-Network-to-Pedestrian (I2N2P) or Pedestrian-to-Network-to-Infrastructure (P2N2I).

[3] IOO is an umbrella term for different local actors in V2X ecosystems, e.g., road traffic authorities, road operators, cities, parking area providers.

[4] The architecture and blueprint solutions described in this TR focus on the application layer. Example use cases and deployment solutions in Chapter 8 and in the annexes in principle work with 4G connectivity. Large-scale deployment of such use cases will benefit from higher system capacities, latency performance, and sophisticated QoS mechanisms of 5G network.

a Digital Age: A Holistic Cross-Stakeholder Approach" [13], which offers an overview of V2X ecosystems and guiding principles for sharing digital information across stakeholders. The White Paper also gives concrete recommendations to policy- and decision-makers.

This Technical Report is organised as follows:

- Chapter 4 describes the V2N2X Application Layer Architecture covering the V2X ecosystem stakeholders, i.e., Vehicle OEMs, V2X SP, and IOOs, with highlights on inter-stakeholder interfaces and the Information Sharing Domain enabling scalable and interoperable data exchange across ecosystem stakeholders.
- Chapter 5 provides an overview of the high-level flow from the V2X application process perspective, which involves V2N2X system components from different ecosystem stakeholder domains.
- Chapter 6 presents the V2N2X blueprint deployment options focusing on the usage of inter-stakeholder interfaces defined in the V2N2X Application Layer Architecture. In this chapter, sections of blueprint deployment options are organised according to different ecosystem stakeholders, so readers from a specific stakeholder group, e.g., vehicle OEMs, SP, or IOO, can find the deployment options that are most relevant to their interests. Each section contains link(s) to corresponding V2N2X application use case implementation example(s) in Chapter 8, which provide the readers with a concrete E2E overview. Section 6.4 is dedicated to the Information Sharing Domain to provide sufficient technical details for the readers to understand its essential role in enabling scalable and interoperable V2X data exchange across a large number of ecosystem stakeholders.
- Chapter 7 explains the technical features of cellular networks as well as the deployment options of the in-vehicle system for V2N2X applications.
- Chapter 8 presents the E2E V2N2X implementation for selected application use cases, including Traffic Event Information Sharing, Traffic Signal Information Sharing, Traffic Signal Priority Request, Emergency Vehicle Approaching, HD MAP Handling, Automated Valet Parking, Object Detection and Sharing, and Vulnerable Road User Protection.
- Chapter 9 concludes the V2N2X architecture and blueprint solution with recommendations for the readers from different V2X ecosystem stakeholders.
- Chapter 10 summarises the go-to-market and business considerations of V2N2X deployments based on the 5GAA Technical Report on "Business Perspectives on Vehicle-to-Network-to-Everything (V2N2X) Deployments". [21]

Annexes of this Technical Report provide further references and technical details, to help readers understanding the V2N2X system architecture and blueprint solutions:

- Annex A presents the generic application layer system architecture, which serves as the basis for the applied V2N2X application layer system architecture and the V2N2X blueprint solutions documented in this Technical Report.

- Annex B provides concrete examples of Information Sharing Instance, described in Section 6.4, based on the EU C-Roads initiative, the Talking Traffic deployment in the Netherlands, and Mobilidata deployment in Belgium.
- Annex C contains technical details of V2X message configuration using cellular communication from the Talking Traffic deployment.
- Annex D elaborates the Quadtree solution for geo-referencing used in many V2N2X applications.
- Annex E explains the 3GPP Quality of Service (QoS) mechanism and the Network Slicing concept, as well as related QoS and core network features available in cellular networks.
- Annex F provides a high-level summary of the logical interfaces in the V2N2X application layer reference architecture.
- Annex G outlines the software system and operation design principles that are recommended for implementors of V2N2X solutions.
- Annex H describes the Advanced Message Queuing Protocol (AMQP) and how to use metadata to allow filtering and facilitate data transcoding for V2X messages for interoperable data exchange cross ecosystem stakeholders.

1 Scope

The present 5GAA Technical Report provides application layer system architecture, solution blueprint, and guidance for V2X ecosystem stakeholders in the development of system solutions for V2X services utilising cellular network communications and information sharing domain.

2 References

- References are either specific (identified by date of publication, edition number, version number, etc.) or nonspecific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

[1]	Void
[2]	5GAA Technical Report "Moving Toward Federated MEC Demos/Trials", March 2023: https://5gaa.org/moving-toward-federated-mec-demos-trials
[3]	5GAA Technical Report "MEC System Interoperability and Test Framework", March 2023,: https://5gaa.org/mec-system-interoperability-and-test-framework
[4]	C-Roads "IP based Interface profile" This specification is one part of the harmonised communication profile for C-ITS services and can be requested via: https://www.c-roads.eu/platform/get-in-touch.html. (Free of charge but using a request procedure to be able to provide updated versions)
[5]	Fifth Generation Cross-Border Control (5GCroCo), Deliverable D3.1 Final Application Architecture, Version: v1.0, 2021-01-31, https://5gcroco.eu/images/templates/rsvario/images/5GCroCo_D3_1.pdf
[6]	HERE, "Quadkeys", [Online]. Available: https://www.here.com/docs/bundle/traffic-api-developer-guide-v6/page/topics/quadkeys.html. [Accessed 16-9-2019]
[7]	J. Schwartz, "Bing Maps Tile System", 2018. [Online]. Available: https://docs.microsoft.com/en-us/bingmaps/articles/bing-maps-tile-system. [Accessed 16-09-2019]
[8]	3GPP TS 23.501, "5G; System Architecture for the 5G System", v15.13.0, 23 March 2022
[9]	Ericsson White Paper, "Ericsson Dynamic Network Slice Selection", 2022: https://www.ericsson.com/48fd7e/assets/local/networks-slicing/docs/ericsson-dynamic-network-slice-selection-2022.pdf
[10]	ISO 23374-1:2023 – "Intelligent transport systems –Automated valet parking systems (AVPS) – Part 1: System framework, requirements for automated driving, and communication interface": https://www.iso.org/standard/78420.html
[11]	5GAA Technical Report, "Automated Valet Parking: Technology Assessment and Use Case Implementation Description – System Architecture and Cellular Public Network and PC5 Direct Communication Solutions", May 2023. https://5gaa.org/report-on-automated-valet-parking-technology-assessment-and-use-case-implementation-description/

[12] 5GCroco, 5GMobix, and 5GMed projects report, "5G technologies for connected automated mobility in cross-border contexts": https://5g-ppp.eu/wp-content/uploads/2023/05/5G-MOBIX_5G-CARMEN_5GCroCo_5G-Technologies-for-CAM-in-cross-border-contexts_V1.0.pdf

[13] 5GAA White Paper, "Road Traffic Operation in a Digital Age: A Holistic Cross-Stakeholder Approach", January 2024: https://5gaa.org/road-traffic-operation-in-a-digital-age-a-holistic-cross-stakeholder-approach/

[14] Ma, Jingtao, "Virtual Roadside Unit (vRSU): A unifying framework and MEC/Cloud implementations in US/China", March 2023. Traffic Technology Services (TTS).

[15] CAMARA, The Telco Global API Alliance: https://camaraproject.org/

[16] 5GAA Technical Report, "C-V2X Use Cases and Service Level Requirements Volume I", December 2020: https://5gaa.org/c-v2x-use-cases-and-service-level-requirements-volume-i/

[17] 5GAA White Paper, "Updated 2030 Roadmap for Advanced Driving Use Cases, Connectivity Technologies, and Radio Spectrum Needs", November 2022: https://5gaa.org/5gaa-publishes-updated-2030-roadmap-for-advanced-driving-use-cases-connectivity-technologies-and-radio-spectrum-needs/

[18] Mobilidata Programme: https://www.mobilidata.be/en/

[19] 5GAA Technical Report, "C-V2X Use Cases and Service Level Requirements Volume III", January 2023: https://5gaa.org/c-v2x-use-cases-and-service-level-requirements-volume-iii/

[20] 5GAA Technical Report, "C-V2X Use Cases and Service Level Requirements Volume II", January 2021: https://5gaa.org/c-v2x-use-cases-and-service-level-requirements-volume-ii/

[21] 5GAA Technical Report, "Business Perspectives on Vehicle-to-Network-to-Everything (V2N2X) Deployments", to add the URL of Task 3 TR here, once published

[22] C-Roads Platform, "WG2 Technical Aspects, Taskforce 2 Service Harmonization", C-ITS Service and Use Case Definitions, Version 2.0.8

3 Definitions, and abbreviations

3.1 Definitions

For the purposes of the present document, the following definitions apply:

Application: An implementation concept describing software and/or hardware implementation of functions required for realising a V2X service. Application implementation that directly interfaces with the V2X service user is called 'App'. To realise the V2X service, an App may require separated implementation that does not directly interact with the V2X service user. Such separated implementation is called Application Server (AS), which collaborates with the App in a service execution.

Service: A business concept describing the process of generating certain value for the service user via applications. Service process usually involves multiple service execution entities based on predefined relations.

Service user: Entity that consumes the service.

Stakeholder: Person, business or other legal entity who is involved in a service or process of a use case. Example stakeholders in V2X services include the driver or traveller, automotive OEM, service provider, road authority, mobile operator, etc.

Stakeholder domain: Part of an entity (a network, an address space etc.) that is managed by a particular commercial or administrative entity from a stakeholder.

Use case: Use cases are the high-level procedures of executing an application in a particular situation with a specific purpose. [16]

V2X Service: A service using vehicle-to-everything communications to realise the values for service users related to road transportation and mobility activities.

3.2 Abbreviations

For the purposes of the present document, the following symbols apply:

3GPP	3rd Generation Partnership Project
5GS	5G System
5QI	5G QoS Identifier
AD	Automated Driving
ADAS	Advanced Driver Assistance Systems
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
APN	Access Point Name
App	Application
APP ID	Application Identifier
AR	Augmented Reality
AS	Application Server
ATMS	Advanced Traffic Management System

AVM	Automated Vehicle Marshalling
AVP	Automated Valet Parking
AVPC	AVP Control
BSM	Basic Safety Message
CA	Certificate Authority
CAM	Cooperative Awareness Message
CCoC	Common Code of Conduct
C-ITS	Cooperative ITS
CPM	Collective Perception Message
CSP	Communication Service Provider
C-V2X	Cellular Vehicle-to-Everything
DENM	Decentralized Environmental Notification Message
DNN	Data Network Name
DTLS	Datagram Transport Layer Security
E2E	End-to-End
EPS	Evolved Packet System
ETSI	European Telecommunications Standards Institute
FM	Facility Management
FQDN	Fully Qualified Domain Name
GBR	Guaranteed Bitrate
GDPR	General Data Protection Regulation
GLOSA	Green Light Optimal Speed Advisory
GNSS	Global Navigation Satellite System
GSMA	Global System for Mobile Communications Association
HMI	Human-Machine Interface
I2N2P	Infrastructure-to-Network-to-Pedestrian
I2N2V	Infrastructure-to-Network-to-Vehicle
IEEE	Institute of Electrical and Electronics Engineers
IOO	Infrastructure Owner and Operator
IoT	Internet of Things
ISI	Information Sharing Interface
IT	Information Technology
ITS	Intelligent Transport System
IVIM	Infrastructure to Vehicle Information Message
LBO	Local Breakout
LTE	Long-Term Evolution
MAPEM	MAP (topology) Extended Message
MBB	Mobile Broad-Band
MEC	Mobile Edge Computing
MNO	Mobile Network Operator
MMI	Multimedia Interface
MQTT	Message Queueing Telemetry Transport
NAP	National Access Point
NEF	Network Exposure Function
NI	Network Identifier
NSSAI	Network Slice Selection Assistance Information
OEM	Original Equipment Manufacturer
OID	Operator Identifier

P2N2I	Pedestrian-to-Network-to-Infrastructure
P2N2V	Pedestrian-to-Network-to-Vehicle
PDU	Packet Data Unit
PDB	Packet Delay Budgets
PDN	Packet Data Network
PER	Packet Error Rates
P-GW	Packet Gateway
PKI	Public Key Infrastructure
PSA	PDU Session Anchor
QCI	QoS Class Identifier
QoD	Quality on Demand
QoS	Quality of Service
RAN	Radio Access Network
RVO	Remote Vehicle Operation
SAE	Society of Automotive Engineering
SD	Slice Differentiator
SDK	Software Development Kit
SDO	Standardisation Development Organisation
SDSM	Sensor Data Sharing Message
SLA	Service Level Agreement
S-NSSAI	Single – NSSAI
SP	Service Provider
SPaT	Signal Phase and Timing
SPATEM	Signal Phase And Timing Extended Message
SSEM	Signal request Status Extended Message
SREM	Signal Request Extended Message
SRTI	Safety Related Traffic Information
SST	Slice Service Type
TCU	Telematic Control Unit
TSI	Traffic Signal Information
TLS	Transport Layer Security
TR	Technical Report
TVRA	Threat, Risk, Vulnerability Assessments
UC	Use Case
UE	User Equipment
UPF	User Plane Function
URL	Uniform Resource Locator
URSP	UE Route Selection Policy
V2I	Vehicle-to-Infrastructure
V2N2I	Vehicle-to-Network-to-Infrastructure
V2N2P	Vehicle-to-Network-to-Pedestrian
V2N2V	Vehicle-to-Network-to-Vehicle
V2N2X	Vehicle-to-Network-to-Everything
V2P	Vehicle-to-Pedestrian
V2X	Vehicle-to-Everything
VDOT	Virginia Department of Transportation
vEPC	virtual Evolved Packet Core
VMC	Vehicle Motion Control
VRU	Vulnerable Road User

4 Application layer reference architecture of V2N2X

The application layer reference architecture of V2N2X is shown in Figure 1[5]. This architecture includes identified ecosystem stakeholders, their domains and system components, as well as logical interfaces at the application layer that are needed for the end-to-end implementation of V2X services using cellular network communications and information sharing. All interfaces in Figure 1 are logical interfaces at the application layer. The implementation details of each interface depend on the deployment options, e.g., using cellular network (Uu interface) or other communication technologies.

The V2N2X application layer reference architecture in Figure 1 can be applied in the implementation of selected V2X services using specific V2N2X deployment option(s) documented in this Technical Report. In some V2N2X implementations, only a subset of the stakeholders, system components, and logical interfaces are needed. The architecture in Figure 1 helps in identifying ecosystem stakeholders, functional allocation, as well as interfaces that need a harmonised or agreed profile[6] for interoperability reason.

Chapter 6 describes logical interfaces in Figure 1, according to the viewpoint from V2X ecosystem stakeholders, namely Vehicle OEM, Service Provider, IOO. Some of the interfaces may need implementation profiles that are harmonised or agreed upon among relevant stakeholders. Annex F provides a table summarising the logical interfaces together with information from implementation examples of some interfaces.

[5] The application layer reference architecture in Figure 1 is an applied system architecture of the generic V2X architecture to V2N2X, as described in Annex A.

[6] Depending on the interests of relevant ecosystem stakeholders, Harmonised or agreed profiles for the identified interfaces may or may not be standardised in Standardisation Development Organisations (SDOs).

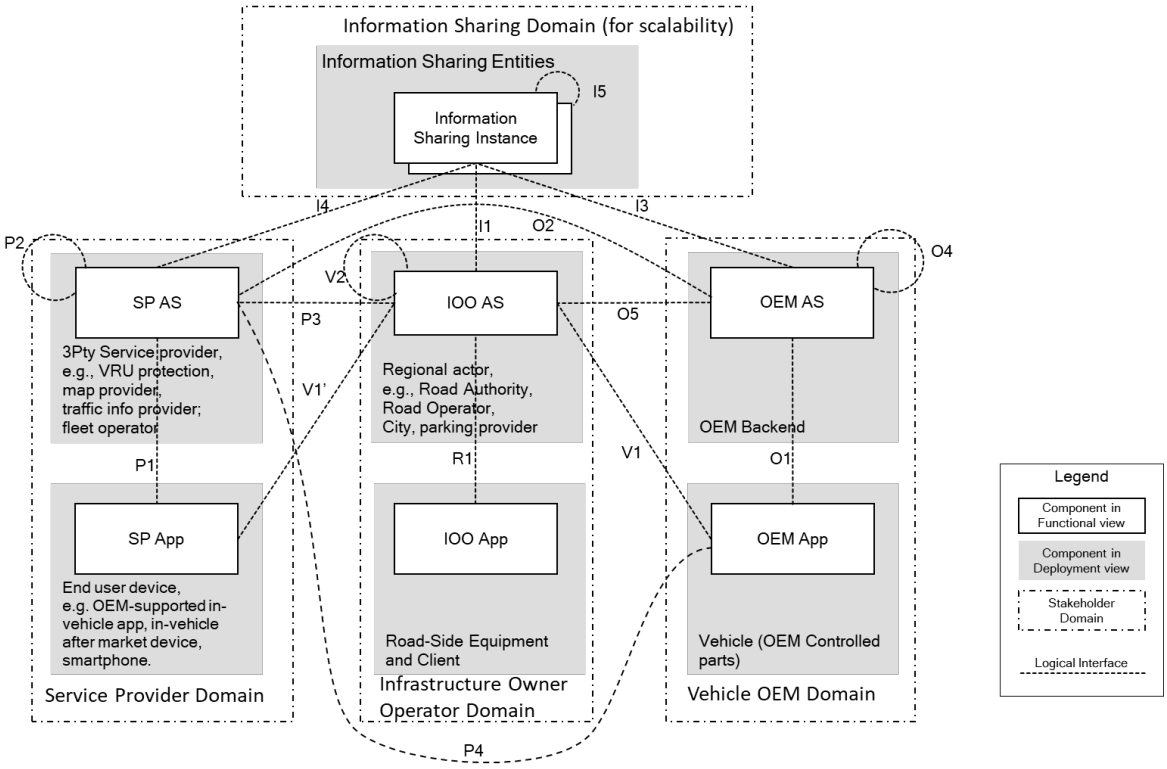


Figure 1: Application layer reference architecture of V2N2X[7, 8]

Brief description of system components

- Infrastructure Owner and Operator AS: In the IOO Domain, this system component is mainly a regional actor that provides services related to the automotive and transport domains. The service is often based on the interaction with IOO App, which is the infrastructure owned by the IOO. This IOO actor could for example be a city, road authority, road operator, or parking provider.
- IOO App: In the IOO Domain, this system component is the infrastructure owned by the IOO including road-side equipment, sensors, and road/parking facilities, etc.
- OEM AS: In the Vehicle OEM Domain, OEM backend component managing OEM App, e.g., control allowed connections for the vehicles. For some services the OEM AS will act as a proxy and filter for the information flow to/ from the vehicle OEM App.
- OEM App: In the Vehicle OEM Domain, in-vehicle component that implements the service function(s) for the service user. For the service to function, the App needs to receive data from other system components. It may implement the function of warning the human driver according to the situation and/or supporting ADAS/AD features in the vehicle.

[7] This architecture figure is developed in the 5GAA V2N2X work item. When this architecture (Figure 1) is used outside of the present Technical Report, a note needs to be added stating that the system architecture shall be used always with reference to the 5GAA V2N2X Technical Report (the present document), where system components and interfaces in this architecture are defined for the V2N2X communication solution blueprint.

[8] The term 'V2X AS' has been defined in early 3GPP work as a generic name for an application providing services related to automotive. The same term has also been defined in 5GAA as the functional entity of exchanging C-ITS messages with 'V2X App'. To minimize confusion due to differing definitions in various sources, the use of this term is avoided in this technical report. Therefore, to better reflect the eco-system actors, this technical report uses the terms 'SP AS', 'OEM AS' and 'IOO AS' to clarify actors and related application servers.

  - Note 1: The OEM App component also considers all related in-vehicle software or hardware system-components to ensure the intended V2N application is functioning correctly. As the details may vary from OEM to OEM and from vehicle model to vehicle model, such details are not illustrated in the V2X application layer reference architecture. Considering the scope of this TR is E2E V2N2X solutions, the OEM App component is not further broken down for the in-vehicle deployment structure.
  - Note 2: Different in-vehicle application deployment options, including the OEM-controlled App (OEM App), are described in Section 7.2.

- Service Provider (SP) AS: In the Service Provider Domain, Service Provider Application Server (SP AS) is a collective term for actors providing services related to the automotive domain. Some examples of services it may provide are VRU protection services, MAP services, traffic info services, and fleet operator services. A SP AS may provide one or multiple services depending on Service Provider area or expertise.
- SP App: In the Service Provider Domain, component that implements the service function(s) in the end user device for the service user. For the service to function, the App needs to receive data from other system components. End user device may be, for example a smartphone, in-vehicle aftermarket device, as well as OEM infotainment system.
  - Note: In-vehicle application deployment options, including the OEM-supported SP App and OEM-independent SP App are described in Section 7.2.
- 'Information Sharing Instance(s)': In the Information Sharing Domain, backend component(s) are interconnected for scalability and to federate the data in order to avoid full mesh connectivity among actors. The Information Sharing Domain supports, for example, service discovery, service subscription and the forwarding of information among backend components from different stakeholders. Interaction with Information Sharing Instances and between Information Sharing Instances should use standard IT technologies, e.g., using TCP/IP for transport layer, AMQP for information sharing (publish/subscribe), and metadata to identify payload, relevant area (e.g., based on quadtree tile concept, see Annex D) etc. to facilitate filtering and facilitate mechanisms for data format transcoding. Information sharing principles are further described in Section 6.4., and system design principles further described in Annex G: Software system and operation design principles. The use of metadata is further described in Annex H: AMQP, metadata and interoperability. National Access Points (NAPs)[9] for safety related traffic information and real-time traffic information are examples of an Information Sharing Instance.

[9] National Access Points are nodes facilitating the exchange of ITS and ITS-related data. More information available at: https://napcore.eu/description-naps/

In the “Domain” of each ecosystem stakeholder, the respective stakeholder is responsible for the operation of services. System components, functionality, protocols, security, etc. are under the control of the stakeholder.

In the “Information Sharing Domain” the interconnected actors form a trust domain. Having agreements in place on what to share and how, data quality, security, etc., this domain becomes important for resolving the scalability challenge in real deployment.

5 Generic V2N2X process flows

The general high-level sequence diagram for the E2E V2X service process is shown in Figure 2. Step 1 "Ecosystem Preparation" and Step 2 "Service Preparation" are not the focus of this document, but they are necessary steps for the real operation of the service and involve interfaces among backend components e.g., O2, O5, P3, I1, I3, I4, etc. Different deployment options may have different details, as described in Section 6.

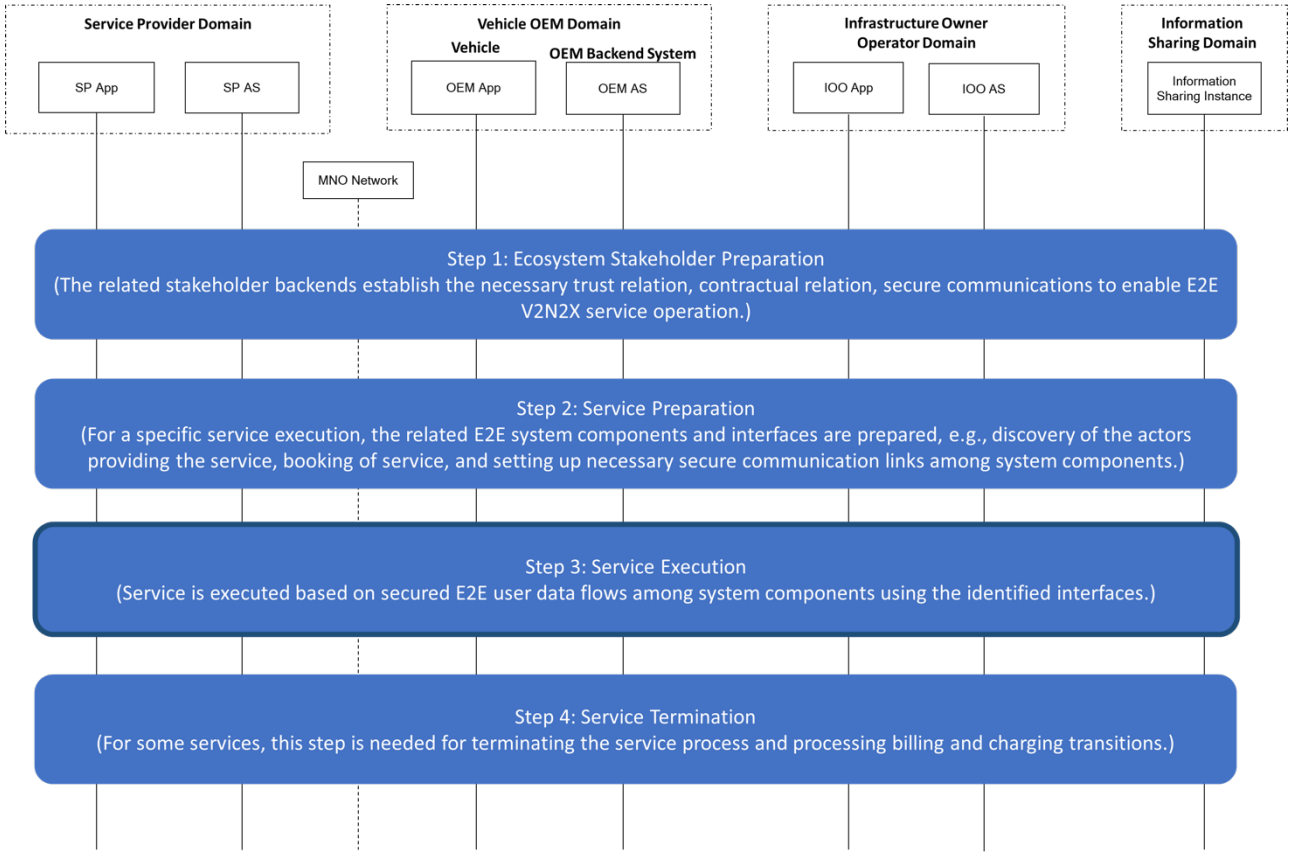


Figure 2: General high-level sequence diagram for E2E V2X service

Step 1 – Ecosystem Stakeholder Preparation

This step covers all preparation tasks to be performed by involved ecosystem stakeholders to ensure successful operation before a V2X service is initiated and executed. One important task is to establish trust through business relations, as often as needed, among the stakeholders for the service operation. This can be managed through bilateral or multilateral contractual agreements or through governance functions settled by administration authorities or by industry organisations.

For deployment of V2X services that involve only a limited number of stakeholders, the trust and business relations can be managed through bilateral or multilateral agreements among the involved stakeholders based on existing regulatory frameworks

and standards. As discussed in Section 8, many V2X use cases have already been deployed as commercial services in many regions. These include not only use cases providing information and alerts to human drivers like Traffic Information Sharing and Emergency Vehicle Approaching in the Netherlands, but also automated driving use cases like Automated Valet Parking (AVP) in Germany, described in Section 8.6. Deeper analysis of the V2N2X deployments from the business perspective can be found in the companying 5GAA Technical Report [21].

For V2X services deployment on the open mass market, including the V2X services e.g., for providing information alerts to human drivers and advanced V2X services involving Automated Driving (AD) discussed above, it is particularly important that the system solution is interoperable and scalable regarding the number of involved stakeholders, e.g., Car OEMs, IOOs, and Service Providers, and supported geographic and market regions, e.g., the number of countries, regions, states and cities where the service is operational. To this end, Section 6.4 introduces the Information Sharing Domain for scalable and interoperable service deployment. As explained in Figure 3 of Section 6.4 with the example from Annex B.1, the Information Sharing Domain also requires the Ecosystem Preparation step consisting of the Governance and Ecosystem initialisation sub-steps, including but not limited to:

- Framework and governance functions have been set up for the open mass market to ensure the service and all involved system components fulfil functional and performance requirements.
- System components from different stakeholders need to undergo the necessary verification processes to demonstrate conformance with the governance framework and technical requirements, e.g.,
  - Security certification of system components and their enrolment in the corresponding Public Key Infrastructure (PKI), known as the security bootstrap process.
  - Conformance and interoperability test of the communicating system components from different stakeholders.

Step 2 – Service Preparation

Upon the initiation or enquiry from the service user, many V2X services need to perform Service Preparation tasks before the (dynamic) user data can be communicated among the V2X applications. Service discovery, service reservation or booking, preparation of communication channels by means of discovering server addresses and exchanging digital certificates are examples of tasks in this step. If it is needed, tasks related to payment are also prepared in this step, to be ready for the payment and billing task in Step 4 Service Termination. The Information Sharing Domain may also provide scalable solutions for Step 2 – Service Preparation, as explained in Section 6.4.

Step 3 – Service Execution

In this step, V2X applications exchange (dynamic) user data via selected interfaces in the E2E system architecture, to realise the service functions and deliver values to the service users. The present Technical Report explains the details of this step for selected use cases in Section 8.

The details of Step 1 Ecosystem Preparation and Step 2 Service Preparation are out of scope of the present Technical Report. In the description of Step 3 – Service Execution for the selected use cases in Section 8, it is assumed that all required tasks in Step 1 and Step 2 have already been accomplished successfully.

Step 4 – Service Termination

This step terminates the service execution and processes the billing and charging transitions, if these are applicable to the service. The detail of this step is also out of this Technical Report's scope.

6 General service deployment options

This chapter describes different service deployment options from the perspective of stakeholders, namely Vehicle OEMs, Service Providers, Infrastructure Owners and Operators, and Information Sharing Entities, as shown in Figure 1. Descriptions are based on the function of system components and the interfaces identified in that Figure 1, which are regarded as the common building blocks or elements for the implemented E2E solution, using cellular networks for respective use cases (use case groups) described in Chapter 8. When using the system building element described in this chapter in actual implementation of V2X services, the stakeholder needs to keep the following in mind:

- The deployment options described in this chapter are for the solution blueprint using cellular networks to support not one specific use case but rather multiple different use cases sharing similar requirements. Actual solutions, including E2E system architecture, use case processes and data flow, as well as application and facilities layer message and protocol configurations, are described in Chapter 8 for selected use cases.
- Among different deployment options described in this chapter, a stakeholder may need to select one or multiple options related to its domain and discuss with other stakeholders for the overall E2E solution, based on interests and preferences.
- For a given use case there may be multiple E2E solutions, or combination of them, depending on the interests and preferences of involved stakeholders.
- Particularly, for the scalable deployment and interoperability among different E2E solutions of a use case, this section also describes the information sharing solution in the information sharing domain.

6.1 Vehicle OEM perspective

This section provides a description of available general deployment options for vehicle OEMs, including interfaces to OEM Apps, and backend interfaces, as well as criteria for vehicle OEMs to select such interfaces and related deployment options. This section also describes technical details of respective interfaces that are generally applicable for different use cases.

6.1.1 Implementation option using interface “O1” between “OEM App” and “OEM AS”

The O1 interface is often used for control and management traffic between vehicle OEM backend and vehicle. The O1 interface is fully controlled by the vehicle OEM from security and protocol perspectives. The O1 interface can be used for user data communication, if for a given use case the performance requirements can be fulfilled,

e.g., data rate, latency, reliability, as well as security, mobility, and scalability. Vehicle OEM can decide the protocol used over the O1 interface.

Example use case implementation descriptions using O1 can be found in Section 8.1 "Traffic event information sharing" and Section 8.2 "Traffic signal information sharing".

6.1.2 Implementation option using interface "V1" between "OEM App" and "IOO AS"

The V1 interface is an inter-stakeholder interface connecting OEM App with IOO AS, which are respectively in the Vehicle OEM Domain and the IOO Domain. V1 is often used for user data traffic between vehicle OEM App and IOO AS, subject to the agreement between the Vehicle OEM and the IOO. The V1 interface is used for communicating the user data of a given use case, if performance and functional requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, mobility, and scalability.

The precondition to using the V1 interface for user data communication is that Step 1 Ecosystem Preparation for establishing the trust and business relations between the Vehicle OEM and the IOO, and Step 2 Service Preparation for discovering, booking, and initiating the respective service session (as described in Section 5) are successfully accomplished. These steps require negotiation and communication between Vehicle OEM and IOO stakeholders using backend interfaces, i.e., O5 directly between OEM AS and IOO AS, or I1 and I3 through the Information Sharing Entities, as explained in Section 6.4.

The deployment options of OEM AS and IOO AS, e.g., when edge computing is used, may have an impact on the stakeholders' decision whether to use V1 for user data communication. When required by the use case for performance considerations, network features like QoS support, mobility management, etc. may be considered in the E2E system solution.

Example use case implementation descriptions using V1 can be found in Section 8.6 AVP/AVM and Section 8.7 "Object Detection and Sharing".

6.1.3 Implementation option using interface "P4" between "OEM App" and "SP AS"

The P4 interface is an inter-stakeholder interface connecting OEM App with SP AS, which are respectively in the Vehicle OEM Domain and the Service Provider Domain. P4 is often used for user data traffic between vehicle OEM App and SP AS, subject to the agreement between the Vehicle OEM and the Service Provider. The P4 interface is used for communicating the user data of a given use case, if performance and functional requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, mobility, and scalability.

The precondition to using the P4 interface for user data communication is that Step 1 Ecosystem Preparation for establishing the trust and business relations between the Vehicle OEM and the Service Provider and Step 2 Service Preparation for discovering, booking, and initiating the respective service session (as described in Section 5), are successfully accomplished. These steps require negotiation and communication between Vehicle OEM and Service Provider stakeholders using backend interfaces, i.e., O2 directly between OEM AS and SP AS, or I3 and I4 through the Information Sharing Entities, as explained in Section 6.4.

The deployment options of OEM AS and SP AS, e.g., when edge computing is used, may have an impact on the stakeholders' decision whether to use P4 for user data communication. When required by the use case for performance considerations, network features like QoS support, mobility management, etc. may be considered in the E2E system solution.

Example use case implementation descriptions using P4 can be found in Section 8.2 "Traffic signal information sharing", Section 8.5 "HD MAP handling", and Section 8.8 "Vulnerable Road User protection".

6.1.4 Backend interface "O2" between "OEM AS" and "SP AS"

The O2 interface is an inter-stakeholder interface connecting OEM AS with SP AS, which are respectively in Vehicle OEM Domain and the Service Provider Domain. O2 is typically used for communication of management data between the backends of the connected stakeholders and it may be used for user data traffic between vehicle OEM AS and SP AS, subject to the agreement between the Vehicle OEM and the Service Provider. The O2 interface may be used for communicating the user data of a given use case, if performance and functional requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, and scalability.

The O2 interface may be used for communication of management data e.g., for Step 1 Ecosystem Preparation for establishing the trust and business relations between the Vehicle OEM and the Service Provider and for Step 2 Service Preparation for discovering, booking, and initiating the respective service session (as described in Section 5). These steps are prerequisites for communicating any user data between the Vehicle OEM and the Service Provider domains.

A limitation of the O2 interface is that it only connects a specific OEM AS to a specific SP AS. This works for V2X services based on bilateral agreement between the connected Vehicle OEM and Service Provider. However, for service deployment involving many Vehicle OEMs and Service Providers, backend connection using the O2 interface results in a complicated many-to-many topology. To resolve this issue, Section 6.4 introduces the Information Sharing Entities leveraging I3 and I4 interfaces, which are explained in Section 6.1.7 and Section 6.2.7.

6.1.5 Backend interface "O4" between "OEM AS" and "OEM AS"

The O4 interface connects two instances of OEM AS for communicating management and user data for V2X service operation. The two OEM AS instances may belong to the same Vehicle OEM or two different Vehicle OEMs.

If the connected instances of OEM AS belong to the same Vehicle OEM, e.g., for different vehicle brands of the same OEM or for offering services in different regions, the Vehicle OEM has full control on the O4 interface and can decide its usage and technical details.

If the connected instances of OEM AS belong to different Vehicle OEMs, the usage and technical details of O4 need to be agreed among the involved Vehicle OEMs.

6.1.6 Backend interface "O5" between "OEM AS" and "IOO AS"

The O5 interface is an inter-stakeholder interface connecting OEM AS with IOO AS, which respectively belong to the Vehicle OEM Domain and the IOO Domain. O5 is typically used for communication of management data traffic between the backends of the connected stakeholders and it may be used for user data traffic between OEM AS and IOO AS, subject to the agreement between the Vehicle OEM and the IOO. The O5 interface may be used for communicating the user data of a given use case, if performance and functional requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, and scalability.

The O5 interface may be used for communication of management data e.g., for Step 1 Ecosystem Preparation for establishing the trust and business relation between the Vehicle OEM and the IOO and for Step 2 Service Preparation for discovering, booking, and initiating the respective service session, as described in Section 5. These steps are the prerequisites for communicating any user data between the Vehicle OEM and the IOO domains.

A limitation of the O5 interface is that it only connects a specific OEM AS to a specific IOO AS. This works for V2X services based on bilateral agreement between the connected Vehicle OEM and IOO. However, for service deployment involving many Vehicle OEMs and IOO, backend connection using the O5 interface results in a complicated many-to-many topology. To resolve this issue, Section 6.4 introduces the Information Sharing Entities leveraging I1, I3, and I4 interfaces, which are explained in Section 6.3.7, Section 6.1.7, and Section 6.2.7.

6.1.7 Backend interface "I3" between "OEM AS" and "Information Sharing Instance"

The I3 interface is an inter-stakeholder interface interconnecting the Vehicle OEM Domain and the Information Sharing Domain. (Details of information sharing, protocols used, etc. are further described in 6.4 Information sharing for scalable and interoperable.) This interconnection using I3 thus provides a common interface and alleviates the need to establish and maintain a multitude of connections between all parties that should exchange information. It is mainly to be used for event data sharing between vehicle OEM AS and other stakeholders in an interconnected ecosystem, subject to agreement made between Vehicle OEM and other stakeholders in the ecosystem.

The I3 interface commonly uses a message queuing protocol, where an Information Sharing Instance can publish data, and an OEM AS can subscribe to information of interest that is published by other actors.

The precondition for using the I3 interface to share data is that Step 1 Ecosystem Preparation and Step 2 Service Preparation, as described Section 5, are successfully accomplished.

6.2 Service Provider perspective

6.2.1 Implementation option using interface "P1" between "SP App" and "SP AS"

The P1 interface is used for user data, as well as for control and management traffic between SP AS and SP App. The P1 interface is fully controlled by the Service Provider from security and protocol perspectives. The P1 interface is used for user data communication, if for a given use case the performance requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, mobility, and scalability. The Service Provider can decide the protocol used over the P1 interface.

Example use case implementation descriptions using P1 can be found in Section 8.2 "Traffic signal information sharing", Section 8.3 "Traffic signal priority request sharing", Section 8.4 "Emergency Vehicle Approaching", Section 8.7 "Object Detection and Sharing", and Section 8.8 "Vulnerable Road User protection".

6.2.2 Implementation option using interface "V1'" between "SP App" and "IOO AS"

The V1' interface is an inter-stakeholder interface connecting SP App with IOO AS, which respectively belong to the Service Provider Domain and the IOO Domain. V1' is often applied to user data traffic between vehicle SP App and IOO AS, subject to the agreement between the Service Provider and the IOO. The V1' interface is used for communicating the user data of a given use case, if performance and functional requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, mobility, and scalability.

The precondition to using the V1' interface for user data communication is that Step 1 Ecosystem Preparation for establishing the trust and business relations between the Service Provider and the IOO and Step 2 Service Preparation for discovering, booking, and initiating the respective service session (as described in Chapter 5) are successfully accomplished. These steps require negotiation and communication between Service Provider and IOO stakeholders using backend interfaces, i.e., P3 directly between SP AS and IOO AS, or I1 and I4 through the Information Sharing Entities, as explained in Section 6.4.

The deployment options of SP AS and IOO AS, e.g., when edge computing is used, may have an impact on the stakeholders' decision whether to use V1' for user data communication. When required by the use case for performance considerations, network features like QoS support, mobility management, etc. may be considered in the E2E system solution.

Example use case implementation descriptions using V1' can be found in Section 8.7 "Object Detection and Sharing".

6.2.3 Implementation option using interface "P4" between "OEM App" and "SP AS"

The P4 interface is described in Section 6.1.3.

6.2.4 Backend interface "P2" between "SP AS" and "SP AS"

The P2 interface connects two instances of SP AS for communicating management and user data for V2X service operation. The two SP AS instances may belong to the same Service Provider or two different Service Providers.

If the connected instances of SP AS belong to the same Service Provider, e.g., for different applications of the same use case (e.g., see Section 8.8 VRU use case) or for offering services in different regions, the Service Provider has full control on the P2 interface and can decide its usage and technical details.

If the connected instances of SP AS belong to different Vehicle OEMs, the usage and technical details of P2 need to be agreed among the involved Service Providers.

6.2.5 Backend interface "P3" between "SP AS" and "IOO AS"

The P3 interface is an inter-stakeholder interface connecting SP AS with IOO AS, which respectively belong to the Service Provider Domain and the IOO Domain. P3 is typically used for communication of management data between the backends of the connected stakeholders and it may be used for user data traffic between SP AS and IOO AS, subject to the agreement between the Service Provider and the IOO. The P3 interface may be used for communicating the user data of a given use case, if performance and functional requirements can be fulfilled, e.g., data rate, latency, reliability, as well as security, mobility, and scalability.

The P3 interface may be used for communication of management data e.g., for Step 1 Ecosystem Preparation for establishing the trust and business relations between the Service Provider and the IOO and for Step 2 Service Preparation for discovering, booking, and initiating the respective service session, as described in Chapter 5. These steps are prerequisites for communicating any user data between the Service Provider and the IOO domains.

A limitation of the P3 interface is that it only connects a specific SP AS to a specific IOO AS. This works for V2X services based on bilateral agreement between the connected Service Provider and IOO. However, for service deployment involving many Service Providers and IOOs, backend connection using the P3 interface results in a complicated many-to-many topology. To resolve this issue, Section 6.4 introduces the Information Sharing Entities leveraging I1, I3, and I4 interfaces, which are explained in Section 6.3.7, Section 6.1.7, and Section 6.2.7.

6.2.6 Backend interface "O2" between "OEM AS" and "SP AS"

The O2 interface is described in Section 6.1.4.

6.2.7 Backend interface "I4" between "SP AS" and "Information Sharing Instance"

The I4 interface is an inter-stakeholder interface interconnecting the Service Provider Domain and the Information Sharing Domain. (Details of information sharing, protocols used, etc. are further described in 6.4 Information sharing for scalable and interoperable.) This interconnection using I4 thus provides a common interface and alleviates the need to establish and maintain a multitude of connections between all parties that should exchange information. It is mainly to be used for event data sharing between SP AS and other stakeholders in an interconnected ecosystem, subject to agreement made between Service Provider and other stakeholders in the ecosystem.

The I4 interface commonly uses a message queuing protocol, where an Information Sharing Instance can publish data, and a SP AS can subscribe to information of interest that is published by other actors.

The precondition to using the I4 interface for data sharing is that Step 1 Ecosystem Preparation and Step 2 Service Preparation, as described Chapter 5, are successfully accomplished.

6.3 IOO perspective

6.3.1 Implementation option using interface "V1" between "OEM App" and "IOO AS"

The V1 interface is described in Section 6.1.2.

6.3.2 Implementation option using interface "V1'" between "SP App" and "IOO AS"

The V1' interface is described in Section 6.2.2.

6.3.3 Implementation option using interface "R1'" between "IOO App" and "IOO AS"

The R1 interface connects the IOO AS and IOO App within the same IOO domain. The IOO App is the system component implemented at the infrastructure owned by the IOO, e.g., road-side equipment including road traffic light controllers, variable electrified message signs, and sensors, or parking facilities. Using the R1 interface the IOO AS can control and manage IOO Apps, as well as send and receive data of V2X application to and from IOO Apps. Such data can be used by IOO AS to provide V2X services to other ecosystem stakeholders, e.g., to OEM App over the V1 interface or to SP App via the V1' interface. Technical details of the R1 interface are decided by the IOO domain owner. The R1 interface can be implemented using mobile network or wired communication, or combination of both.

6.3.4 Backend interface "V2" between "IOO AS" and "IOO AS"

The V2 interface connects two instances of IOO AS for communicating management and user data for V2X service operation. The two IOO AS instances may belong to the same IOO or two different IOOs.

If the connected instances of IOO AS belong to the same IOO (e.g., road traffic authority), e.g., for offering services in different regions, the IOO has full control on the V2 interface and can decide its usage and technical details.

If the connected instances of IOO AS belong to different IOOs (e.g., road traffic authorities of different countries), the usage and technical details of V2 need to be agreed among the involved IOOs.

6.3.5 Backend interface "O5" between "OEM AS" and "IOO AS"

The O5 interface is described in Section 6.1.6.

6.3.6 Backend interface "P3" between "SP AS" and "IOO AS"

The P3 interface is described in Section 6.2.5.

6.3.7 Backend interface "I1" between "IOO AS" and "Information Sharing Instance"

The I1 interface is an inter-stakeholder interface interconnecting the IOO Domain and the Information Sharing Domain. (Details of information sharing, protocols used, etc. are further described in Section 6.4 Information sharing for scalable and interoperable.) This interconnection using I1 thus provides a common interface and alleviates the need to establish and maintain a multitude of connections between all parties that should exchange information. It is mainly to be used for event data sharing between vehicle IOO AS and other stakeholders in an interconnected ecosystem, subject to agreement made between the IOO (e.g., road traffic authority) and other stakeholders in the ecosystem.

The I1 interface commonly uses a message queuing protocol, where an Information Sharing Instance can publish data, and a IOO AS can subscribe to information of interest that is published by other actors or publish information for interests of other actors.

6.4 Information sharing for scalable and interoperable deployment

When the ecosystem scales up and involves multiple actors, there is a need to use Information Sharing Entities, e.g., to avoid a full mesh of connectivity among actors. This section describes some market approaches to achieve this.

Information Sharing Instances operate within the context of the Information Sharing Domain and Information Sharing Entity function, to efficiently exchange data and re-direct connections, and host interfaces between "Information Sharing Entities" and other stakeholders' backend. These interfaces enable communication and interaction, enabling scalable connectivity without the need for a full mesh among actors.

Additionally, it can participate in authorisation and security-related authentication processes for service execution. Depending on the result, it can determine whether to process, reject, or suggest alternative service(s) to the requesting entity.

Information Sharing Instances can monitor and manage information/data intended for services. They can be classified and provided based on specific attributes like position or service type. If needed, information/data tailored to the situation of the data user system can be recommended for service, or information/data matching results based on the user system status can be delivered.

The Information Sharing Domain constitutes a dedicated B2B data sharing trust domain, linking IT backends of clearly identified Information Sharing Instances.

For a larger ecosystem, especially comprising many Information Sharing Entities, governance mechanisms are required, as indicated by the dashed boxes across the top in Figure 3 below. **Governance** would for example comprise a "governing body" that sets the rules (e.g., a framework for data sharing, data quality, privacy, and security). It provides the financial framework and defines an operational CCoC reflecting the public interest in the cross-stakeholder V2X information sharing.

Only those ecosystem stakeholders agreeing to a CCoC for information sharing, -retrieval and -usage, and committed to behaving according to the CCoC principles,

should be allowed to access the Information Sharing Domain and integrate their IT systems with an Information Sharing Instance.

Upon confirmation of compliance, an ecosystem actor will receive a digital certificate and become an authorised V2N2X actor. Having signed the CCoC, a system function linking the validation of a joining actor to a digital certificate for that actor is part of the "**ecosystem initialisation**" functions, indicated by the second horizontal dashed box in Figure 3 below.

Key functions in an Information Sharing Domain comprise, for example, data exchange, databases for static or semi-static data, information about system status, operation and data-quality monitoring, including alert management, and information about internal operational events in the system. Key functions should also comprise support for the validation and logging of shared information to facilitate traceability in adhering to CCoC and quality agreements, e.g., to be able to identify malfunction or misbehaving components or systems.

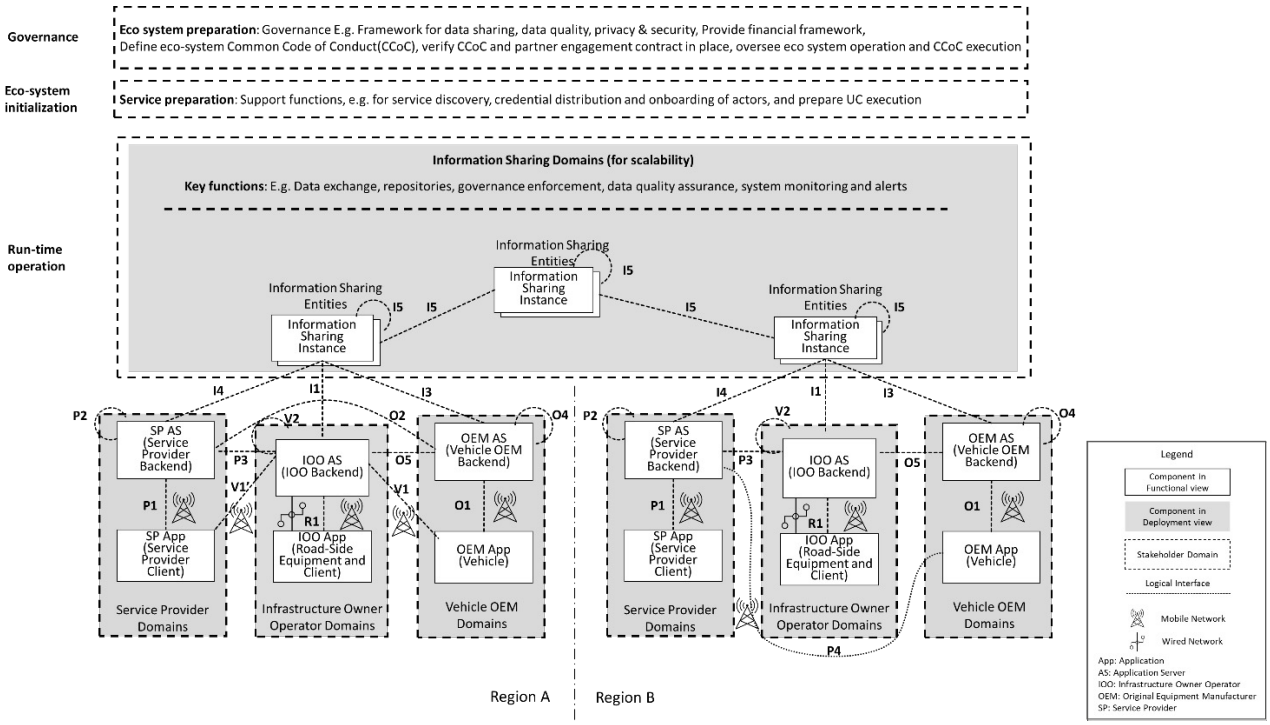


Figure 3: "Information Sharing Entities" provides service for cross-stakeholder information sharing

6.4.1 Interfaces I1, I3, I4, I5 between "Information Sharing Entities" and other stakeholders' backend

'Information Sharing Entities' are used to share information and interact in a scalable way, i.e., no full mesh among actors needed (using the direct interfaces P2, P3, O2, O4, O5, V2). Instead, actors are generally connected to at least one Information Sharing Instance, e.g., in one country or region, which is then interconnected with

Information Sharing Instances in other countries or regions. (Note: There can be more than one Information Sharing Instance per country or region depending on topology, organisations, load, etc.)

The Information Sharing Domain consists of several interconnected Information Sharing Instances, utilising the I5 interface(s). Different topologies can be considered depending on the nature of data sharing, as well as the deployment and operational ambitions. Different ecosystem stakeholders connect (via the I1, I3, I4 interfaces) to at least one instance of the networked Data Sharing Domain, ensuring operational scalability and resilience of the Information Sharing Domain.

The network of interconnected Information Sharing Instances thus provides a federated information sharing backbone via I5 interface, where information from the whole ecosystem is available wherever an actor is connected. (Note: An actor can be redirected to an Information Sharing Instance closer to the data source, e.g., to shorten the data path). This federated information sharing backbone network must provide information through standardised data specifications and methods to realise stable services and business models, and security of the communication network must be secured. In addition, data reliability must be secured, and a quality management system must be established.

To ensure scalability for the information sharing, a protocol providing publish/subscribe methods is needed, the commonly available ones are Advanced Messaging Queuing Protocol (as previously shortened to AMQP) and Message Queuing Telemetry Transport (MQTT). AMQP is a suitable protocol because it is rich in capabilities e.g., for filtering, and especially because communication in this Information Sharing Domain between backend systems is not bandwidth constrained. The MQTT protocol is more suited to simple devices with limited capabilities and bandwidth constrained networks; MQTT is more applicable for communication between backend systems and end clients, e.g., vehicles and smartphones, and would as such add an additional scalability layer.

In Annex B1, the C-Roads implementation of Information Sharing Domain and the related interfaces are explained in detail. In Annex B2, the operational Talking Traffic solution is described and in Annex B3 the Mobilidata solutions that build on the C-Roads model are described. For more about AMQP, see Annex H.

6.4.2 Security and privacy

As described earlier in Section 6.4, once an actor has signed contracts, agreed to CCoC, passed validation, etc., and has been approved to join the Information Sharing Domain as a producer/consumer or as an Information Sharing Entity, the governing body should issue the actor a X509[10] certificate(s) to be used to secure communication and for actor identification. The certificates thus allow for mutual authentication and TLS connections, i.e., TLS connections on I1, I3, I4 interfaces between information consumers/producers and Information Sharing Instances and TLS connection on I5 interface between Information Sharing Instances. To simplify the integration of actors and provide a flexible way to connect Information Sharing Instances and actors, a limited number of trust roots should be used, i.e., only a few root Certificate Authorities (CAs) should be in the actors' trust list. Furthermore, for scalability and operational reasons, intermediate CAs may be used to issue and distribute the actual certificates.

[10] https://en.wikipedia.org/wiki/X.509

Depending on the “trust model” agreed to be used, the certificates may also be used for signing shared information to help trace the originator, or trust may be based on agreements among approved actors, adding actor identification to information shared, applying validation and logging of shared information etc. to further ensure traceability. If a solution with CAs and PKI for distributing IEEE 1609.2 or ETSI TS 103 097 certificates is in place, such a solution could also be leveraged to provide X509 certificates and provide a common “trust anchor”.

Privacy should be governed by contracts and agreed CCoC, as described earlier, and complemented with technical measures. For communication within a domain, e.g., between an SP AS and the SP App or between an OEM AS and the OEM App, privacy is protected by security measures subject to the decision of the domain owner – e.g., using TLS connections for integrity and confidentiality to prevent leakage of sensitive private information. In this case, user consent for the AS to handle personal data can be in place as part of user acceptance to access the services.

For communication in the Information Sharing Domain, as described above, secured connections (e.g., based on TLS) are used for I1, I3, I4, I5 interfaces between authorised actors, to ensure the integrity and confidentiality of the communication. Additionally, for the actual information (payload data) conveyed, before an AS transmits any data in the Information Sharing Domain, it should ensure that the data does not contain personal data e.g., by applying data anonymisation methods. This means if the payload contains personal data, e.g., the data is based on received information from an SP App or OEM App, the AS should remove any sensitive private information before transmitting it. If identity information is required by the V2X use case, the AS may use its identification for the anonymised data, e.g., insert a default identifier for the AS. In many cases, an AS improves payload data quality by analysing and fusing multiple inputs from individual SP Apps or OEM Apps. In such cases, it would be normal and common practice for the AS to use its identity to transmit the processed data instead of using individual identification of the SP Apps or OEM Apps.

For V2X use cases requiring two-way communication, e.g., for requesting traffic signal priority and receiving a response, to protect the privacy of the actual requesters, the requesting AS can act as a proxy for the actual requesters. The proxy can allocate temporary identifiers associated with the actual requesters and use the temporary identifiers in the request message. When receiving a response, the AS can map back to the actual requester. In this way, the personal data of the actual requester is protected.

7 Component deployment options

7.1 Application Server and network component deployment options

Multi-access Edge Computing (MEC) is a feature used to reduce latency, i.e. it is where core network and cloud computing capabilities are moved to the "edge" of the network – typically within a Mobile Network Operator, MNO – closer to the customer, reducing the physical distance for communication. Even when multiple MNOs are involved, solutions exist through federated MEC implementations [2]. Furthermore, MEC also simplifies contracting relations as the same entity, usually the MNO, provides connectivity and edge computing/hosting.

While cellular communication infrastructures are the first and foremost foundation of enabling connected vehicle services, the edge computing element must not be neglected, especially for more advanced services. Here, the deployment of regional MEC sites in reasonable proximity to the network edge will become pivotal for completing the enabler infrastructure elements required for advanced services. As a start, regional MEC deployments/sites – i.e., one per region within the respective corridor sections of the involved countries – are more likely due to economic considerations. These can scale by deploying more computing power per MEC site or by deploying more distributed MEC infrastructures in subregions – and the combination of both. If the vehicle/road user is in roaming condition, MEC is used in combination with Local Breakout (LBO). By using LBO the visiting user can benefit from lower latency and better performance since sessions can be terminated locally at the respective MEC. The use of MEC is further described in [2] and [12].

In addition, an MNO can provide QoS support, i.e., priority for sessions with more stringent requirements on latency, bounded latency or throughput, and "Network Slicing" to control resource usage. See Annex E: 3GPP QoS assurance and Network Slicing mechanisms for further details.

In 5G networks, QoS support can be requested and controlled by Network Exposure Function (NEF) interfaces which allow more dynamic interaction. The 5G network "exposes" different Network Services that can be viewed, configured, or modified by authorised Application Service Providers. The NEF interfaces follow the HTTP REST Model, which is widely used in the internet community. 3GPP has standardised a set of mobile network APIs.

The CAMARA initiative [15] provides an abstraction of the network APIs to simplify the use of 3GPP network features, e.g., for "QoS on Demand". By hiding telecommunications complexity behind APIs and making them available across telco networks and countries, CAMARA enables simple and seamless access. CAMARA is an open-source project within the Linux Foundation to define, develop and test the APIs. It works in close collaboration with the GSMA Operator Platform Group to align API requirements and definitions. Harmonisation of APIs is achieved through fast and agile working code with developer-friendly documentation. API definitions and reference implementations

are free to use (under Apache2.0 licence). Currently, more than 25 “hyperscalers”, aggregators, telco operators and vendors are part of CAMARA. [15]

7.2 Deployment options of in-vehicle Application components

There are multiple deployment options of V2X application for the end user, whether a driver or a driving automation system, to use a V2X service in a vehicle. This section describes three types of in-vehicle V2X application deployment options, namely *automotive OEM-controlled App (OEM App)*, *automotive OEM-supported SP App* being installed or interacting with the vehicle, *automotive OEM-independent SP App* on a smartphone or aftermarket device used in the vehicle. Table 1 provides an overview of the three types of in-vehicle V2X application deployment options. It is worth noting that the classification here mainly considers the responsibility split between an automotive OEM and other service providers, rather than the implementation details.

Table 1: Overview of three types of in-vehicle V2X application deployment options

In-Vehicle App Type		Implemented as	End User	Source of Application Data[1] and Functions	Access to In-vehicle Resources
Automotive OEM-controlled App (OEM App) (See Section 7.2.1)		Application or part of vehicle functions implemented or integrated by OEM. (OEM is responsible for the V2X service provided to the end user.)	Machine (e.g., Driving Automation System, ADAS) and/or Human (e.g., Drivers)	OEM (Optionally in collaboration with SP and/or IOO.)	High-level access to essential resources, e.g., vehicle control, vehicle dynamics information, timing and positioning information, computation and power resource, HMI. (Under OEM control.)
Automotive OEM-supported SP App (See Section 7.2.2)	**(Type-A)**	SP App installed in vehicle infotainment system. (SP is responsible for the V2X service provided to the end user.)	Human (e.g., Drivers)	SP[2]	Basic in-Vehicle Information, e.g., timing and positioning information, computation and power resource, HMI. (Via agreed or standardised APIs provided by OEM. See Section 7.2.2.)
	(Type-B)	SP App on end user device connected to vehicle HMI. (SP is responsible for the V2X service provided to the end user.)	Human (e.g., Drivers)	SP	Limited to vehicle HMI (Via standardised interfaces, e.g., Apple CarPlay or Android Auto based mutual certification. See Section 7.2.2.)[3]
Automotive OEM-independent SP App (See Section 7.2.3)		SP App on end user device used in vehicle, e.g., smartphone app, after-market device. (SP is responsible for the V2X service provided to the end user.)	Human (e.g., Drivers)	SP	None[3]

Note:

(1) In the present report, Application Data refers to essential data for an application

to function properly during the V2X service execution (Step 3 described in Chapter 5.). Such information may include timing information, positioning and dynamics information of ego-vehicle/user device and/or other road users/infrastructure, coded or unprocessed static and/or dynamic information about the environment or events, operation instructions or commands, etc.

(2) In this option (OEM-supported SP App), as the SP remains the responsible entity for the V2X service, the application data source is marked as SP, though the data may be obtained via in-vehicle APIs provided by OEMs (according to the agreement between OEM and SP or following related standards.)

(3) In this option (OEM-supported SP App Type-B or OEM-independent SP App), the SP App may use the power supply from the vehicle without specific agreement between the SP and OEM.

7.2.1 Automotive OEM-controlled App (OEM App)

Automotive OEM-controlled App (OEM App) for V2X service is integrated and fully controlled by the automotive OEM. In this case the OEM is responsible for the implementation and for the provided information and service, as shown in Figure 4.

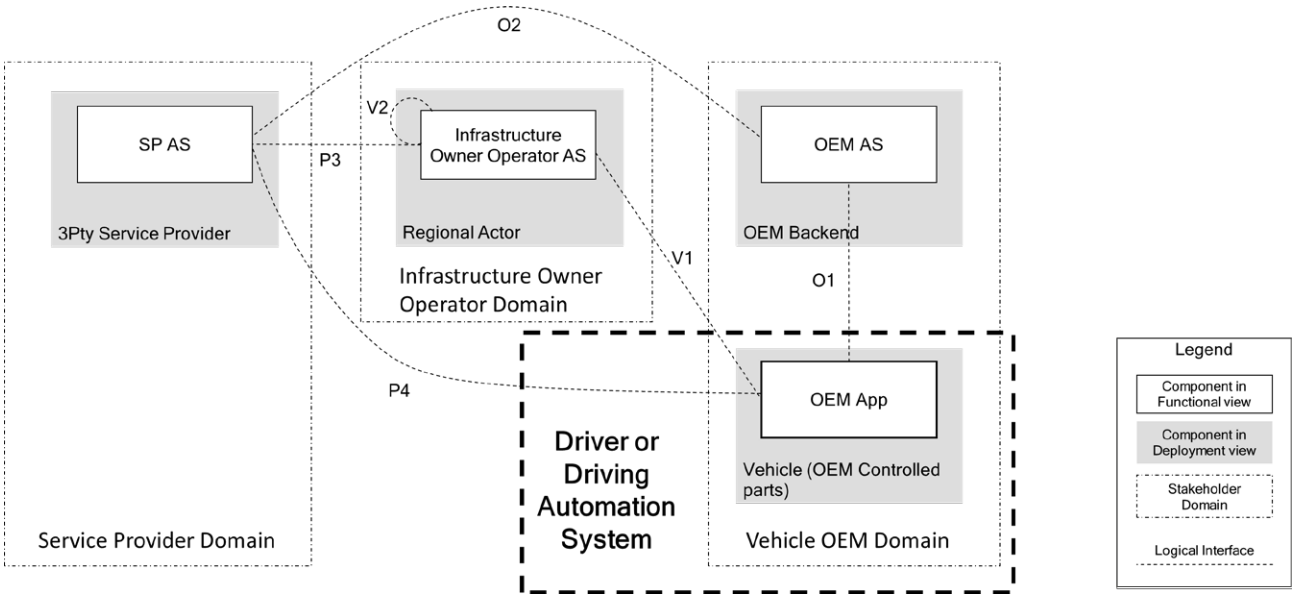


Figure 4: OEM-controlled App (OEM App) (the bold dashed black box indicates the border of the vehicle)

Note: The end user of the OEM-controlled App can be either a human driver via HMI or the driving automation system in the vehicle if the provided information via V2X communications fulfils the requirements of the application.

Note: For some use cases, there is also joint responsibility of vehicle OEM and IOO/SP, e.g., in the AVP use case.

Note: Split SP/OEM service architecture for P4 is possible, i.e., one variant of the

approach, in which an OEM App interconnects with a SP AS over the P4 results in a split in terms of service architecture. In this variant, the OEM App implements the SP's "connection & transport" protocols, message standards and agreed security features. The OEM App can control, to some extent, the types and scope of V2X data that it receives from the SP via a "subscribe" mechanism. The SP sends information messages to the OEM App, which in turn presents the resulting information to the vehicle/driver according to the OEM's own policy, using OEM-specific service logic. The resulting application architecture can be described as "split" because the SP (and its interconnect partners) is responsible for the authenticity and timeliness of the information and the OEM is responsible for the resulting information (warnings etc.) that it presents to the driver or the vehicle (ADAS). Agreements between the SP and the OEM cover the authenticity and accuracy (e.g., GNSS, timeliness, vehicle type, and other information elements) of data generated by the OEM App that will be used in the solution to support agreed V2X use cases.

7.2.2 Automotive OEM-supported SP App

Automotive OEM-Supported Service Provider (SP) App is developed and supplied by a Service Provider, e.g., Waze, Apple Maps, Google Maps. The end user of the automotive OEM-Supported SP App is a human driver. The SP App can be downloaded from the OEM's application store or an OEM's authorised application store, e.g., the official Google Play store. In this second case, the OEM has no control over the application. In its operation, SP Apps utilise certain resources from the vehicle via predefined in-vehicle interfaces or APIs, e.g., computation and power resource, HMI of the vehicle, timing/positioning data, and any other data from vehicles, as allowed by the Software Development Kit (SDK). However, the SP is still the provider and responsible for application data and the functions of the OEM-supported SP App. For this reason, certain authorisation is needed for such SP App to access required vehicle resources and function well, either from the OEM or from the approved App store authority (e.g., from Google).

Depending on the implementation option and required vehicle resources, two sub-categories of Automotive OEM-supported SP App are identified.

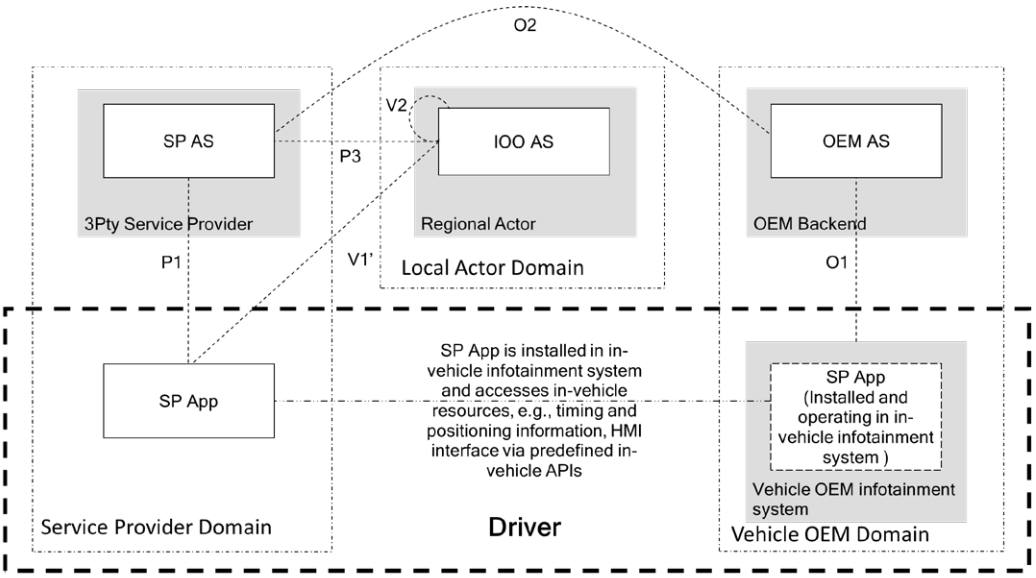
a) Type-A: Automotive OEM-Supported SP App installed in in-vehicle infotainment system or platform:

   Such SP Apps need to come from an OEM-approved app store to be installed and operated in the vehicle's infotainment system, which also provides supporting data, including timing and positioning information, and resources, e.g., HMI, computation, and power, to the SP App via predefined APIs. See Figure 5 (Type-A).

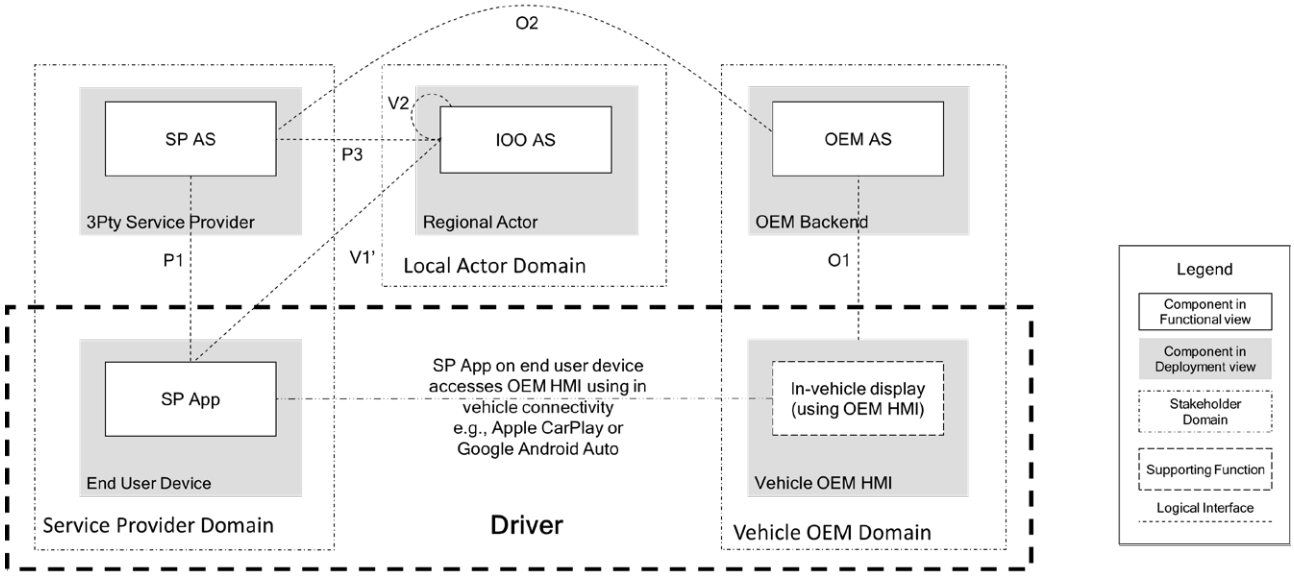
b) Type-B: SP App implemented on a smartphone or portable end user device connected to the vehicle's HMI:

   Such SP Apps operate on smartphones or other portable end user devices and connect to the vehicle's HMI using in-vehicle interface like Apple CarPlay, Android Auto, MirrorLink, as shown in Figure 5 (Type-B). To ensure the user experience of such SP Apps, the smartphone, portable end user device and OEM vehicle HMI system all need to be compliant with certain specifications.

In many cases, the in-vehicle HMI system also needs to be certified to support such SP Apps, e.g., through the Apple MFi (Made For iPhone/iPod/ iPad) programme for CarPlay and Google's certification programme for Android Auto.



Type-A: Automotive OEM-approved SP App installed in in-vehicle infotainment system or platform



Type-B: Automotive OEM-approved SP App implemented on smartphone or portable end user device that connected to the vehicle's HMI

Figure 5: Automotive OEM-supported SP App in vehicle (the bold dashed black box indicates the border of the vehicle)

7.2.3 Automotive OEM-independent SP App

SP Apps on smartphones or aftermarket devices in the vehicle are developed and supplied by a service provider for use in vehicles, as shown in Figure 6. For automotive OEM-independent SP Apps, the service provider is responsible for the provided information, data, and the V2X service to the end user. The vehicle OEMs take no responsibility when such SP Apps are used in vehicles. Such an implementation option in principle does not need access to vehicle resources, except for a power supply, which does not need specific agreement between the SP and the OEM. The end user of automotive OEM-independent SP Apps is the human driver.

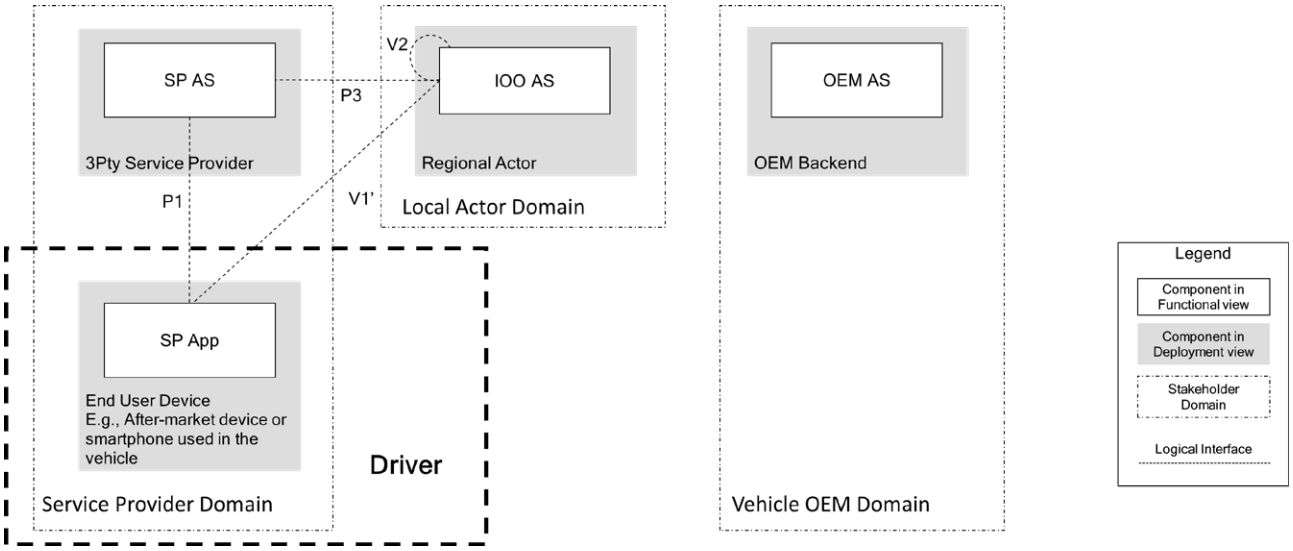


Figure 6: OEM-independent SP App on a smartphone or aftermarket device used in vehicle (the bold dashed black box indicates the border of the vehicle)

8 Use case implementation examples

This chapter provides implementation examples for different V2X use cases using the V2N2X solution blueprint described in Chapter 6. Each implementation example contains the description of the use case, the prerequisites of the implementation, and end-to-end data flow of the service execution step (see Chapter 5) using the deployment option(s) and the information sharing concept described in Chapter 6. It is worth noting that it is not the purpose of this chapter to describe all possible V2N2X implementation options of the selected use cases. For any use case in this chapter the stakeholders are free to use other implementation options than the one(s) described in the example(s) here. However, the implementation examples of different use cases described in this chapter should collectively provide a good overview of all V2N2X service deployment options described in Chapter 6.

8.1 Use case I: Traffic Event Information Sharing

Traffic event information sharing applications allow information sharing between vehicles, between vehicles and other road users, as well as between road infrastructure and vehicles and/or other road users, to improve road safety and traffic efficiency. Examples of traffic event information shared are hazard warnings, such as road works, closed lanes, animal/person on the road, school zone/bus, wrong way driver, broken down vehicle, road works vehicle, slippery road, traffic jam, as well as other road traffic and infrastructure related information such as "in-vehicle information" conveying speed limit information. Use cases (UCs) of traffic event information sharing applications include, but are not limited to:

- UCs described in Clause 6.1.5 of the 5GAA Technical Report "C-V2X Use Cases and Service Level Requirements Volume I" [16],
- UCs in Annex I of the 5GAA White Paper "Updated 2030 Roadmap for Advanced Driving Use Cases, Connectivity Technologies, and Radio Spectrum Needs" [17],
- Sharing of Safety Related Traffic Information (SRTI) in EU,
- UCs deployed in MobiliData programme [18],
- UCs deployed by Virginia Department of Transportation (VDOT) with Audi[11] leveraging the SmarterRoad Open Data Portal from VDOT[12].

[11] Further details about the C-V2X deployment with Audi on Virginia highways are available here.

[12] Further details about the SmarterRoad Open Data Portal from Virginia Department of Transport are available here.

8.1.1 Implementation options

8.1.1.1 Implementation option using interface "O1" between "OEM App" and "OEM AS"

The below architecture is relevant when a limited number of actors share information via interconnected backends. A more scalable solution is described in Section 8.1.2 Scalable deployment using .

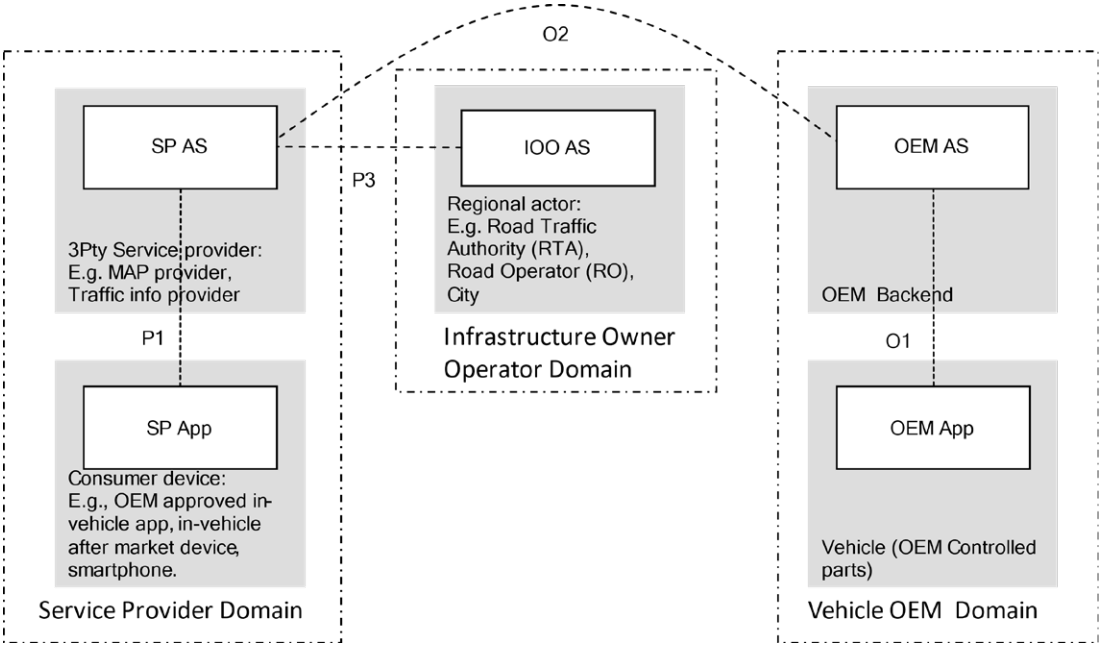


Figure 7: System architecture of traffic event information sharing UC – using O1 interface

Use case deployment solution description

In this UC a Service Provider takes the role to support OEMs and interconnects with the OEM backend to provide traffic event information.

Prerequisites:

A. The SP has established a trust and contractual relationship with the participating OEMs. A secure connection is established between SP and OEM backend, i.e., over the O2 interface.

B. The SP has established trust relations with IOOs and obtains information over the secured interface, i.e. P3.

C. OEM ASs communicate with their vehicles (OEM Apps) over their proprietary interface, i.e., O1, and acts as a proxy/filter if needed.

UC execution alternative 1: OEM AS maintains a digital twin[13]

1. Vehicles (OEM Apps) report their position to OEM backend using O1, assuming contractual relations and methods are already in place to handle regional/local regulation requirements, e.g., for personal data protection.

[13] "Digital twin" here refers to the mechanism for addressing the vehicle clients (OEM Apps) in specific geographical areas, i.e., filtering out irrelevant information and only sharing relevant event information.

2. SP AS constantly obtains traffic event information from IOO ASs using P3, e.g., about roadworks, roadwork vehicles, closed lanes/streets, temporary speed limits.
    1. An information sharing protocol such as Message Queueing Telemetry Transport (MQTT) or Advanced Messaging Queuing Protocol may be used on P3 for IOO ASs to publish events on certain "topics" (message queues) related to certain areas, message types, etc.
    2. Alternatively, specific data format and protocol are available from IOO for SP AS to fetch data from IOO ASs.
3. SP AS informs attached OEM AS (clients) over the O2 interface.
    1. SP AS may run an information sharing protocol such as MQTT or AMQP on the O2 interface and "re-publish" events, which are obtained over P3 from IOO AS, as agreed with the attached OEM AS (clients).
    2. Alternatively, a protocol agreed between SP and OEM can be used for OEM AS to periodically request information from SP AS using the O2 interface. The query is only related to an area, instead of an individual OEM App, to protect the personal data. Or using an agreed protocol over the O2 interface, SP AS may inform OEM AS traffic event information periodically or based on events.
4. OEM AS informs its vehicles (OEM Apps) about relevant event information using the O1 interface.
    1. Information on the O1 interface may be shared using the MQTT protocol (considered more suitable than AMQP over cellular network connectivity).
5. OEM App acts on the received traffic event information, e.g., triggering warning to the driver or visualising the information on the vehicle's HMI.

UC execution alternative 2: SP AS maintains a digital twin

1. Vehicles (OEM Apps) report their position to OEM AS using the O1 interface, assuming the contractual relations and methods are in place to handle regional/local regulation requirements, e.g., for personal data protection.
2. OEM AS periodically reports position of vehicles to SP AS using the O2 interface. To protect personal data, position information shared with SP AS needs to be anonymised by OEM AS to hide the actual identity of the vehicle (OEM App), or SP AS contractually obliged to fulfil personal data protection regulation, like GDPR in Europe.
3. SP AS constantly obtains traffic event information from IOO ASs using the P3 interface, e.g., information about roadworks, roadwork vehicles, closed lanes/streets, temporary speed limits, etc.
    1. An information sharing protocol such as MQTT or AMQP may be used the P3 interface for the IOO AS to publish traffic event information on certain "topics" (message queues) that are related to certain areas, message types, etc.

2. Alternatively, specific data format and protocol are available from IOO for the SP AS to fetch data from IOO ASs.

4. SP AS informs OEM AS about relevant traffic event information using the O2 interface.
    1. In the shared traffic event information, SP AS may provide the 'reference', which OEM AS can use to relay the information to the vehicle.
5. OEM AS informs its vehicles (OEM Apps) about relevant traffic event information using the O1 interface.
6. OEM App acts on the received traffic event information, e.g., triggering warning to the driver or visualising the information on the vehicle's HMI.

Reporting traffic event information from vehicles (OEM Apps) (applicable for both alternative 1 & 2)

1. Vehicle internal sensors detects traffic event information to be reported.
2. Vehicle (OEM App) reports this on the O1 interface, either as part of position report message or using dedicated message.
3. OEM AS forwards information to SP AS in an anonymous way on the O2 interface, if there are agreements between OEM and SP to share such information.
4. SP AS validates the received traffic event information, e.g., by using received information from other sources, before sharing the formation with other entities.

Protocols used

O2 interface: The protocol agreed between SP and OEM, likely based on the SP proprietary protocol used on the P1 interface but with extensions, e.g., for higher security requirements.

P3 interface: The protocol used by IOO and also implemented by SP, e.g., a standardised protocol such as DATEX, ETSI DENM, ETSI IVIM, or SAE BSM Part 2 (for event information).

Note: Procedures for the protocol could be profiled according to the "IP based interface profile" [4] using AMQP, i.e., based on a publish/subscribe model, the SP AS is notified about new information that it subscribes to, as soon as IOO AS publishes new traffic event information on the same topic.

O1 interface: The protocol is OEM proprietary.

Note: Messages communicated over the O1 and O2 interfaces may use existing standards, e.g., using ETSI CAM or SAE BSM for position information and ETSI DENM or BSM Part 2 or DATEX for traffic event information. Compared to local broadcast using short-range communications, the message periodicity over the O1 and O2 interfaces, which use cellular network communication and wired communication, can be variable and lower, e.g., lowered to ~1 per second for CAM. In Annex C: 'Talking Traffic' message frequency profile, message frequency profiles used in the operational "Talking Traffic" deployment is provided for reference.

8.1.2 Scalable deployment using Information Sharing Entities

When the deployment scales up and involves more ecosystem stakeholders, the traffic event information sharing UCs described above will require the use of Information Sharing Entities, e.g., to avoid a full mesh of connectivity among actors. "Information Sharing Concept" and related preparation are further described in Section 6.4 Information sharing for scalable and interoperable

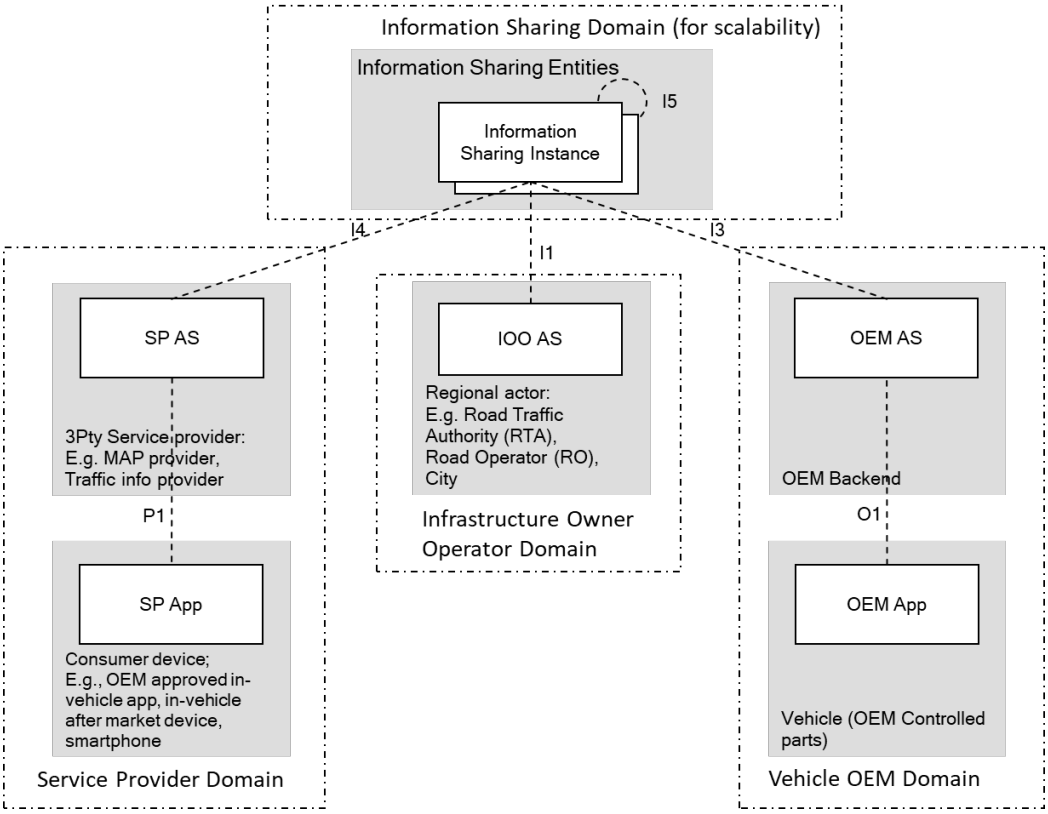


Figure 8: System architecture of traffic event information sharing UC – using Information Sharing Entities

Note: The above figure only shows cross-domain backend interfaces that are relevant to the Information Sharing Entities. Although not shown in the figure, cross-domain backend interfaces based on bilateral agreements can also be used between ecosystem stakeholders, e.g., O2, O5, P3 in Figure 1.

Deployment solution description

In this scenario, "Information Sharing Entities" are used to share traffic event information and interact in a scalable way. The backend of an actor, e.g., vehicle OEM, IOO, or SP, is in general connected to one "Information Sharing Instance", e.g., in one country or region. This Information Sharing Instance is then interconnected with Information Sharing Instances in other countries or regions.

Note: There can be more than one Information Sharing Instance per country or region depending on the system topology, organisations, data traffic load, etc.

The network of interconnected Information Sharing Instances thus provides a federated information sharing backbone, where information from the whole ecosystem is available wherever an actor is connected. (Note: An actor can be redirected to an

Information Sharing Instance closer to the data source, e.g., to shorten the data path).

The following description uses the solution developed by the EU C-Roads platform [4] as an example. In this example, the communication and information exchange is based on the C-Roads "specification" for "IP based interface profile" [4] enabling a publish/subscribe model using AMQP with metadata (AMQP application properties) to allow message filtering based on what an actor is interested in, e.g., location, type of message, etc.

Information exchange UC: As described in Section 6.4 Information sharing for scalable and interoperable , once preparations are in place, i.e., connectivity, publishing agreements and subscription filters have been established, information exchange can be performed.

1. An AS (operating as a "client" in this instance), e.g., SP AS, IOO AS and OEM AS, has identified an event, which is relevant and/or agreed to be shared in the ecosystem and has achieved a level of trustworthiness or quality, e.g., based on reports from several independent sources like vehicles.
2. The AS then anonymises this event information and publishes it to the Information Sharing Instance with accompanying AMQP metadata indicating type of message, location (Country & quadtree tile, see Annex D), producer of the information, etc., as described in the C-Roads "IP based interface profile" [4]. The publication is carried on one of the following I1, I3 or I4 interfaces, which follow the same general approach or with some variances.
3. The receiving Information Sharing Instance checks which AS clients have a matching subscription based on the established filters, and pushes the information to those clients using I1, I3 or I4 interfaces following the same general approach with some variances.

Note: Here the federated Information Sharing Domain facilitated by the I5 interface between Information Sharing Instances is applicable, i.e., Information Sharing Instances connected to another Information Sharing Instance are also informed about the traffic event information that they subscribe to.

4. An AS client receiving the information can thus select to forward this to its relevant clients e.g., OEM Apps or SP Apps depending on their location and heading.
    1. It is assumed that a user consent is in place with the end user and the way the OEM or SP handles personal data is compliant with GDPR.
    2. Alternatively, a user can indicate an "area of interest" to its serving backend, i.e., the OEM AS or SP AS, to mitigate the privacy issue, given the "area of interest" large enough. The drawback of this method is that additional local filtering is needed to receive the relevant information and filter out irrelevant information.

Protocols used

On I1, I3, I4, and I5 interfaces, standard IT technology and processes should be used (see Annex G), e.g., AMQP can be used for information sharing (publish/subscribe) and for providing metadata required in filtering operation (see Annex H) to identify the

payload, relevant area (e.g., based on quadtree tile concept, see Annex D), etc. TLS 1.3 with mutual authentication can be used for security.

Note: The I5 interface is used in scenarios where actors connected to different Information Sharing Instances. In such cases, subscriptions are federated between the Information Sharing Instances.

The payload encapsulated by AMQP can be according to agreed formats among actors (i.e., the transport and information sharing solutions are payload-agnostic). For example, in C-Roads "IP based interface profile" [4] the following ETSI messages formats are supported and encapsulated as AMQP payload: DENM, IVIM, SPATEM, MAPEM, SREM, SSEM, CAM.

Note: Since AMQP is payload-agnostic, SAE messages can be encapsulated, if supporting AMQP metadata are defined and available.

In the Nordic Way project[14], Information Sharing Instances also support DATEX II (with defined metadata) as AMQP payload. For supported messages and protocols in C-Roads, Talking Traffic, and Mobilidata solutions, see Annex B.

8.2 Use Case II: Traffic Signal Information Sharing

Traffic Signal Information (TSI) sharing refers to the exchange of real-time data and information of traffic signals between the IOO, e.g., Advanced Traffic Management System (ATMS), Traffic Light Controllers (TLCs), or other traffic management systems, and vehicles. This sharing of information allows for better coordination and optimisation of traffic flow, leading to improved safety, efficiency, and reduced congestion on the roads.

With Traffic Signal Information sharing, vehicles can obtain TSI about the current signal phase (e.g., green, yellow, or red) and the time remaining until the signal changes (e.g., Time-to-Green, Red Light Countdown). Vehicles supporting the use case can receive this information and use it to adjust their speed and behaviour accordingly (e.g., Green Light Optimal Speed Advisory GLOSA). This UC has been deployed in the Mobilidata[15] and Talking Traffic[16] programmes, which are further described in Annex B.3 and B.2, respectively.

Overall, TSI sharing plays a crucial role in improving traffic management and enhancing the overall efficiency and safety in Intelligent Transportation Systems.

[14] https://www.nordicway.net/services

[15] Mobilidata programme defined 31 traffic solutions in five different categories (intelligent traffic lights, navigation and parking management, risk and hazard notifications, traffic rules notifications and policy support) based on road-vehicle data collection and sharing: https://www.mobilidata.be/en.

[16] Talking Traffic is a successful innovation programme to bring digital infrastructure and connected vehicles to large-scale deployment in The Netherlands, leveraging the existing cellular networks. Talking Traffic use cases include priority/pre-emption for designated road users, leveraging vehicle probe data for improved traffic flow efficiency, and GLOSA/TTG. Talking Traffic website: https://www.talking-traffic.com/nl/.

8.2.1 Implementation options

8.2.1.1 Implementation option using interface "O1" between "OEM App" and "OEM AS"

In this implementation option, the OEM AS provides service directly to the vehicle OEM App. The OEM App is implemented and fully controlled by the vehicle OEM, and it is responsible for the provided information and service. The OEM AS controls the OEM App via O1 and assists with discovery and security functions using O1.

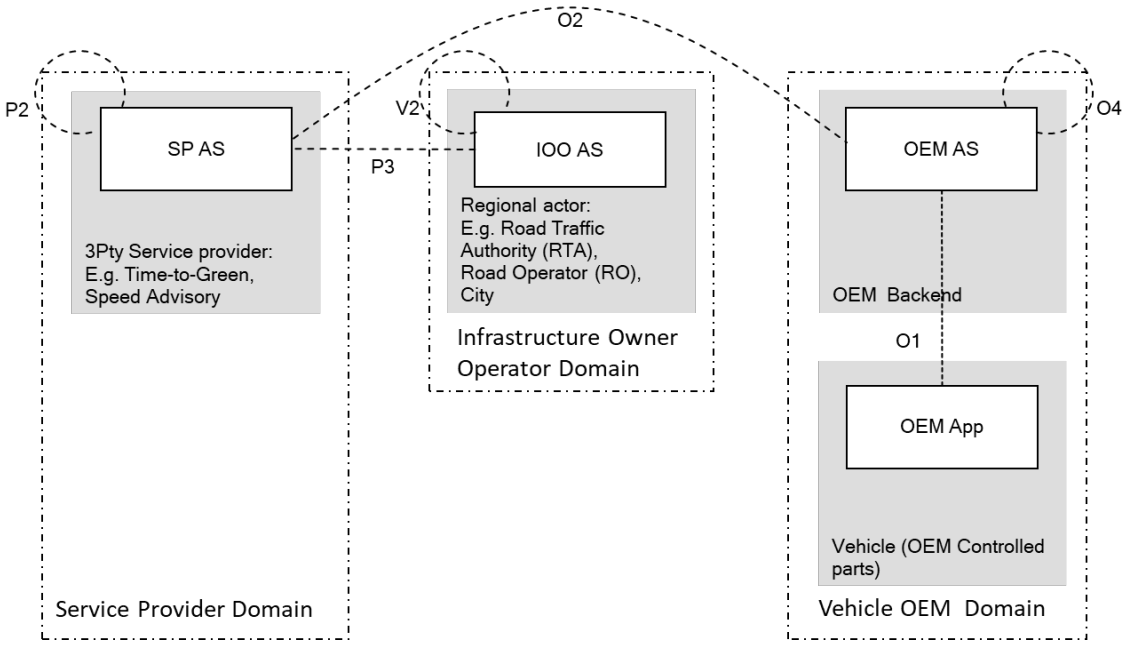


Figure 9: System architecture of Traffic Signal Information sharing UC – OEM AS provides TSI to OEM App over O1 interface

Pre-requisites:

A. IOO license the use of traffic signal status data to SP.

B. SP has made an agreement with OEM to provide TSI and established a secure communication channel between SP AS and OEM AS via the O2 interface.

C. OEM and SP agreed on the service and then inform about where services are available. OEM provides to the vehicles information about the data sources via O1 interface.

UC execution:

1. IOO AS provides real-time data (e.g., SPaT/MAP) to SP AS via P3 interface.

   Note: SP AS may also be licensed to develop SPaT/MAP messages from raw signal status data, intersection drawings and signal timing plan information for each traffic signal location.

2. OEM AS sends SP AS a TSI request with anonymised vehicle ID, heading, manoeuvre and geolocation via O2 interface.
3. SP AS returns to OEM AS the targeted MAP/SPaT message for the specific signal location.
4. OEM AS manages connectivity for delivering MAP/SPaT to OEM App via O1 interface.
5. TSI is displayed to end user via in-vehicle HMI interface.

Protocols used (O1 interface):

For the O1 interface, the protocol and message formats are proprietary to the OEM.

8.2.1.2 Implementation option using interface "P1" between "SP App" and "SP AS"

In this implementation option, the SP AS provides service directly to SP App, e.g., OEM-independent SP App on or OEM-supported SP App, as discussed in Section 7.2.2 and Section 7.2.3. The in-vehicle after-market device is developed and supplied by the SP and it is responsible for the provided information and services.

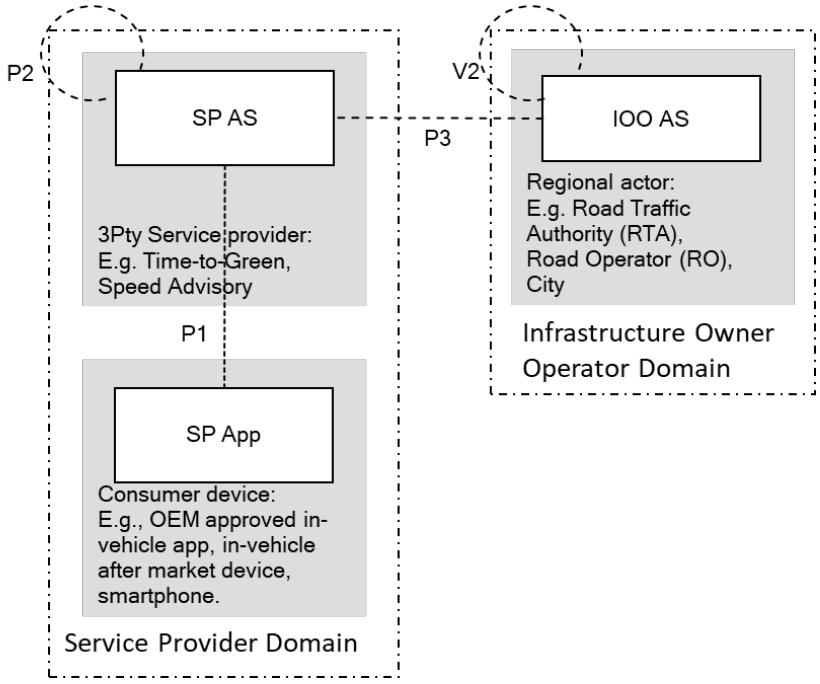


Figure 10: System architecture of Traffic Signal Information sharing UC – SP AS provides TSI to SP App via P1 interface

Note: In this implementation option, the vehicle OEM takes no responsibility for SP App. However, for OEM-supported SP Apps the OEM may have restricted the services provided, e.g., to avoid information that may be conflicting with other Vehicle Information or confusing to the end user.

Pre-requisites:

A. IOO licenses the use of traffic signal status data to SP.

B. SP App is pre-integrated into consumer navigation application and handles geolocation tasks.

C. SP App on smartphone, can potentially be mirrored in the vehicle's multimedia interface (MMI) for OEM-supported SP App. (See Section 7.2.2 Type-B.)

UC execution:

1. IOO AS provides real-time data (e.g., SPaT/MAP) to SP AS via P3 interface.

   Note: SP AS may also be licensed to develop SPaT/MAP messages from raw traffic signal status data, intersection drawings and signal timing plan information for each traffic signal location.

2. SP App sends request to SP AS for TSI service with vehicle ID, vehicle heading, manoeuvre and geolocation – via P1 interface.

   Note: Necessary methods need to be taken to ensure the compliance to the personal data protection regulation in the concerned region, e.g., user consent, anonymity of the vehicle ID.

3. SP AS matches vehicle's location to MAP message, returns targeted SPaT content to SP App via P1 interface.

4. TSI is displayed to end user via consumer smartphone application (for OEM-independent SP App) and/or in-vehicle MMI (for OEM-supported SP App).

Protocols used (P1 interface):

For the P1 interface, the protocols and message formats for the exchange of information are proprietary to the SP. This protocol is applicable for scenarios when "OEM-supported SP App" or OEM-independent SP App are used in vehicle.

8.2.1.3 Implementation option using interface "P4" between "OEM App" and "SP AS"

In this implementation option, the SP AS provides service directly to the vehicle OEM App. The OEM App is implemented and fully controlled by the vehicle OEM, and it is responsible for the provided information and service. The OEM AS controls the OEM App via O1 and assists with discovery and security functions using O1.

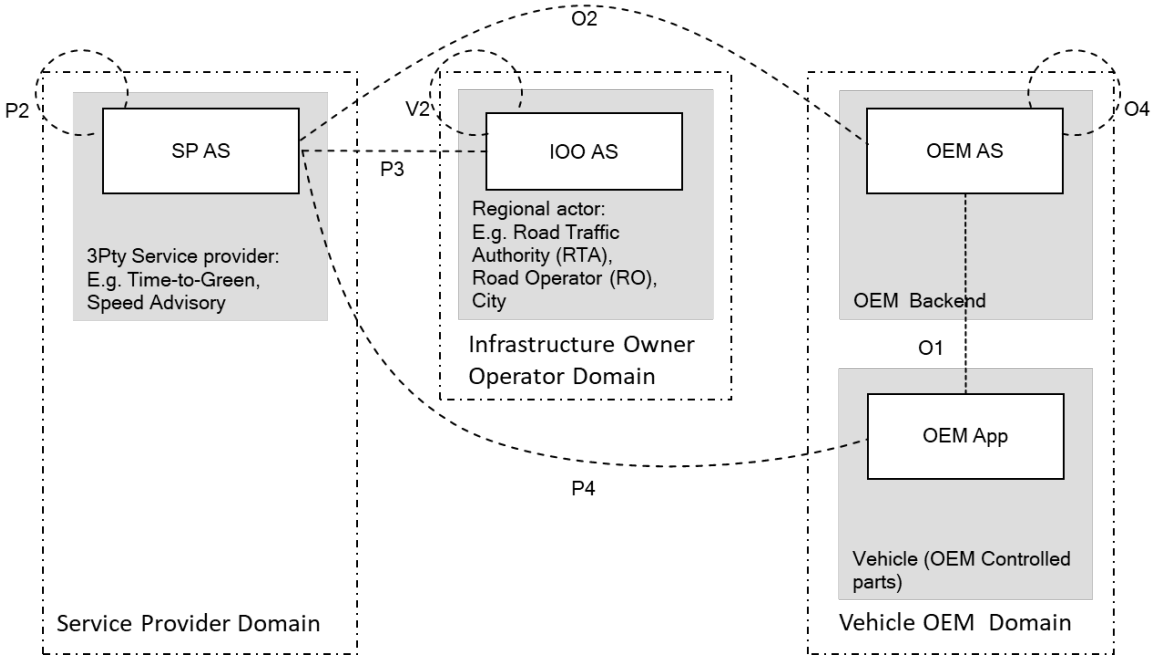


Figure 11: System architecture of Traffic Signal Information sharing UC – SP AS provides TSI to OEM App via P4 interface

Pre-requisites:

A. IOO licenses the use of traffic signal status data to SP.

B. SP has made an agreement with OEM to provide TSI and established a secure communication channel between SP AS and OEM AS via the O2 interface.

C. OEM and SP agree on the service cities and then write the respective service addresses to the vehicles with over-the-air update capabilities via O1 interface.

UC execution:

1. IOO AS provides real-time data (e.g., SPaT/MAP) to SP AS via P3 interface.

   Note: SP AS may also be licensed to develop SPaT/MAP messages from raw signal status data, intersection drawings and signal timing plan information for each traffic signal location.

2. OEM App registers the service with the OEM AS, acquiring the corresponding vehicle ID, via the O1 interface.

3. SP AS authenticates the vehicle ID and starts communicating with OEM App under corresponding address via P4 interface.

   Note: The communication between SP AS and OEM App over P4 interface needs to be secured with the assistance from OEM AS, e.g., for preparing the necessary security certificates.

   Note: Necessary methods need to be taken to ensure the compliance

to the personal data protection regulation in the concerned region, e.g., user consent, anonymity of the vehicle ID.

4. OEM App sends request to SP AS with anonymised vehicle ID, vehicle heading, manoeuvre and geolocation via P4 interface.
5. SP AS matches vehicle's location to MAP message, returns targeted SPaT content to OEM App via P4 interface.
6. TSI is displayed to end user via in-vehicle HMI interface.

Protocols used (P4 interface):

For the P4 interface, the protocol and message formats need to be agreed between SP and OEM. Hence, it is recommended to use the MAP/SPaT message formats that are compliant with the regional ITS standards, e.g., SAE International, ETSI ITS, Chinese Standard YD/T 3709-2020.

8.2.2 Scalable deployment using Information Sharing Entities

The TSI sharing UC described above will, when scaling up and involve several actors, require the use of Information Sharing Entities, e.g., to avoid a full mesh of connectivity among actors. "Information Sharing Concept" and related preparation is further described in Section 6.4 Information sharing for scalable and interoperable deployment t".

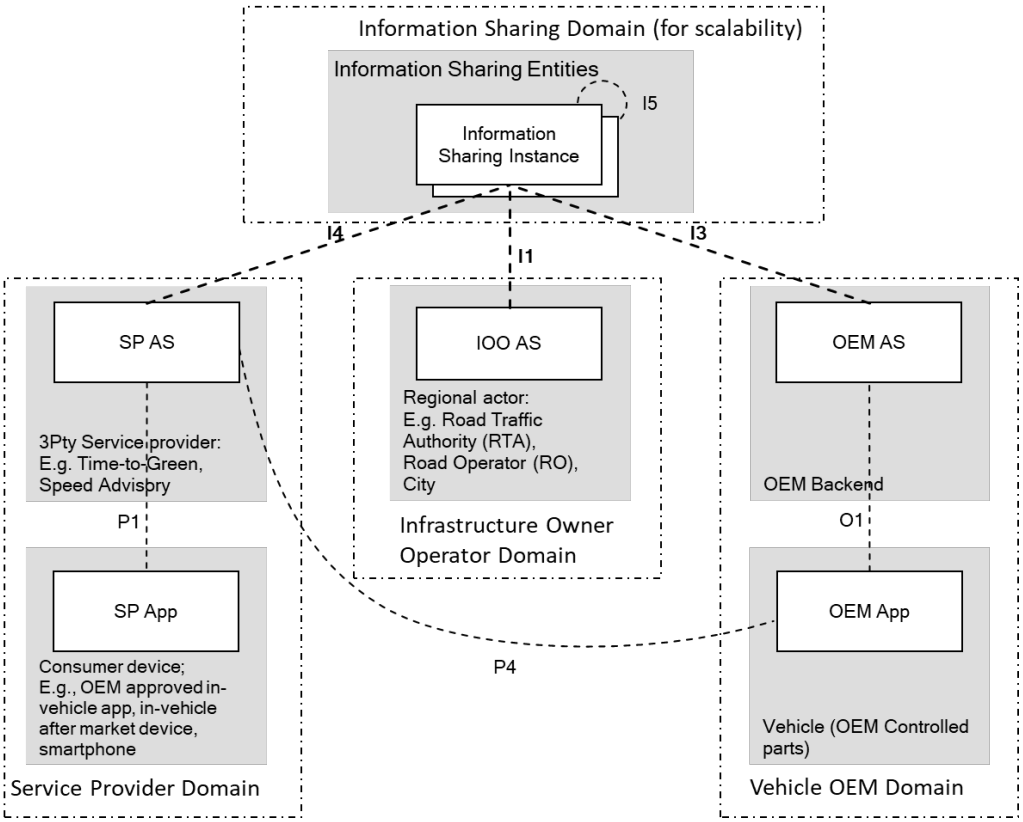


Figure 12: System architecture of Traffic Signal Information sharing UC – using Information Sharing Entities

Note: The above figure only shows cross-domain backend interfaces that are relevant to the Information Sharing Entities. Although not shown in the figure, cross-domain backend interfaces based on bilateral agreements can also be used between ecosystem stakeholders, e.g., O2 and P3 in Figure 1.

The deployment solution using the Information Sharing Entities described in Section 8.1.2 are also applicable to the TSI sharing UC.

Information exchange steps

As described in Section 6.4 Information sharing for scalable and interoperable ", once preparations are in place, i.e., connectivity, publishing agreements and subscription filters have been established, information exchange can be performed.

1. The IOO AS(s) that provide TSI in the ecosystem and achieved a level of trustworthiness, e.g., from the road authority, publishes it to the "Information Sharing Instance" with accompanying AMQP metadata indicating the type of message (e.g., MAP/SPaT), location (Country & quadtree tile, see Annex D), producer of the information, etc., as described in the C-Roads "IP based interface profile" [4]. The publication is carried on the I1interfaces.
2. The receiving Information Sharing Instance checks which AS clients, e.g., SP AS(s), have a matching subscription based on the established filters, and pushes the information to those clients using the I4 interface.

   Note: here the federated Information Sharing Domain facilitated by the I5 interface between Information Sharing Instances is applicable, i.e., Information Sharing Instances connected to another Information Sharing Instance are also informed about the traffic event information that they subscribe to.
3. An AS client, e.g., SP AS, receiving the TSI can thus select to forward this to its relevant clients e.g., SP Apps (via P1 as described in Section 8.2.2) or OEM Apps (via P4 as described in Section 8.2.3), depending on their location and heading.

   Note: It is assumed that a user consent is in place with the end user and the way the OEM or SP handles personal data is compliant with GDPR.
4. As an alternative to step 3.), if the TSI use case is implemented using option O1, as described in Section 8.2.1, the AS client, i.e., the SP AS, receiving the TSI can thus forward this to its relevant clients, i.e., OEM AS(s) via O2. The OEM AS then forward the TSI to its connected OEM App(s), as described in Section 8.1.2.

Protocol used

When the AMQP protocol is used for information sharing over the I1, and I4 interfaces, as described in Section 8.1.2, the payload message formats for the TSI sharing UC should be MAP/SPaT that are compliant with the regional ITS standards, e.g., SAE International, ETSI ITS, Chinese Standard YD/T 3709-2020.

Considerations on message format, profiling, and security of TSI sharing UC

MAP and SPaT messages can be delivered over public cellular networks (Uu interface), leveraging the 4G/5G network ability to unicast the messages directly to specific

vehicles.

The unicast method has several advantages, such as:

- Vehicle authentication as a prerequisite can be done flexibly for not only identity checks but also serviceability validation.
- Add-on features like flexible geofencing techniques to derive targeted applications and reduce workload.

Deployment considerations of TSI sharing UC

TSI sharing can be deployed from either Public Cloud or Multi-access Edge Computing (MEC) servers via the public cellular network to vehicles or mobile devices (see Section 7.1.). With the combination of MEC and 4G LTE/5G networks, this use case shall be able to guarantee low-latency message transmission within 100ms to support time-sensitive applications [14]. For now, the TSI UC is to provide information to the human driver, as supplementary information rather than replacing the primary optical traffic signal at intersections. Such TSI, when used in GLOSA application, can improve the overall traffic efficiency.

Summary of V2N2X implementation for TSI sharing UC

In summary, the proposed V2N2X reference architecture provides a blueprint to support a wide range of traffic information sharing use cases, implemented via multiple (logical) interfaces across multiple stakeholder domains – with some already in live commercial operation (e.g. Talking Traffic[17], Audi's Traffic Light Information[18]). These can be deployed by both local actors (e.g., city/municipality) and regional actors (e.g., road authorities) – based on the data accessibility needs and governance data structure to enable scalable (federated) deployments. By adopting the traffic information sharing approach depicted in this section, the various cross-sector ecosystem partners will have the foundation to implement a technically feasible service/solution that adheres to V2N2X use case best practices.

8.3 Use case III: Traffic Signal Priority Request

This use case allows vehicles to request priority of traffic signal using bidirectional communication with traffic control backend. In this UC a Service Provider (SP), for example a fleet operator, has made arrangements and agreements to request traffic signal priority for the operated vehicles, e.g., to better optimise traffic flow for public transport or heavy vehicles. This UC has been deployed in the Mobilidata[19] and Talking Traffic[20] programmes, which are further described in Annex B.3 and B.2.

[17] https://dmi-ecosysteem.nl/en/theme-page-urban-traffic/talking-traffic/
[18] https://media.audiusa.com/en-us/releases/412
[19] Mobilidata programme defined 31 traffic solutions in five different categories (intelligent traffic lights, navigation and parking management, risk and hazard notifications, traffic rules notifications and policy support) based on road-vehicle data collection and sharing https://www.mobilidata.be/en.
[20] Talking Traffic is a successful innovation program to bring digital infrastructure and connected vehicles to large-scale deployment in The Netherlands, leveraging the existing cellular networks. Talking Traffic use cases include: Priority/pre-emption for designated road users, leveraging vehicle probe data for improved traffic flow efficiency, and GLOSA/TTG. Talking Traffic website: https://dmi-ecosysteem.nl/en/theme-page-urban-traffic/talking-traffic/.

8.3.1 Implementation options

8.3.1.1 Implementation option using interface "P1" between "SP App" and "SP AS"

The below architecture for interacting UCs is applicable when there is a limited number of interacting actors. A more scalable solution is described in Section 8.3.2 for scenarios with large number of interacting actors.

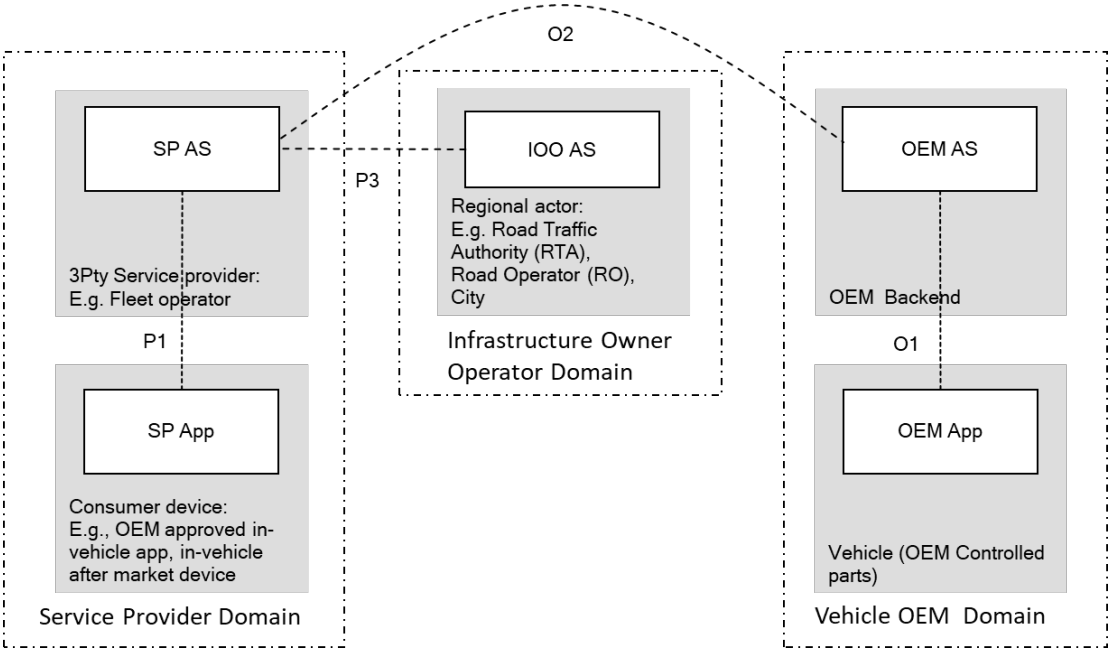


Figure 13: System architecture of traffic signal priority request UC – using P3 interface

Prerequisites:

A. The SP AS, e.g., from a public transport operator or an ambulance operator, has access to Vehicle Information, e.g., location, direction, speed.

   1. If the OEM AS and the OEM-controlled App (OEM App) (see Section 7.2.1) are used to obtain Vehicle Information and the SP has established trust and contractual relations with the participating OEMs, a secure connection is established between SP AS and OEM AS, i.e., over the O2 interface. In this scenario it is also assumed that OEM ASs communicate with their vehicles (OEM Apps) over their proprietary interface O1 and act as proxy/filter for OEM Apps.
   2. If the SP App is located in the vehicle and implemented as OEM-supported SP App (see Section 7.2.2) or OEM-independent SP App on aftermarket device (see Section 7.2.3), the SP App can provide "Vehicle Information".

B. The SP has established trust relations with IOOs and have permission to request traffic signal priority over a secured interface, i.e., P3.

UC execution:

1. The SP AS periodically obtains Vehicle Information from its vehicles, e.g., location, heading and speed. The interface for obtaining Vehicle Information depends on in-vehicle deployment: via P1 or O2. In the latter case, the OEM AS obtains Vehicle Information via O1 interface.
2. The SP AS maintains information about traffic signals that allow priority request, including their identifiers, location, etc. Such information is received via P3 from entity managing the traffic signals, e.g., using MAP messages with topology information.
3. When a vehicle approaches an intersection, the SP AS requests priority by sending a SREM message via P3 to the entity managing the traffic signals, if needed, e.g., when an ambulance has blue light on.
4. If the traffic signal priority can be granted, the entity managing the traffic signals switches traffic signal state and replies with a SSEM message via P3.

Protocols used

Traffic signal priority request use case may use SREM/SSEM messages defined in ETSI ITS at the application (also known as ITS Facilities) layer, or other messages defined in other regional SDOs. For the SP AS to obtain periodical Vehicle Information update, CAM defined in ETSI ITS at the application layer, or other messages defined in other regional SDOs, can be used. See Annex C “Talking Traffic” message frequency profile for cellular network implementation.

8.3.2 Scalable deployment using Information Sharing Entities

When the deployment scales up and involves more ecosystem stakeholders, the UC described above will require the use of Information Sharing Entities, e.g., to avoid a full mesh of connectivity among actors. Once preparations, as described in Section 6.4 Information sharing for scalable and interoperable , are in place, e.g., connectivity, publishing agreements and subscription filters are established, information exchange can be performed.

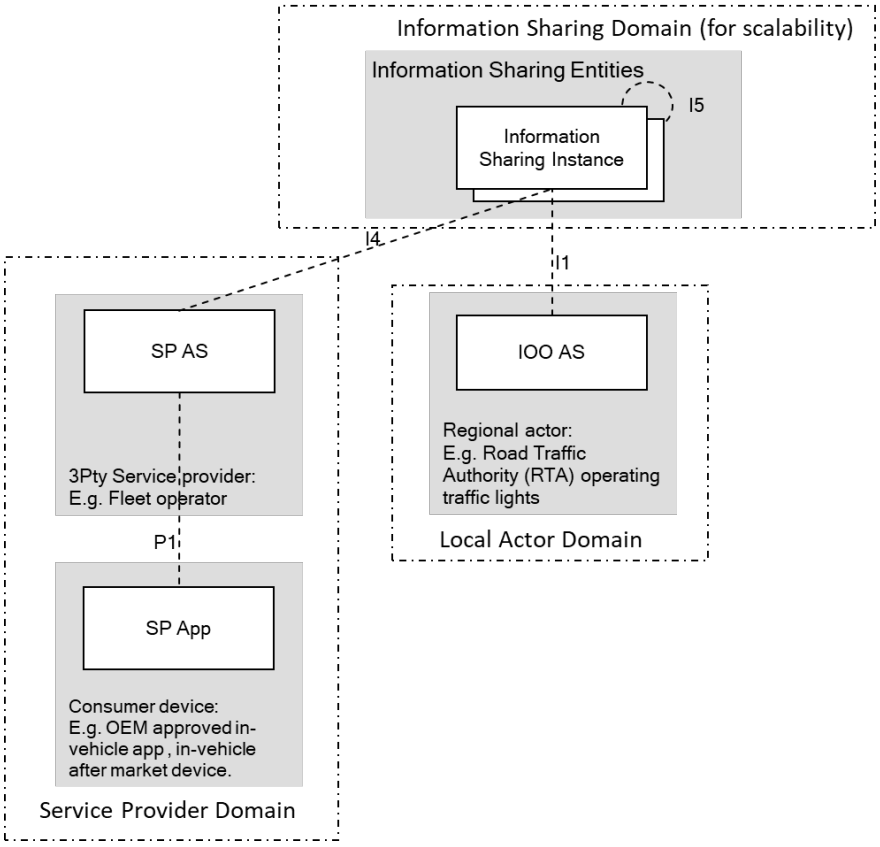


Figure 14: System architecture of traffic signal priority request UC – using Information Sharing Entities

Note: The above figure only shows cross-domain backend interfaces that are relevant to the Information Sharing Entities. Although not shown in the figure, cross-domain backend interfaces based on bilateral agreements can also be used between ecosystem stakeholders, e.g., P3 in Figure 1.

In this UC a SP, e.g., a fleet operator, has made arrangements and agreements with an IOO to request traffic signal priority for its operated vehicles, e.g., to better optimise traffic flow for public transport or heavy vehicles.

Prerequisites:

A. The SP AS, e.g., from a public transport operator or an ambulance operator, has access to Vehicle Information, e.g., location, direction, speed.

   1. If the SP App is located in the vehicle and implemented as OEM-supported SP App (see Section 7.2.2) or OEM-independent SP App on aftermarket device (see Section 7.2.3), the SP App can provide Vehicle Information. Note: this alternative is shown in Figure 14.
   2. If the OEM AS and the OEM-controlled App (OEM App) (See Section 7.2.1) are used to obtain Vehicle Information and the SP has established a trust and contractual relationship with the participating OEMs, a secure connection is established between SP AS and OEM AS, i.e., over the O2 interface. In this scenario it is also assumed that OEM ASs communicate with their vehicles (OEM Apps) over their proprietary interface O1 and act as proxy/filter for OEM Apps.

B. The SP has established a trust relationship with IOOs and has permission to request traffic signal priority in the trusted ecosystem, i.e., by publishing a SREM message in the Information Sharing Domain. The SP AS has obtained information about traffic signals that allow priority requests, including their identifiers, locations, etc. This can be done in several ways, e.g., by parsing official information made available from IOOs, through bilateral information sharing, by using the Information Sharing Domain. In the last case, the information can be published by IOO AS from the entity managing the traffic signals on the Information Sharing Instance using the I1 interface, e.g., by using ETSI MAP messages with topology information. The SP AS can then pick up this information on the I4 interface from the Information Sharing Instance.

C. The IOO ASs that control traffic signals have established a subscription on priority request messages (SREM) in the Information Sharing Domain.

UC execution:

1. The SP AS periodically obtains Vehicle Information from its vehicles e.g., location, direction, speed. The interface for obtaining Vehicle Information depends on in-vehicle deployment: via P1 or O2. In the latter case, the OEM AS obtains Vehicle Information via O1, as shown in Figure 13.
2. When a vehicle approaches an intersection, the SP AS requests priority if needed, e.g., when a public transportation vehicle or an ambulance is on a mission, by publishing a SREM message on the Information Sharing Instance using the I4 interface. The published message includes metadata indicating included message (the SREM) and geographic location (e.g., a quadtree tile, see Annex D) for the correct IOO AS to get the message, in case different IOOs control traffic signals in different regions.

   Note: The SREM messages may be generated by the SP App when the vehicle approaches an intersection. In this case, the SP AS verifies the SREM received via the P1 interface and prepare the format for sharing the SREM message (including the meta data) on the I4 interface.
3. The entity managing the traffic signals in the certain region receives the SREM from the Information Sharing Instance on I1, checks if requesting party is allowed to request priority.
4. If priority can be granted, the entity managing the traffic signals switches traffic signal state and publishes a SSEM message via I1on the Information Sharing Instance that distribute the message via I4 to the requesting SP AS that subscribes to SSEM messages.

Protocols used

On I1 and I4 interfaces, standard IT technology and processes should be used (see Annex G). E.g., AMQP is used for information sharing (publish/subscribe) and for providing metadata required in filtering operation (see Annex H) to identify payload, relevant area (e.g., based on quadtree tile concept, see Annex D), etc. TLS 1.3 with mutual authentication can be used for security.

> Note: The I5 interface is used in scenarios where SP and IOO connected to different Information Sharing Instances. In such case, subscriptions for SREM and SSEM messages are federated between the Information Sharing Instances.

The payload encapsulated by AMQP can be according to agreed formats among actors (i.e., the transport and information sharing solutions are payload-agnostic). For example, in C-Roads "IP based interface profile" [4] the following ETSI messages formats supported and are encapsulated as AMQP payload: SREM, SSEM, and CAM.

> Note: Since AMQP is payload-agnostic, SAE messages could be encapsulated, if supporting AMQP metadata are defined and available.

In the Nordic way project, Information Sharing Instances also support DATEX II (with defined meta data) as AMQP payload. For supported messages and protocols in C-roads, Talking Traffic, and Mobilidata solutions and Information Sharing Instances, see Annex B: Examples of.

A variant of this UC is supported in Talking Traffic and Mobilidata to prioritise bicycles at intersections. In this UC variance, infrastructure is used to identify bicycles using object detection. This detecting infrastructure then generates CAMs, which are sent to the Information Sharing Instance and then forwarded them to the relevant traffic signal controller. The traffic signal controller can apply priority according to its algorithm, e.g., number of bicycles needed, weather situation.

8.4 Use case IV: Emergency Vehicle Approaching

This UC is a special case of traffic event information sharing. In this UC a SP, e.g., a fleet operator of ambulances or fire brigade vehicles, has made arrangements and agreements to provide their location, direction, speed to other traffic participants in order to ease access. This UC has been deployed in the Mobilidata[21] and Talking Traffic[22] programmes, which are further described in Annex B.3 and B.2.

> Note: The position of police cars would likely not be shared due to other concerns.

[21] Mobilidata programme defined 31 traffic solutions in 5 different categories (intelligent traffic lights, navigation and parking management, risk and hazard notifications, traffic rules notifications and policy support) based on road-vehicle data collection and sharing https://www.mobilidata.be/en.

[22] This use-case was part of the Safety Priority Services sub-programme: https://dmi-ecosysteem.nl/en/themapagina-stedelijk-verkeer/ deployed of the Talking Traffic innovation program in the Netherlands: https://dmi-ecosysteem.nl/en/theme-page-urban-traffic/talking-traffic/.

8.4.1 Implementation options

8.4.1.1 Implementation option using interface "P1" between "SP App" and "SP AS"

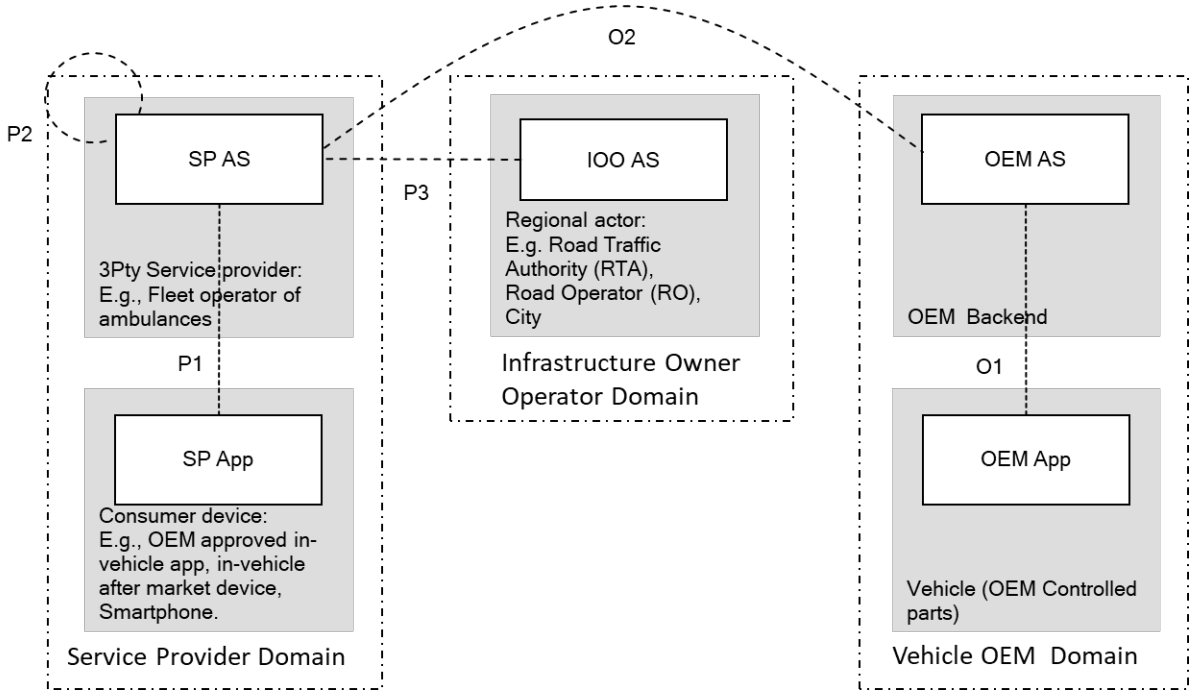


Figure 15: System architecture of Emergency Vehicle Approaching UC – using P3 interface

Prerequisites:

A. Access to Vehicle Information, e.g., location, direction, speed.

1. If the OEM AS and the OEM-controlled App (OEM App) (see Section 7.2.1) are used to obtain Vehicle Information and the SP has established a trust and contractual relationship with the participating OEMs, a secure connection is established between SP AS and OEM AS, i.e., over the O2 interface.
   In this scenario it is also assumed that OEM ASs communicate with their vehicles (OEM Apps) over their proprietary interface O1 and acts proxy/filter for OEM Apps.
2. If the SP App is located in the vehicle and implemented as OEM-supported SP App (see Section 7.2.2) or OEM-independent SP App on aftermarket device (see Section 7.2.3), the SP App can provide Vehicle Information via the interface P1.

B. The SP handling emergency vehicles has established trust relations and secured connection with other actors, e.g., OEMs and other SPs, which will provide Emergency Vehicle Approaching service to their clients (e.g., their connected OEM Apps and SP Apps).

UC execution:

1. The SP AS periodically obtains Vehicle Information from emergency vehicles including, e.g., location, heading and speed. The interface for obtaining Vehicle Information depends on in-vehicle deployment: via P1 or O2. In the latter case, the OEM AS obtains Vehicle Information via O1.
2. The SP AS periodically share emergency Vehicle Information with interconnected SP ASs (via the P2 interface) and / or OEM ASs (via the O2 interface).
    a. The SP AS may run an information sharing protocol such as AMQP on the P2 and / or O2 interface(s) and publish emergency Vehicle Information. So that the attached and subscribing AMQP clients, i.e., SP ASs and OEM ASs, are notified about the updated emergency Vehicle Information.
3. The interconnected SP ASs and/or OEM ASs disseminate the emergency Vehicle Information to their connected SP Apps (via P1) and/or OEM Apps (via O1) that are relevant to / affected by the information.
4. The receiving OEM Apps and SP Apps act on the received information, e.g., display the "Emergency Vehicle Approaching" information on the HMI or other available screens.

Optimisations of the UC are possible, e.g., in some scenarios the SP operating emergency vehicles may know the expected route of the emergency vehicle, calculate estimated times on positions along the route, and share the information with interconnected actors well in advance. This optimisation of the UC provides more time for road users to make space for the emergency vehicle.

Protocols used

The Emergency Vehicle Approaching use case may use DENM message defined in ETSI ITS at the application (also known as ITS Facilities) layer, or other messages defined in other regional SDOs, to convey the Emergency Vehicle Approaching information. For the SP AS to obtain periodically updated emergency Vehicle Information, CAM defined in ETSI ITS at the application layer, or similar messages defined in other regional SDOs, can be used. See Annex C "Talking Traffic" message frequency profile for cellular network implementation.

8.4.2 Scalable deployment using Information Sharing Entities

When the deployment scales up and involves more ecosystem stakeholders, the UC described above will require the use of Information Sharing Entities, e.g., to avoid a full mesh of connectivity among actors. Information Sharing Entity is further described in Section 6.4.

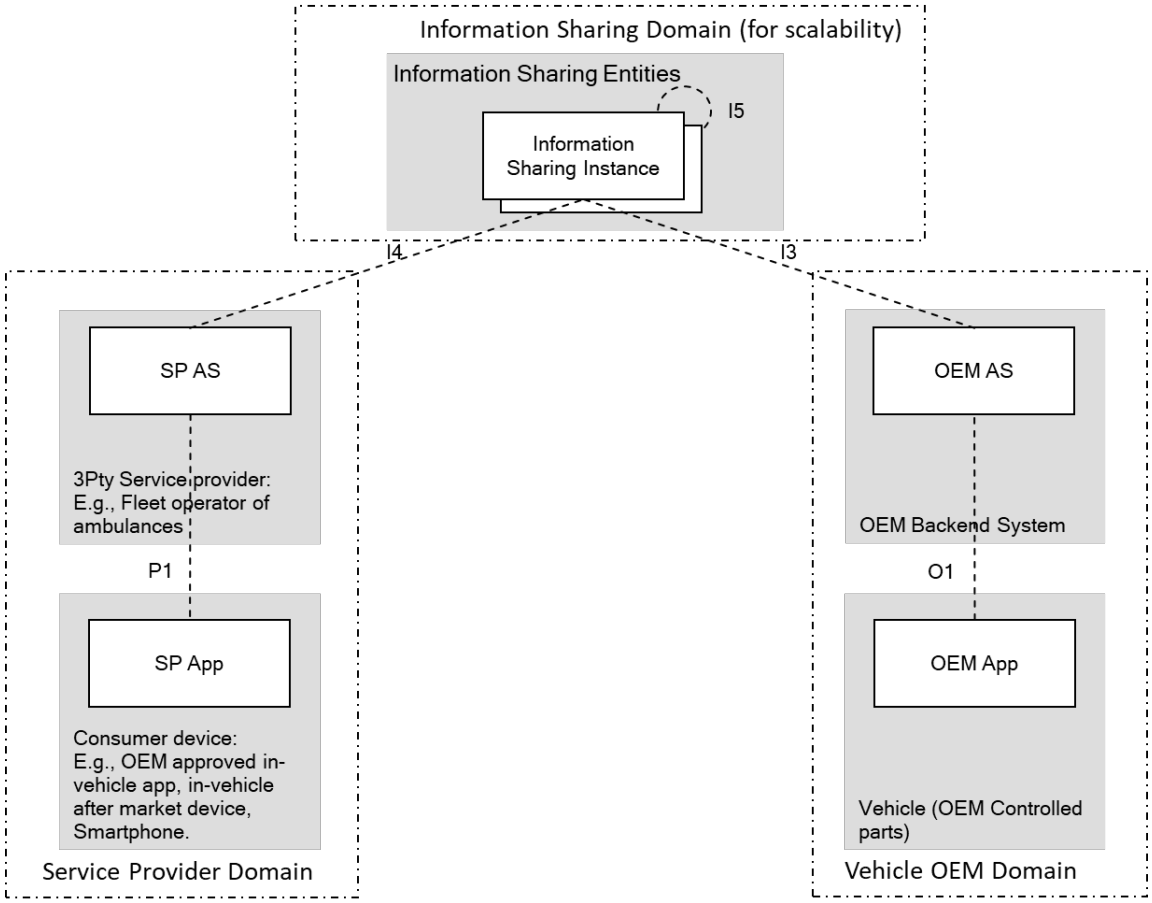


Figure 16: System architecture of Emergency Vehicle Approaching UC – using Information Sharing Entities

Note: The above figure only shows cross-domain backend interfaces that are relevant to the Information Sharing Entities. Although not shown in the figure, cross-domain backend interfaces based on bilateral agreements can also be used between ecosystem stakeholders, e.g., O2 in Figure 1.

In this UC a SP, e.g., a fleet operator of ambulances or fire brigade vehicles, has made arrangements and agreements to provide emergency Vehicle Information to other traffic participants in order to ease access.

Note: The position of police cars would likely not be shared due to other concerns.

Prerequisites:

A. The SP, e.g., the ambulance operator, has access to Vehicle Information of emergency vehicles, e.g., location, direction, speed. There are two scenarios:

   a. If the SP App is located in the vehicle and implemented as OEM-supported SP App (see Section 7.2.2), or OEM-independent SP App on aftermarket device (see Section 7.2.3), the SP App can provide Vehicle Information. This scenario is assumed in this example.

   b. If the OEM AS and the OEM-control OEM App (see Section 7.2.1) are used to obtain Vehicle Information and the SP has established trust and contractual relations with the participating OEMs,

a secure connection is established between SP AS and OEM AS, i.e., over the O2 interface.
In this scenario it is also assumed that OEM ASs communicate with their vehicles (OEM Apps) over their proprietary O1 interface and act as proxy/filter for OEM Apps.

UC execution:

1. The SP AS periodically obtains Vehicle Information from SP Apps via P1.

   Note: For the other scenario (see bullet A.b. in the Prerequisites list above), the Vehicle Information may be obtained from OEM AS via O2. (The OEM AS obtains Vehicle Information from OEM App via O1.)

2. The SP AS periodically publishes information about its emergency vehicles, including e.g., position, heading, speed, on the I4 interface.
3. The interconnected actors subscribe to this type of information and receives this information, e.g., SP ASs on I4 and OEM ASs on I3. Subsequently, SP ASs inform their relevant (affected) SP Apps via P1, and OEM ASs inform their relevant (affected) OEM Apps via O1.
4. The receiving SP Apps or OEM Apps act on the received information, e.g., display the "Emergency Vehicle Approaching" information on the HMI or visualise it on available screens.

Optimisations of the UC are possible, e.g., in some scenarios the SP operating emergency vehicles may know the expected route of the emergency vehicle, calculate estimated times on positions along the route, and share the information with interconnected actors well in advance. Such optimisation of the UC provides more time for road users to make space for the emergency vehicle.

Protocols used

On I1, I3, I4 and I5 interfaces standard IT technology and processes should be used (see Annex G). E.g., AMQP is used for information sharing (publish/subscribe) and for providing metadata required infiltering operation (see Annex H) to identify payload, relevant area (e.g., based on quadtree tile concept, see Annex D) etc. TLS 1.3 with mutual authentication can be used for security. For the I5 interface, additionally a HTTP REST based protocol is used for controlling signals, e.g., exchange capabilities information and handle subscriptions and data transfer between Information Sharing Instances on behalf of clients.

Typically, in Europe the AMQP payload data for this UC is DENM indicating "emergency vehicle", defined in ETSI. [21] Metadata indicating the payload type DENM is used in AMQP implementation of this UC for publishing/subscribing the emergency Vehicle Information. In other regions other messages may be applicable, e.g., for SAE BSM Part 2 could be used.

8.5 Use case V: HD MAP handling

In this UC a SP, i.e., a map provider offers services to the vehicle and provides an accurate HD MAP updated in real-time on the basis of information shared by other vehicles and optionally by infrastructure sensors. See the use case description in Section 5.4.6 of [20].

8.5.1 Implementation options

8.5.1.1 Implementation option using interface "P4" between "OEM App" and "SP AS"

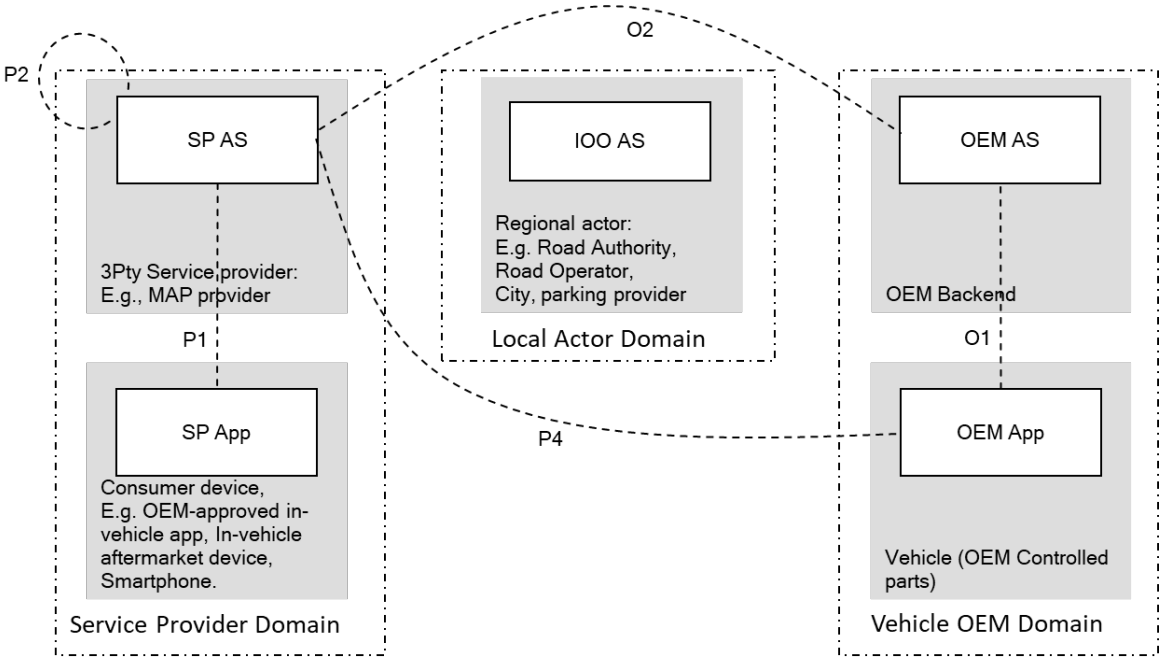


Figure 17: System architecture of HD MAP handling UC – using P4 interface

Use case deployment solution description

In this UC implementation option, a SP provides HD MAP services directly to the vehicle (OEM App) using the P4 interface, instead of going via OEM AS.[23] This deployment option is suitable if data volume to be transferred between SP AS and OEM Apps is high and the OEM wants to avoid backend handling of this data, e.g., to avoid scaling up the OEM backend resources. The deployment is also applicable when lower latency for data is required, i.e., to avoid processing delay added by the OEM backend. Examples of such services include HD MAP handling, i.e., download of MAP data, supporting more dynamic MAP layers, and upload of MAP data, Augmented Reality (AR) services to drivers/passengers, streaming of music and real time sports event, online gaming services. For this type of service, the OEM AS needs to allow the vehicle (OEM App) to connect to a SP AS and assist the vehicle (OEM App) with preparation of security credentials and additional information needed for the connection, e.g., SP AS addressing information.

[23] Providing HD MAP data and updates using OEM AS and the O1 interface is also a valid implementation option, while this section focuses on the option using the P4 interface.

Prerequisites:

A. The SP has established a trust and contractual relationship with the OEMs, for which the SP provides service. A secure connection is established between SP AS and OEM AS over the O2 interface for, e.g., preparing the security credentials to be used by the connection on the P4 interface. Agreements are in place for GDPR compliance.

B. OEM ASs communicate with their vehicles (OEM Apps) over their proprietary O1 interface for controlling and management purpose.

UC preparation for HD MAP handling with OEM-controlled App (OEM App) (HD MAP used as example)

1. HD MAP service is activated.
2. OEM AS receives address information and credentials (certificate) of the SP AS via the O2 interface.
3. OEM AS asks the OEM App to create a certificate using the O1 interface.
4. OEM App sends the vehicle certificate to OEM AS via the O1 interface. OEM AS signs the vehicle certificate and forwards it to SP AS via the O2 interface.
5. Via the O1 interface, OEM AS provides OEM App with SP AS address information and the SP credentials (certificate), and asks the vehicle to connect to the SP AS and establish a TLS connection using the exchanged credentials, i.e., to establish the P4 interface.
    a. P4 interface may be similar to the P1 interface that a SP uses for its own clients. However, additional OEM requirements need to be in place, e.g., regarding security and feature behaviour such as the agreed MAP layers, information to be included, update rate, etc.

UC execution for HD MAP handling with OEM-controlled App (OEM App) (HD MAP used as example)

1. Vehicles (OEM Apps) requests HD MAP information from SP AS using P4.
2. If agreement is in place, OEM App may also upload information for HD MAP to SP AS using P4, e.g., information about traffic events detected by vehicle sensors.
3. OEM App visualises HD MAP on HMI or potentially uses it in other in-vehicle functions.

Protocols used

O2 interface: SP and OEM agreed protocol.

P4 interface: SP and OEM agreed protocol, likely based on SP internal P1 protocol with extensions according to OEM requirements, e.g., regarding features, security.

O1 interface: OEM proprietary with support to handle credentials, to allow direct connectivity between OEM App and approved SP ASs.

P1 interface: SP proprietary, this protocol may also be applicable for scenarios with "OEM-supported SP Apps" (see Section 7.2.2) or "OEM-independent SP Apps" on aftermarket device or smartphone used in vehicle (see Section 7.2.3).

8.5.2 Scalable deployment using Information Sharing Entities

HD MAP handling is not a UC considered for Information sharing domains, it is based on business relations between map providers and their consumers. It is worth noting that HD MAP can be used by AD/ADAS functions Therefore, safety analysis and Threat, Risk, Vulnerability Assessments (TRVA) are required and will lead to strong additional requirements on the end-to-end data link and on the SP AS.

8.6 Use case VI: Automated Valet Parking/ Automated Vehicle Marshalling

When a vehicle arrives at the designated hand-over zone at the destination, the driver leaves the vehicle, and the vehicle is parked by an Automated Valet Parking System (AVPS) after being authorised by the driver. The use case description can be found in Section 5.4.3 of [19].[24] Such a service is also applicable for other "low-speed automation", known as AVM, e.g., factory parking or ranging of vehicles.

Automated Valet Parking/Automated Vehicle Marshalling (AVP/AVM) is a L4 driverless operation service bringing unoccupied vehicles from one location to another. ISO 23374-1 [10] defines three types of AVP according to the split of automated dynamic driving tasks between the infrastructure and the vehicle. AVP Type-2 is the first type of AVP service to be deployed by the industry, as the infrastructure takes the responsibility of sensing the environment and sending detailed driving instructions to the vehicle, making the UC already working with L2 vehicles.

When deployed in factories and logistic hubs, AVP/AVM can save cost by reducing labour hours, decreasing the need for human drivers and driver transportation, and improve productivity and quality. When offered as a service to private customers or fleet owners in public garages or areas, AVP provides convenience to the end users and saves their time, and potential to optimise parking space usage.

This section describes V2N2X deployment options and go-to-market considerations for AVP Type-2 as an example of such commercial service offered in public garages.

> Note: Section 8.1 and Section 8.2 in [11] elaborate the considerations for AVP Type-2 service deployment using cellular public networks (PN) and cellular standalone non-public networks (SNPN) respectively. For deployment using public networks, [11] discusses network coverage in parking facilities, network switching and mobility support among different MNO networks, QoS support and QoS on demand service for AVP Type-2/AVM, as well as the roaming situation. For deployment using SNPN, [11] elaborates the network authentication techniques potentially enabling the mobility between PN and SNPN.

[24] At the time this report is developed, the commercial AVP deployment by Bosch and APCOA in Germany received special permit to operate in public garage for selected vehicles without a safety driver. https://www.bosch-mobility.com/en/about-us/current-news/driverless-parking-from-hamburg-to-munich/ In these AVP deployments, WiFi technology is used for the communication between vehicles and the remote vehicle operation at the infrastructure. But the legal framework in principle does not preclude using other wireless communication technologies for AVP.

8.6.1 Implementation options

5GAA AVP TR [11] describes the application-level system architecture, end-to-end communication sequences and information flow, and cellular network solutions for AVP Type-2. In this section we represent the AVP Type-2 system architecture and discuss the deployment option using the reference architecture and conventions from Section 6 of the present report.

8.6.1.1 Implementation option using interface "V1" between "OEM App" and "IOO AS"

The application-level system architecture of AVP Type-2 [11] is represented in Figure 18 following the conventions from the V2N2X system architecture in Figure 1. Bold italic text in Figure 18 are names of system components and logical interfaces mapped to the V2N2X system architecture. Particularly, the Vehicle Motion Control (VMC) logical interface and the AVP Control (AVPC) interface defined in [11] map to interface V1 and O5 in Figure 1. It is worth noting:

- For simplicity reason, all sub-system components within the AVP Operator System, including AVP Operator AS, AVP Remote Vehicle Operator (RVO) AS, AVP Facility Management (FM) AS, and AVP FM App, are represented by the single system component "Infrastructure Owner Operator AS". This is because that the interfaces and communications among these sub-system components in the AVP Operator System are not of interests for this V2N2X work.
- In this work we only consider the implementation option of VMC interface directly between AVP RVO AS and the Vehicle App, i.e., without traversing through the vehicle/OEM backend.

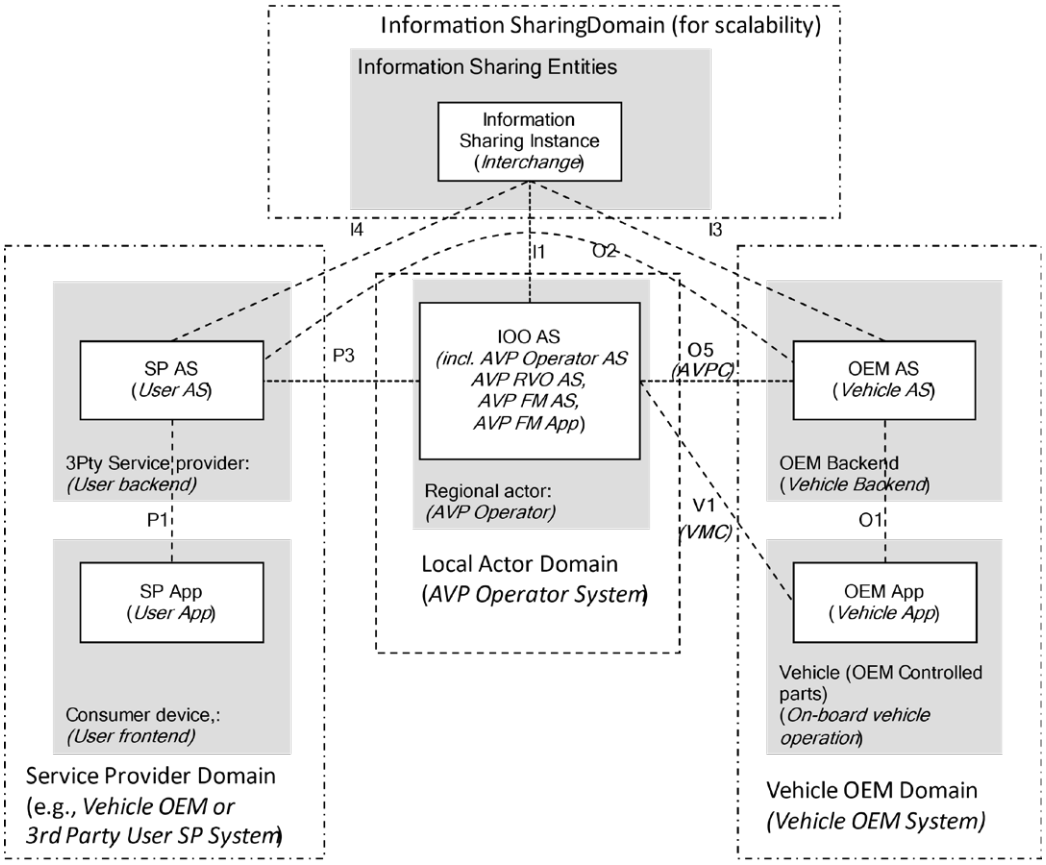


Figure 18: System architecture of AVP-Type 2/AVM UC – VMC using V1 interface

Note: In Figure 18, the AVP Type-2 application level system architecture [11] is mapped to the V2N2X architecture in Figure 1. Component names and interface names in brackets are defined in [11].

Prerequisites:

AVP Type-2 is a L4 driverless operation service, for which the AVP operator system needs the authorisation from the end user to take the responsibility of automatically driving the vehicle in the parking facility. To make it possible, a trust relationship shall be established between the AVP operator system and the vehicle OEM system, and between the concerned RVO AS and the served vehicle (and the end user). This requires:

- Before any AVP Type-2 session,
  - The parking facility shall be “approved”, e.g., according to certain certification process, for providing the AVP Type-2 service.
  - The vehicle brand and model shall be “approved”, e.g., according to certain certification process, for using AVP Type-2 service.
- For a given AVP Type-2 session, trust between the RVO AS and the vehicle shall be established:
  - At the network level, the user equipment at the vehicle and the AVP

network in the parking facility shall mutually authenticate each other, e.g., using €SIM or digital certificates, before exchanging any AVP Type-2 user data.

- At the application level
  - The OEM AS and AVP SP AS shall mutually authenticate each other before any AVP Type-2 session.
  - For any AVP Type-2 mission, the AVP RVO AS needs to be mutually authenticated with the connected Vehicle App.

UC execution:

The detailed communication sequence diagrams of AVP Type-2 that comply with ISO 23374-1 are documented in [11] for cellular network-based implementations. The process of AVP Type-2 service includes service discovery, service reservation, AVP Type-2 vehicle parking process, and if necessary, service payment.

- Section 7.2.1 in [11] describes the communication sequences for service discovery and reservation. Information Sharing Instance (Interchange) improve the scalability of this step, when multiple AVP operators and vehicle OEMs are involved in the deployment, as explained in Section 8.6.2.
- Section 7.3 in [11] describes the communication sequences for AVP Type-2 vehicle parking process. The whole process is divided into reusable modules so that AVP Type-2 missions like vehicle parking, vehicle reparking, and vehicle retrieving, can be implemented by combining the reusable modules. Ten such modules are described in subsections 7.3.1 to 7.3.10 in [11] following the AVP Type-2 application-level system architecture, which is mapped to the V2N2X architecture in Figure 18. Communication sequence modules in [11] also explain the interaction with underlying cellular network, e.g., when QoS on demand is needed for the VMC interface between AVP RVO AS and Vehicle App.

Protocols used:

For vehicle motion control over the VMC interface, messages and protocols specified in SDO, e.g., TS 103 882 AVM Service from ETSI, should be used. For AVP/AVM control signals over the AVPC interface, related stakeholders, e.g., parking operators, vehicle OEMs, and AVP/AVM technology suppliers, are still working on the messages and protocols.

To fulfil the security and privacy requirements of AVP Type-2, the implementation uses TLS/DTLS for end-to-end encryption of all communications over the VMC, AVPC, and other interfaces.

To ensure the interoperability among AVP/AVM Operators and vehicles from different OEM brands, stakeholders involved in the deployment also need to agree on implementation profiles of VMC and AVPC messages and protocols, which configure the VMC and AVPC standards to avoid ambiguous interpretation and implementation.

8.6.2 Scalable deployment using Information Sharing Entities

Vehicle Motion Control (VMC) communication in AVP Type-2/AVM is not a UC considered for Information Sharing Domains. It is a point-to-point connection between the AVP RVO AS and Vehicle App and has stringent latency and availability requirements. However, information sharing domain may serve a role to enable a scalable solution for announcing "parking availability information" in the service discovery step. For example, parking operators may regularly publish information about available parking service, address to parking facility, supported capabilities (for AVP type), location, which can be indicated as a tile according to the quadtree concept (see Annex D), Contact information (e.g.URL) for parking reservation, and potentially additional information like price. A vehicle OEM, whose vehicles support AVP Type-2, can subscribe to this information and, when a user of a car request for parking service, the vehicle OEM system can recommend or select an appropriate parking facility.

8.7 Use case VII: Object Detection and Sharing

This use case is the "Infrastructure Sensor Sharing" variant of "Data Sharing of Dynamic Object" described in Section 5.4.1 of [19]. Here the road infrastructure collects information about dynamic objects on/around the road, as well as vehicle sensor data. They share the relevant information as processed data.

8.7.1 Implementation options

8.7.1.1 Implementation option using interface "V1" between "OEM App" and "IOO AS"

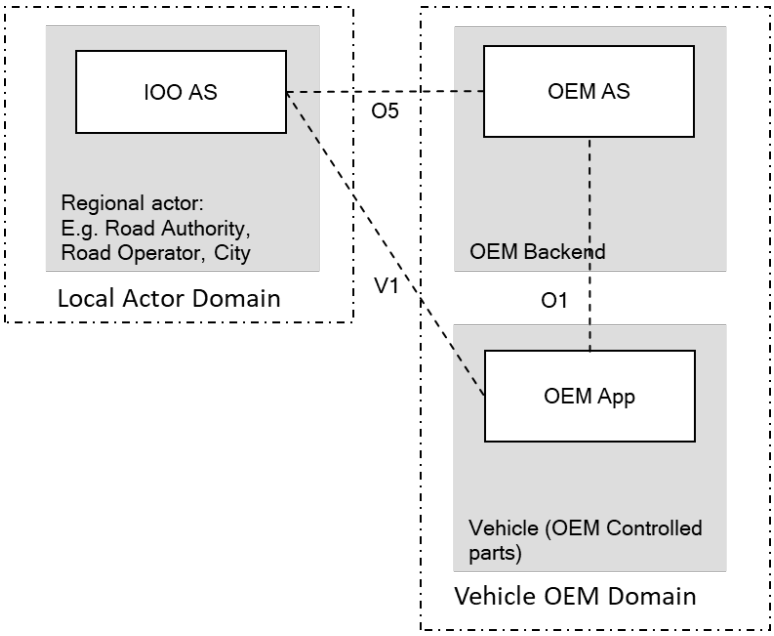


Figure 19: System architecture of Object Detection and Sharing UC – IOO provides service to vehicle via V1 interface

Use case deployment solution description

In this UC an IOO provides services directly to the vehicle, instead of going via the OEM backend. This deployment option is suitable if data volume to be transferred between the IOO AS and the OEM Apps is high and the OEM wants to avoid its backend handling this data, e.g., to avoid scaling up the OEM backend resources. The deployment is also applicable when lower latency for data transfer is required, i.e., to avoid processing delay added by the OEM backend. An example of such service is object sharing from infrastructure, e.g., a city, road operator, or road authority has installed cameras and/or other sensors with object detection at accident prone locations, such as intersections or zebra crossings, and provides information (such as position and time stamp) of the detected objects to vehicles in the vicinity. For this type of service, the OEM AS needs to allow the vehicle (OEM App) to connect to a IOO AS and assists the vehicle (OEM App) with the address information of IOO ASs and preparation of security credentials.

Prerequisites:

A. The IOO has established a trust and contractual relationship with the OEMs, to which the IOO provides service. A secure connection is established between the IOO AS and the OEM AS over the O5 interface for, e.g., preparing the security credentials to be used by the connection on the V1 interface. Agreements are in place for GDPR compliance.

B. OEM ASs communicate with their vehicles (OEM Apps) over their proprietary O1 interface and act as proxy for credential handling between vehicles (OEM Apps) and IOO AS.

UC preparation for object sharing to OEM-controlled App (OEM App)

1. Object sharing service is activated.
2. OEM AS receives address information of the IOO AS and the IOO credentials (certificate) via the O5 interface.
3. OEM AS asks the OEM App to create a certificate via the O1 interface.
4. OEM App sends the vehicle certificate to OEM AS via the O1 interface. OEM AS signs the vehicle certificate and forwards it to IOO AS via the O5 interface.
5. Via the O1 interface, OEM AS provides the OEM App with the address information of IOO AS and the IOO credentials (certificate) and asks the OEM App to connect to the IOO AS and establish a TLS connection using the exchanged credentials, i.e., to establish the V1 interface.

UC execution for object sharing to OEM-controlled App (OEM App)

1. Vehicles (OEM Apps) request object information from IOO AS using the V1 interface. This can be done using the following options:
   a. If the IOO AS has provided information about locations where cameras and/or other sensors are available, the OEM App can then generate requests based on its location, i.e., when it is in the vicinity of or approaching a camera location.
   b. The OEM App sends its geographical location, heading, and speed to the IOO AS and the IOO AS maps the vehicle location to relevant cameras and/or other sensors.
   c. Based on the "tile system" used by the IOO AS, the OEM App requests object information using tiles it is approaching.
2. The IOO AS provides object information related to the request.
3. OEM App visualises the received object information on HMI or potentially uses it for other vehicle functions.

Protocols used

O5 interface: IOO and OEM agreed protocol.

V1 interface: IOO and OEM agreed protocol. In some cases, e.g., to support AD/ADAS applications, the protocol needs to support functional safety. Object information (the payload part) may use ETSI CPM format, or SAE SDSM format. The format used for the objects can for example be pre-agreed based on region or indicated in the service request.[25] CAM or BSM messages can potentially be leveraged for a service request, since such message contains information of position, speed, heading, etc. The periodicity of CPM, SDSM, CAM, BSM should be adapted for cellular networks. See Annex C for the example message frequency profile used in cellular network-based implementation.

[25] The message frequencies to be used for handling CPM and SDSM are assumed to be the same as is currently used for sharing of SPaT messages to end user clients in Talking Traffic since similar need assumed, see Annex C.

O1 interface: OEM proprietary protocol with capabilities to handle credentials, provide information like IOO AS address to OEM Apps, and to allow direct connectivity between OEM Apps and approved IOOs.

8.7.1.2 Implementation option using interface "V1'" between "SP App" and "IOO AS"

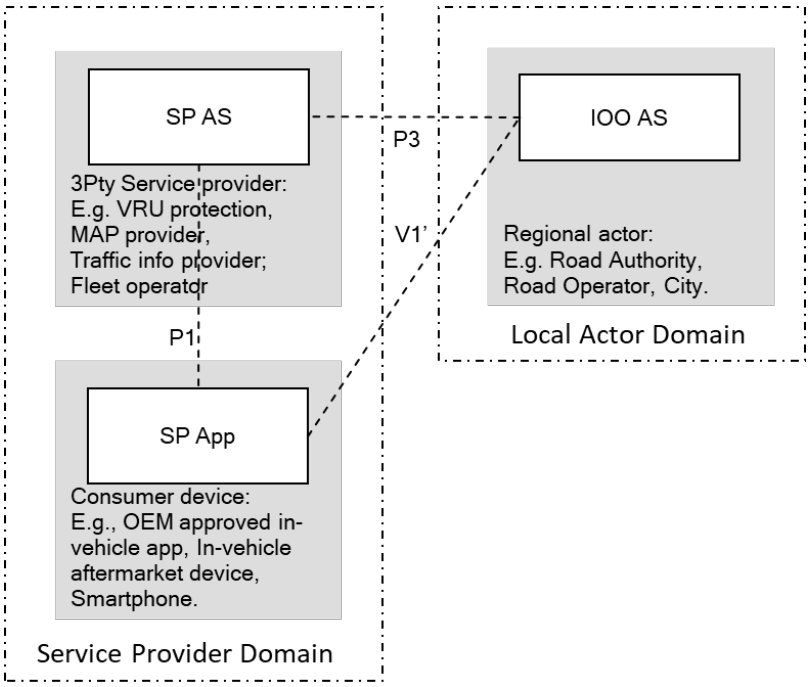


Figure 20: System architecture of Object Detection and Sharing UC – IOO provides service to SP App via V1' interface

Use case deployment solution description

In this UC an IOO provides services directly to a SP App, e.g., OEM-supported SP App or OEM-independent SP App as described in Section 7.2.2 and 7.2.3, instead of going via the SP backend (SP AS). This deployment option is suitable if large amounts of data need to be transferred between IOO AS and vehicle (OEM App), to avoid the SP backend (SP AS) handling the data and to reduce resource requirement of the SP backend. The deployment is also applicable when lower latency for data is required, i.e., to avoid processing delay added by the SP AS. An example of such services is object sharing from infrastructure, e.g., a city, road operator, or road authority installs cameras and/ or other sensors with object detection capability at accident prone locations such as intersections or zebra crossings and assist a bus operator (i.e., a fleet operator SP) with increased perception by providing the detected objects. For this type of service, the SP AS needs to allow its SP Apps to connect to a IOO AS and assist the SP App with preparation of security credentials.

Prerequisites:

A. The IOO has established a trust and contractual relationship with the SP, to which the IOO provides service. A secure connection is established between IOO AS and SP AS over the P3 interface for, e.g., preparing the security credentials to be used by the connection on the V1' interface. Agreements are in place for GDPR compliance.

B. SP ASs communicate with its clients (SP Apps) over their proprietary P1 interface and act as proxy for credential handling between SP App and IOO AS.

UC preparation for object sharing to SP App

1. Object sharing service is activated.
2. SP AS receives address information of the IOO AS and the IOO credentials (certificate) via the P3 interface.
3. SP AS asks the SP App to create a certificate via the P1 interface.
4. SP App sends the certificate to SP AS via the P1 interface. SP AS signs the certificate and forwards it to IOO AS via the P3 interface.
5. Via the P1 interface, SP AS provides the vehicle with address information of IOO AS and the IOO credentials (certificate) and asks the SP App to connect to the IOO AS and establish a TLS connection using the exchanged credentials, i.e., to establish the V1' interface.

UC execution for object sharing to SP App

1. SP Apps requests object information from IOO AS using V1'. Possible options are outlined below:
   a. If the SP AS has provided information about locations where cameras and / or other sensors are available, the SP App can then generate requests based on its location, i.e., when it is in the vicinity of or approaching a camera location.
   b. The SP App sends its geographical location, heading, and speed to the IOO AS and the IOO AS maps the SP App location to relevant cameras and/or other sensors.
   c. Based on "tile system" used by the IOO AS, the SP App requests object information based on the tiles it is approaching.
2. The IOO AS provides object information related to the request.
3. SP App visualise the received objects on available screen.

Protocols used

P3 interface: IOO and SP agreed protocol.

V1' interface: IOO and SP agreed protocol. Object information (the payload part) may use ETSI CPM format, or SAE SDSM format. ETSI CAM or SAE BSM messages can potentially be leveraged for a request since it contains information of position, speed, heading etc. The periodicity of CAM or BSM should be adapted for cellular networks. See Annex C for the example message frequency profile used in cellular network-based implementation.

P1 interface: SP proprietary protocol with capability to handle credentials, provide information like IOO AS address to SP Apps, and to allow direct connectivity between SP Apps and approved IOOs.

8.7.1.3 Implementation option using interface "P1" between "SP App" and "SP AS"

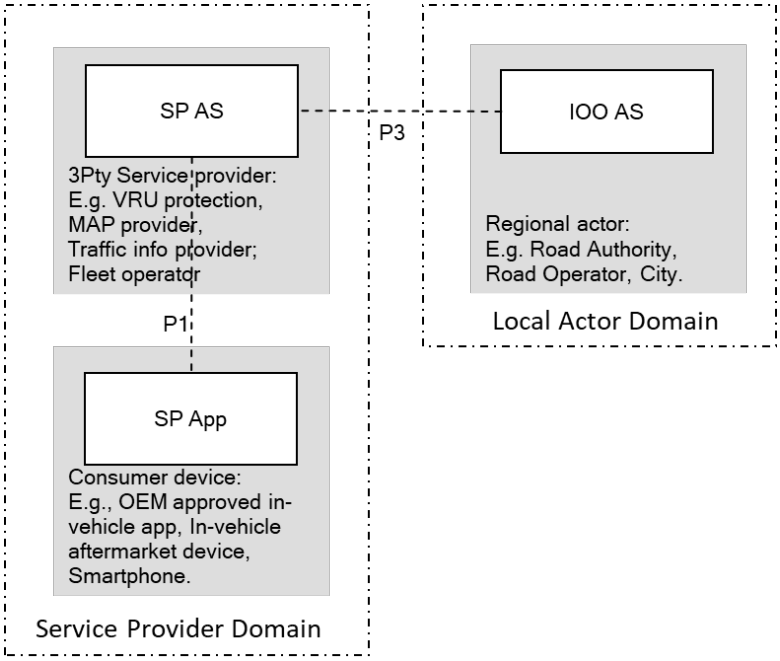


Figure 21: System architecture of Object Detection and Sharing UC – SP provides service to SP App via P1 interface

Use case deployment solution description

In this UC the end user use SP App from a SP and risk warning information received via the P1 interface in V2X applications, e.g., to improve the perception of the environment. Such risk warnings are generated by the SP AS based on the status (e.g., location, speed, etc.) information form the SP App and the object data in the vicinity of the SP App provided by the IOO(s), i.e., from IOO AS to SP AS. Depending on the required data rate and latency performance by the V2X application, the SP AS may have different deployment options. In case of low data rate and relaxed latency requirements, the SP AS may be implemented in central cloud and connected to higher number of IOO ASs. Otherwise, if the data rate and latency requirements are stringent, the SP may prefer to deploy the SP ASs on MEC platform and closer to a limited number of IOO ASs, e.g., in a region, a city, or even at an intersection. The application layer deployment solution described below is generally applicable for both central cloud and MEC deployment. If safety-critical alerts are needed, though not described in this V2N2X implementation option, direct communication would be used as well, if available.

Prerequisites:

A. The IOO has established a trust and contractual relationship with the SP, to which the IOO provides service. A secure connection is established between IOO AS and SP AS over the P3 interface for transmitting the object data. Agreements are in place for compliance to personal data protection law, e.g., GDPR.

B. SP ASs communicate with its clients (SP Apps) over their proprietary P1 interface. Methods should be taken to allow SP AS to receive and handle personal data from SP App, e.g., position information, in compliance to the applicable personal data protection law, e.g., GDPR.

UC preparation for object sharing to SP App

1. Object sharing service is activated.
2. SP AS receives periodical position information from SP App via P1 interface. Based on the position information the SP AS identifies the proper IOO AS that can provide the object data in relevance to the SP App and establishes the corresponding P3 connection to the identified IOO AS.

   Note: If the SP AS can identify another SP AS that can better serve the SP App, e.g., a SP AS that is deployed closer to the target IOO AS, it suggests the SP App switching the P1 connection to the new SP App. Upon the request from the SP App or SP AS, the new SP AS will stablish the P3 connection to the IOO AS.

UC execution for Object Detection and Sharing to SP AS:

1. The SP AS requests IOO AS to sharing object data via the P3 interface.

   Note: The request may be triggered by the SP App via the P1 interface or triggered by SP AS based on the position information of the SP App, who has the object sharing service activated.
2. IOO AS performs object detection (e.g., VRU detection) using camera/sensor data at the infrastructure.
3. SP AS receives object data from IOO AS via the P3 interface.

   Note: If needed, the SP AS may perform advanced AI Video Analytics (e.g., AI tracking, path history/predictions, collision detection algorithms, etc.)
4. SP AS processes the received object data and performs the risk and hazard detection based on the information of SP App, e.g., position and speed. The SP AS provides safety warning (e.g., PSM, BSM, or DENM) to SP App via the P1 interface, if potential collision risk is detected.

Protocols used

P3 Interface: IOO and SP agreed protocols for service discovery, negotiation, and connection establishment. For sharing the object data over P3, using standardized message format, e.g., ETSI CPM or SAE SDSM, provides cross-vender interoperability.

P1 Interface: This interface uses SP proprietary protocol with capability to handle SP App status information, risk warning messages, and provide information like addresses

of other SP ASs to SP Apps. ETSI CAM or SAE BSM messages can potentially be leveraged for service request or position update, since it contains information of position, speed, heading etc. The periodicity of CAM or BSM should be adapted for cellular networks. See Annex C for the example message frequency profile used in cellular network-based implementation.

8.7.2 Scalable deployment using Information Sharing Entities

The sharing of objects via information sharing domain and backend systems for use in clients (SP Apps or OEM Apps) may not be optimal, unless data load and latency performance can be accepted[26]. Sharing of object data thus will likely be done using V1' and V1 interfaces utilising a direct connection between the end user client (SP Apps or OEM Apps) and the provider of object data (IOO ASs). Establishment of such connections are usually under the control by respective stakeholders' backends, i.e., the SP AS and OEM AS in this UC.

However, Information Sharing Domain and backend systems can be used for scalable service discovery to obtain information about where objects data are provided and how to fetch them. Information sharing is further described in Section 6.4.

[26] One UC using Information Sharing Entities for detected objects is traffic light priority for intersections in Talking Traffic and Mobilidata solutions. For this UC latency and load are acceptable. In this UC infrastructure is used to identify bicycles using object detection. This detecting infrastructure then generates CAMs which are sent to the Information Sharing Instance, which then forwards the CAMs to the relevant traffic light controller which can apply priority as deemed appropriate.

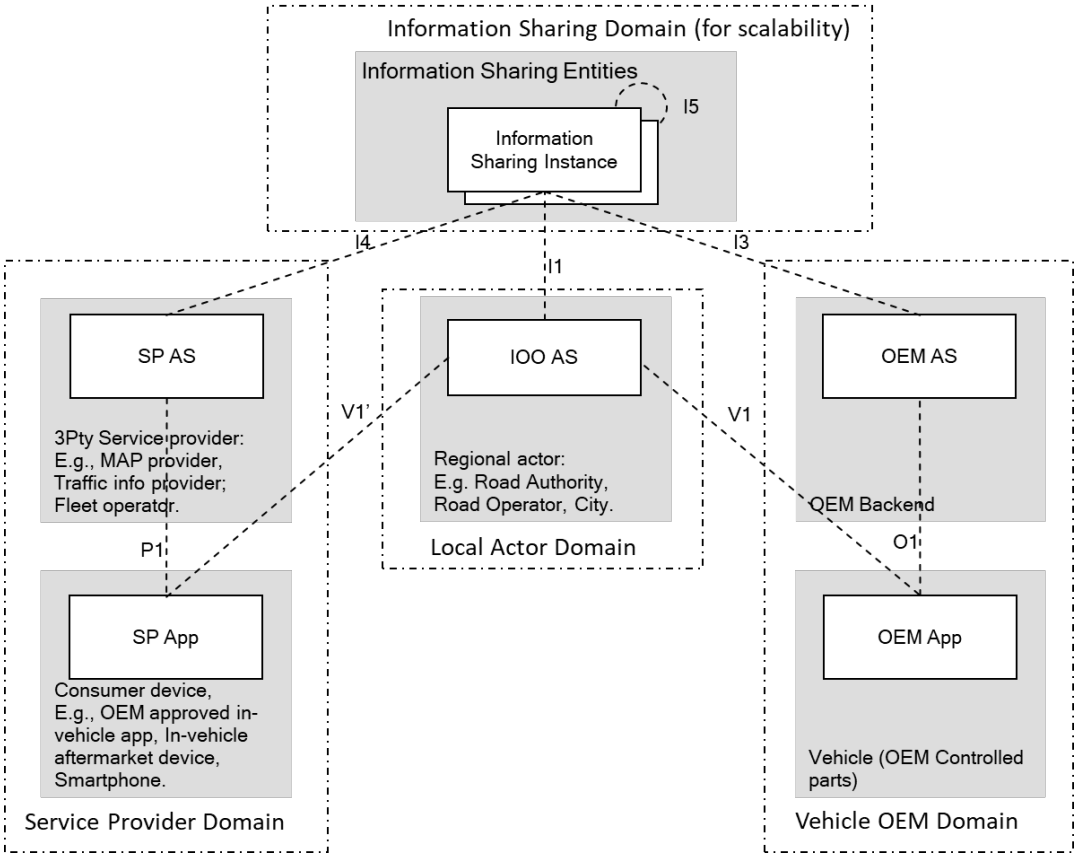


Figure 22: System architecture of Object Detection and Sharing UC – using Information Sharing Entities

Note: The above figure only shows cross-domain backend interfaces that are relevant to the Information Sharing Entities. Although not shown in the figure, cross-domain backend interfaces based on bilateral agreements can also be used between ecosystem stakeholders, e.g., O2, O5, P3 in Figure 1.

For a scalable service discovery, 'Information Sharing Entities' are used to share information about where Object Detection and Sharing services are available. In general, actors in, e.g., one country or one region, are connected to one Information Sharing Instance. This Information Sharing Instance is then interconnected with Information Sharing Instances in other countries or regions.

> Note: There can be more than one Information Sharing Instance per country or region depending on system topology, organisations, data traffic load situation, etc.

The network of interconnected Information Sharing Instances thus form a federated information sharing backbone, where information from the whole ecosystem is available wherever an actor is connected. This means that the backend of an actor that is a user of the Object Detection and Sharing service, e.g., a SP AS or an OEM AS, connected to one Information Sharing Instance, may obtain information about the providers of the service connected to another Information Sharing Instance.

In this example, the service discovery is based on a publish/subscribe model using the AMQP protocol with metadata (implemented as AMQP application properties), to

allow message filtering based on what an actor is interested in, e.g., location, type of message, etc.

Information exchange UC: Once preparations as described in Section 6.4 are in place and connectivity, publishing agreements and subscription filters are established, information exchange can be performed.

1. A trusted actor in the interconnected ecosystem, e.g., an IOO, a City, or a road operator has deployed infrastructure for object detection at certain locations, e.g., accident prone locations such as intersections, zebra crossings or bus stops.
2. The trusted actor, e.g., the IOO AS, publishes information to the connected Information Sharing Instance with associated AMQP metadata indicating, e.g., format of the detected object (e.g., ETSI CPM or SAE SDSM), location of the object detecting entity (e.g., country & quadtree tile, see Annex D), producer of the information, and address information where the object data can be fetched (e.g., URL). This publishing is done using I1.
3. The receiving Information Sharing Instance checks which backend clients (SP ASs and/or OEM ASs) have a matching subscription based on the established filters and pushes the information to those backend clients (SP ASs and/or OEM ASs) using the I3 and/or I4 interfaces. Operation on both interfaces basically follow the same mechanism but may have different filter configurations.

   Note: Here the federated Information Sharing Domain is applicable, i.e., a client (SP AS or OEM AS) connected to another Information Sharing Instance but subscribing to the same information/event can also get this information.
4. A backend client (SP AS or OEM AS) receiving the information about the availability of detected objects can thus select to forward this information to its relevant clients (e.g., SP Apps or OEM Apps) depending on their location and heading.
5. SP Apps or OEM Apps, if allowed by the respective SP AS or OEM AS, can thus establish a connection to the object data source (IOO AS) and obtain object data using the V1' or V1 interface.

Protocols used

On I1, I3 and I4 interfaces, standard IT technology and processes should be used (see Annex G), e.g., AMQP is used for information sharing (publish/subscribe) and for providing metadata required in filtering operation (see Annex H) to identify payload, relevant area (e.g., based on quadtree tile concept, see Annex D), etc. TLS 1.3 with mutual authentication can be used for security. For the I5 interface, additionally a HTTP REST based protocol is used for controlling signals, e.g., exchange capabilities information and handle subscriptions and data transfer between Information Sharing Instances on behalf of clients.

AMQP metadata is needed for service discovery as outlined above in the "Information Exchange UC" part.

8.8 Use case VIII: Vulnerable Road User protection – VRU Collision Risk Prediction and Alert

Several examples of this use case have been demonstrated by 5GAA members at recent open events, with slightly varying approaches to the service hosting architecture. All currently known approaches are outlined here, for completeness, although the functional architecture remains the same.

In the demonstration by Vodafone (Malaga 2022), the Vulnerable Road User (VRU) device – smartphone mounted on cycle handlebar – and the vehicle device – a smartphone mounted on the interior vehicle windscreen – generated ETSI CAM message and sent them to the Service Provider's central function over the Uu interface. In this case, both VRU and vehicle devices were connected to the same SP AS. The role of the central function was to relay the CAM to nearby road users' clients. Each client was able to actively control the geographical area from which corresponding road users' CAMs are received, to manage processing load on the client, through the subscription process. The receiver client used its current position in combination with received CAM to predict the collision risk between the two road users. When a certain level of collision risk was predicted a visual, tactile or audible alert was generated within the road user's device. The use case can be similarly implemented using a dedicated VRU awareness message or similar messages from other standards bodies (e.g., BSM Part 1). A demonstration using a similar approach was presented by KDDI in the 5GAA Tokyo meeting 2024[27].

In the approach taken by 5GAA members Deutsche Telekom, Telefonica and Continental, also Verizon Wireless, Telus and Harman, vehicle and VRU-based clients generated CAM (in Europe) and BSM Part 1 (in North America) respectively and sent them to a central function hosted on the SP's edge platform (i.e., MEC). In this solution the messages to be relayed to corresponding end users (via a peer application server hosted by the other SP), are filtered by the Application Server, to reduce processing load on the end device, according to the relative proximity, direction of travel and combined speed, between pairs of road users. The receiving clients calculate the risk of collision for each received message and if necessary, generate a visual, tactile or audible alert to the road user. Road user clients were hosted on consumer smartphones, in both examples.

At the 2023 5GAA Detroit meeting, Verizon, T-Mobile, LGE, Commsignia, Keysight, and Anritsu showcased an Interoperability VRU DEMO[28]. This demonstration set up Application Servers from LGE and Commsignia, each hosted on Verizon and T-Mobile's Edge platforms, communicating through an MQTT protocol-based solution and an Information Sharing Instance for interworking. VRU and vehicle applications exchanged PSM and BSM messages via the Uu interface, enhancing mutual awareness. The

[27] More information about the KDDI and Toyota demonstration can be found using the following links:
- Demonstration details: https://news.kddi.com/kddi/corporate/newsrelease/2023/01/30/6519.html (in Japanese)
- Background about the demonstration and early implementation of a safety and secure mobility society: https://news.kddi.com/kddi/corporate/english/newsrelease/2024/02/20/7291.html

[28] More information about the 5GAA demonstration of Interoperability of VRU Protection Services via Network Connection can be found at https://5gaa.org/5gaa-showcases-cutting-edge-c-v2x-technology-pioneering-the-future-of-vehicle-connectivity/.

demonstration showed a practical example of VRUs communicating their crossing intentions to vehicles, thereby facilitating safer crossings.

In the above approaches the performance of the solution is optimised by the deployment of the main server functionality on an MNO-hosted MEC platform, reducing the latencies experienced during the transmission of C-ITS messages towards the collision prediction functionality, thereby leading to an earlier collision risk prediction and alert generation, all other things being equal.

8.8.1 Implementation options

Several variants of the use case are possible, using one or more SPs. In the example where there are two, each road user is subscribed to a different SP's service and an interconnecting service-level interface between the two must be established in order to enable low-latency sharing of the other road user's current position, velocity, etc., for the collision risk prediction to be made in each SP's domain. In this interconnect scenario SPs will perform collision risk prediction in their own service domains, rather than rely on a prediction made remotely in another SP's domain.

In the above-mentioned interconnect scenario the different SPs' server applications could be hosted in the same edge compute domain (roaming, federated edge scenario [2]) or in separate edge compute domains (non-roaming or no federated edge scenario). In the latter example, a low-latency interconnecting inter-MEC interface must be established to support the service, however this interface is out of scope here, it is addressed in the 5GAA gMEC4Auto WI [3].

8.8.1.1 Implementation option with a single V2N2X service provider

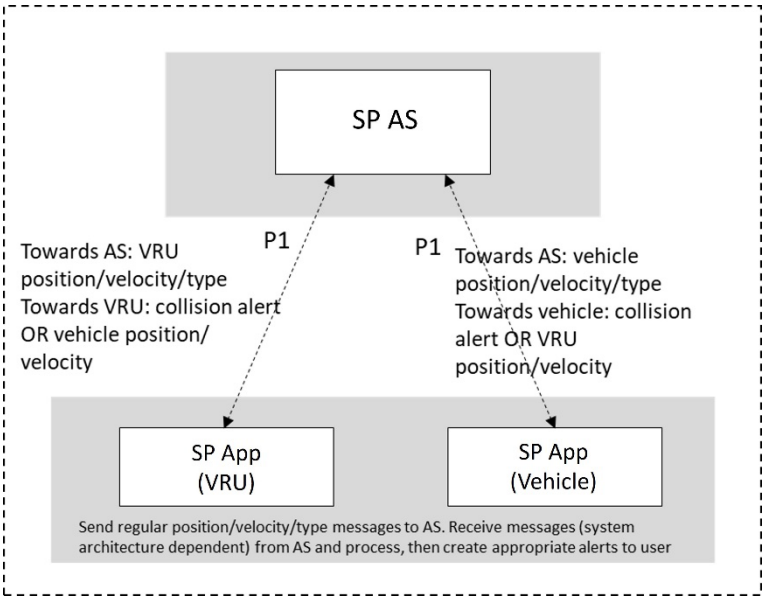


Figure 23: Single SP VRU Collision Prediction and Alert use case

In this UC the service provides a suitably low-latency VRU-to-vehicle collision prediction and visual/audible/tactile alert signal to end users.

1. The SP has provided the device software (SP App) to the UC participants (road users), and a trust relationship between the user and the SP is established. GDPR requirements are observed throughout.
2. Communications between the SP Apps and the SP AS are over Uu, with a secured client-server connection.
3. Each SP App is configured to present the participant's role (e.g., bicycle, e-Scooter, vehicle, etc.) with messages conveying position, velocity (including direction) and vehicle type information.
4. Each SP App frequently generates the above-mentioned message and sends it towards the SP AS function (P1 interface).
5. The SP AS identifies pairs (or sets) of road users with a need for the Collision Prediction service to be applied based on a combination of proximity, direction of travel, and speed. Depending on the service architecture, the SP AS either:
    a. Relays the relevant position/velocity information to both road users' SP App (P1 interface), depending on proximity and combined velocity, or
    b. Performs the collision risk prediction locally and sends resulting collision warning messages to both road users SP App (P1 interface).
6. When the SP App receives a position/velocity message (P1 interface) it performs a collision risk prediction and when certain proprietary parameters are exceeded then a visual, audible, or tactile alert is presented to the road user (i.e., VRU and vehicle driver).
7. Alternatively (to #6), when a SP App receives a collision warning message (P1 interface) then a visual, audible, or tactile alert is presented to the road user (i.e., VRU and vehicle driver) based on the information conveyed within the message.
8. The road user(s) will manually react to the alert as deemed appropriate to avoid the collision.

Protocols used

P1 interface: In this instance there is no requirement for any of the messages used to be standardised. Both user clients (SP Apps) are provided by the same SP and they can therefore use proprietary message formats since the system is "closed". However, there are existing standard messages that are appropriate for this use case (i.e., CAM, VAM, CPM, BSM Part1, DENM, BSM Part2) so the SP *could* adopt these. Currently, no standardised, profiles exist to determine the rate at which these messages should be generated and there is no specific need for a standard profile in a closed, proprietary system.

8.8.1.2 Implementation option using separate Service Providers

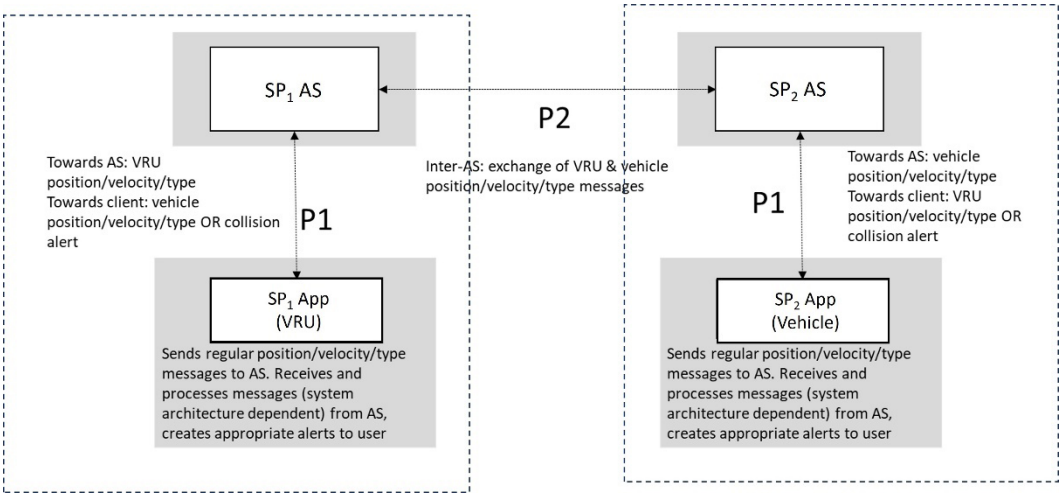


Figure 24: Dual SPs exchange road-user real-time messages to enable VRU Collision Risk Prediction and Alert

In this variant two Service Providers' SP ASs interconnect directly to provide the VRU Collision Prediction Service.

1. Each SP has provided the device software (SP App) to their respective UC participant, and a trust relationship between the user and the SP is established. GDPR requirements are observed throughout.
2. Communications between the SP Apps and the SP AS are over Uu, with a secured client-server connection.
3. The SP AS function of one SP implements an interface (P2) towards the other SP's SP AS, for the purpose of sending and receiving road users' position/ velocity and other related parameters in real time, with appropriate low latency.
4. Each SP App is configured to represent the participant's role (i.e. bicycle, e-Scooter, vehicle, etc.) with messages conveying position, velocity (including direction) and vehicle type information.
5. Each SP App frequently generates the above-mentioned message and sends it towards the SP AS (P1 interface).
6. On reception of the road user's message (P1) the SP AS immediately forwards the message over the interconnect interface to the corresponding SP AS, also retaining a local copy of the message to enable collision prediction locally, if this is the service architecture adopted.
7. On receiving a message over the P2 interconnect interface, the SP AS either
    a. Relays the relevant position/velocity information to appropriate road users' SP App (P1 interface), based on relative proximity and combined velocity or some other geographic information (i.e., geofence), or

b. Performs the collision risk prediction and, when certain proprietary parameters are exceeded, sends resulting collision warning message(s) to the road user (P1 interface).

8. When the SP App receives a position/velocity message (P1 interface) it performs a collision risk prediction and when certain proprietary parameters are exceeded then a visual, audible, or tactile alert is presented to the road user (i.e., VRU and vehicle driver).
9. Alternatively (to #8), when a SP App receives a collision warning message (P1 interface) then a visual, audible or tactile alert is presented to the road user (i.e., VRU and vehicle driver) based on the information conveyed within the message.
10. The road user will manually react to the alert as deemed appropriate to avoid the collision.

Protocols used

P1 interface: Similar to variant 1, there is no service level requirement for messages sent over the P1 interface to be standardised, since both SP Apps are provided by their respective SP so they could use proprietary message formats and each client-server system is closed. However, in this case where each message is relayed to a corresponding peer VRU collision risk prediction SP AS (P2 interface), the use of standardised ITS messages over the P1 interface will enable the SP AS to relay messages in a standardised format with minimal adaptation/translation required.

P2 interface: Since each road user's SP App generated message (conveying position, velocity, etc.) must be relayed over the P2 interface to the other Service Provider's SP AS function, using ITS message formats standardised by regional SDOs would enable interconnect architectures based on open, public standards. Existing standard ITS message formats are appropriate for this use case (i.e. CAM, VAM, CPM, BSM Part1) and these *could* be implemented across this interface (depending on the region of operation).

Service discovery mechanisms: *TBD.*

8.8.1.3 Integrated VRU client application options

Most, if not all, new vehicles have a cellular modem installed during manufacture to support OEM business-related services (i.e., telematics) and driver comfort/ infotainment services (e.g., sat-nav, information). VRU Collision Risk and Warning client applications for safety enhancing scenarios could be hosted by vehicles, re-using the integrated 4G/5G cellular modem, GNSS system, and MMI system. The benefit of integration of such apps into the vehicle could be a wider (e.g., default) usage of the service by drivers, compared to smartphone-based solutions, and improved delivery of warnings to drivers. In addition, service data (such as CAM or BSM) generated by the vehicle itself would be more accurate and generatable at higher message frequencies, due to superior on-board GNSS antenna and processor systems, since current existing consumer smartphones have limitations in these areas.

It is worth noting that other systems, such as video and lidar sensors or short-range

V2P/V2I solutions, can be used to implement a similar service at closer range (safety critical scenario) and if those systems are not occluded. The SP-based VRU service would be a complement to such a service, the vehicle can be expected to make decisions as to the appropriate usage of data arriving from multiple sources (i.e., Uu/PC5/sensor), based upon metadata carried with ITS messages.

Some options for this integration are outlined below. Approaches discussed here do not refer to screen-mirroring type solutions, as seen with Android Auto and Apple Car Play (see Section 7.2.2 Type-B), since these do not leverage the vehicle's in-built GNSS and cellular modem. Service performance is also not in-scope in this section, it will be addressed elsewhere.

Should OEMs enable integrated VRU-related applications into their vehicles, a number of approaches appear to be possible. Options are:

1. Third-party SP App hosted by the vehicle (see Section 7.2.2 Type-A). The SP App could be delivered via an OEM's or associated service provider's OEM-curated app store (e.g. within Android Automotive). In this case the SP App provides the full functionality of the service (communications interface and UI aspects). The VRU application service is hosted by the SP. Connectivity between the SP App and the SP AS uses the MNO's internet Access Point Name (APN) or dedicated APN supporting MEC-hosted deployments.
2. OEM-controlled VRU in-vehicle service client application installed during manufacture (see Section 7.2.1). The VRU application service (OEM AS) is hosted in the OEM's service domain, which could be in the OEM cloud or hosted on a MEC. The communication between OEM AS and OEM App is via the interface O1. The supporting APN would be dedicated to the OEM's services and could also include MEC-hosted OEM server application deployments.
3. OEM-originated client, installed during manufacture, incorporating a third-party SP's client-server interface (P4), hosted in the SP's environment (see Section 7.2.1). The OEM App is responsible for aspects enabling the primary service other than message exchange (i.e., risk calculation, alert generation, etc.) In this case the SP AS provides functionality for relaying timely service data between on-road participants (vehicle and VRU). Connectivity between the app and the server would use the MNO's internet APN or an APN dedicated to the SP's AS, including MEC-hosted deployments.

Interfaces used

Option 1 – SP App and SP AS via P1. The SP App, implementing the SP's application logic for collision prediction and UI, could be pre-installed (by the OEM) or installed/enabled by the user post-sale (see Section 7.2.2 OEM-supported SP App Type-A). In this case the SP's end-to-end VRU Collision Prediction and Alert service is likely to be required to satisfy some permissions and functional/security/privacy requirements set by the OEM into whose vehicles it will be installed. One option for post-sales integration is for the SP App to be provided to the vehicle via an OEM-curated app store (or by an OEM partner). The SP's VRU service will also be expected to meet the OEM's performance requirements for latency, interconnections to other systems, and prediction accuracy. If the corresponding VRU (i.e., pedestrian, cyclist, etc.) is hosted by the SP, the P1

interface to the VRU completes the message path, otherwise SP AS interfaces with other service providers via the ISI (I4) or, if this is unavailable, via a direct interface (P2 to other SPs or O2 to OEMs implementing the VRU service). Service authorisation towards the SP would be provided by the OEM over the O2 interface on a regular basis, giving the OEM control over the service provision for its vehicles.

Option 2 – OEM App and OEM AS via O1. The service is hosted and provided fully within the OEM's environment. The client application is installed in the vehicle by the OEM during manufacture and can be activated by the owner/driver after purchase (see Section 7.2.1 OEM-Controlled App). In this case, the OEM App and the application server are created and maintained by the OEM (or its Tier-1 partner). The O1 interface connects the two and is not required to support published standards. The OEM AS includes all of the functionalities that comprise the VRU Collision Risk and Prediction service, including timely vehicle tracking, message relay and interconnect between the OEM AS and other SPs, via the I3, O2 or O4 interfaces. These interfaces must support common message standards and interconnect protocols, so message format translation is likely if open standard message formats are not supported on the O1. A clear requirement of this service is the need to support low-latency message exchange with other SP ASs, so it is likely that the OEM AS will not be hosted in the traditional OEM backend cloud platform, but could be hosted by a third party (e.g., MNO) distributed edge cloud which enables lower latencies between the vehicle client and the application server, together with lower interconnect latencies to other SPs in the region or locality.

Option 3 – OEM App supported by SP AS via P4. In this option the OEM (or its Tier-1 provider) elects to create/maintain the OEM App (including the collision prediction logic and UI) but to leverage a third-party SP to provide the message relay and interconnect functionality described above (see Section 7.2.1 OEM-Controlled App). In this approach the OEM App implements an interface toward the SP AS, which is defined and implemented by the SP, (the P4 interface), which implements the same messages and protocols of the P1 but includes additional aspects tailored for the OEMs. This option allows the OEM to provide the VRU service to its customers without the requirement to maintain the server functionality. The OEM is responsible for implementing the algorithms to predict VRU collision risks and provide alerts to the driver, and for generating vehicle-based messages to be sent towards the SP AS (and on towards the SP App). The SP AS is responsible for efficiently implementing VRU-related message delivery to and from the OEM App. The OEM will select the SP based upon its service quality, which could include interconnect scale, latency performance (including interconnect scenarios outlined in Option 1) and data efficiency. In this approach, the OEM AS could include the interface functionality of more than one SP, if this offers some advantage to the OEM. Service authorisation towards the SP AS would be provided by the OEM AS over the O2 interface on a regular basis, giving the OEM control over the service provision towards its vehicles.

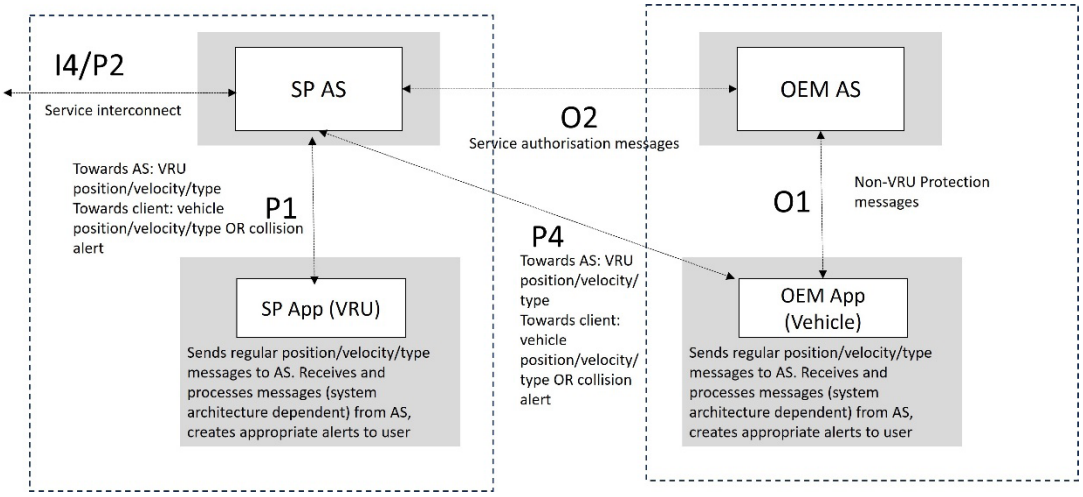


Figure 25: VRU Collision Risk Prediction and Alert using in-vehicle application implementation Option 3, OEM-controlled App (OEM App) using P4 interface towards SP AS

8.8.2 Scalable deployment using Information Sharing Instance

The VRU Collision Prediction use case will ultimately require the use of Information Sharing Instances, e.g., to avoid an inefficient full mesh of connectivity among actors. Information sharing is described in chapter 6.4 Note: figure only show cross-domain interfaces related to interaction with the information sharing domain, there might also be additional cross-domain interfaces between actors based on bilateral business agreements.

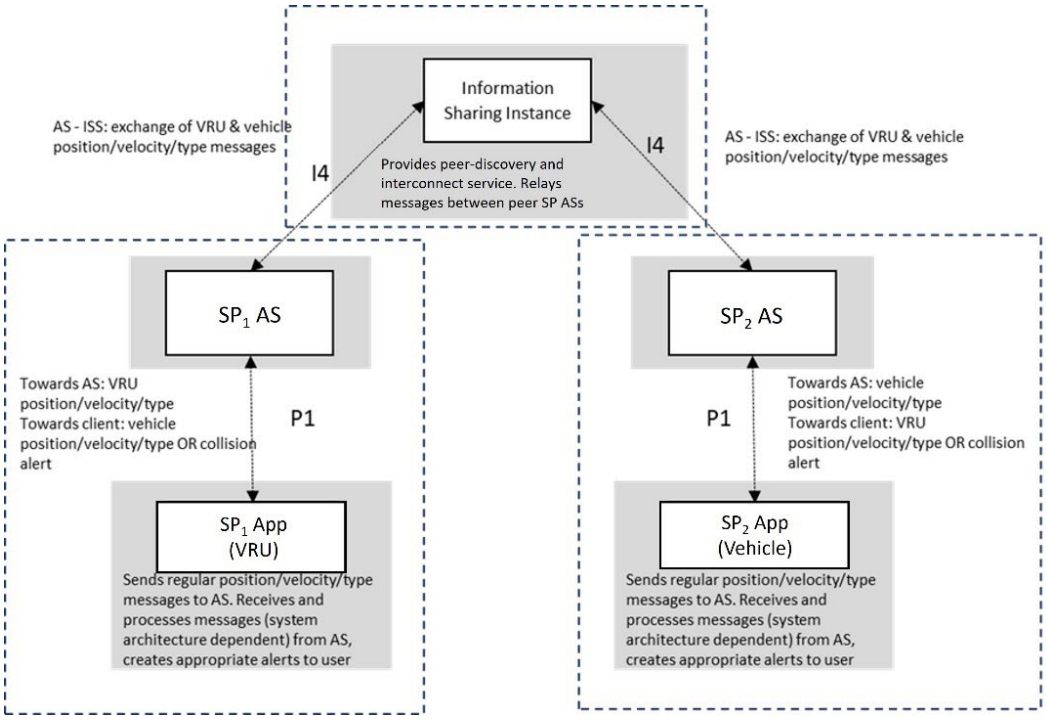


Figure 26: Dual SP connected via Information Sharing Instance to enable VRU Collision Risk Prediction and Alerts (NB. non-integrated SP App (vehicle) represented in this example)

Note: The above figure only highlights cross-domain backend interfaces that are relevant to the Information Sharing Entities. However, bilateral cross-domain backend interfaces not shown in the figure can also be used between ecosystem stakeholders, e.g., P2 in Figure 1.

In this variant two SP application servers interconnect via an Information Sharing Instance to provide the VRU Collision Prediction service

1. Each SP has provided the device software (client app) to their UC participant, and a trust relationship between the user and the SP is established. GDPR requirements are observed throughout.
2. Communications between the client app(s) and the application server are over Uu, with a secured client-server connection.
3. The SP AS function implements an interface (I4) towards a common Information Sharing Instance service, for the purpose of exchanging road users' position/velocity and other service-related parameters in real time, with appropriately low latency.
4. Each SP App is configured to represent the participant's role (i.e., VRU cyclist or vehicle) with messages conveying position, velocity (including direction), and vehicle type information.
5. Each client frequently generates the road user's position, velocity etc. message and sends it towards the associated SP AS (P1 interface).
6. On reception of the road user's message (via P1) the SP AS immediately forwards the message over the interconnect interface to the corresponding AS, retaining a local copy of the message to enable collision prediction (for the attached SP App) at the SP AS, if this is the service architecture adopted.
7. On receiving a message over the I4 interconnect interface, the SP AS either
    a. Relays the relevant position/velocity information to appropriate road users' client (P1 interface), based on relative proximity and combined velocity or some other geographic information (i.e., geofence), or
    b. Performs the collision risk prediction and, when certain proprietary parameters are exceeded, sends resulting collision warning message(s) to the road user (P1 interface).
8. When the SP App receives a position/velocity message (P1 interface) it performs a Collision Risk Prediction and when certain proprietary parameters are exceeded then a visual, audible or tactile alert is presented to the road user (i.e., VRU and vehicle driver).
9. Alternatively (to #8), when the SP App receives a collision warning message (P1 interface) then a visual, audible or tactile alert is presented to the road user (i.e., VRU and vehicle driver) based on the information conveyed within the message.
10. The road user will manually react to the alert as deemed appropriate to avoid the collision.

Protocols used

P1: Similar to variants 1 and 2, there is no service level requirement for messages sent over the P1 interface to be standardised, since both SP Apps are provided by their respective SP so they could use proprietary message formats because each client-server system is closed. However, in this case where each message is relayed to a corresponding peer VRU collision risk prediction SP AS (I4 interface) via an Information Sharing Instance, the use of standardised ITS messages over the P1 interface would enable the SP AS to forward messages in a standardised format with minimal adaptation required by the participating SPs.

I4: Since each road user client generated message (conveying position, velocity, etc.) must be relayed over the I4 interface to other SP AS functions, via the Information Sharing Instance, adopting ITS message formats standardised by regional SDOs would enable interconnect architectures based on open, public standards. Existing standard ITS message formats are appropriate for this use case (i.e. CAM, VAM, CPM, BSM Part1) and these *should* be implemented across this interface (depending on the region of operation).

Service registration and discovery mechanisms are implemented between the SP ASs and the Information Sharing Instance. This aspect is for further study.

8.9 Deployment considerations for V2N2X use cases

As shown in this chapter, the V2N2X implementation examples of V2N2X solution blueprint, as described in Chapter 6, cover a broad range of use cases associated with different application layers and network requirements. The following observations are made from the UC examples discussed in this chapter:

The O1 deployment option provides vehicle OEMs better control of the data. As a result, such data may be used for vehicle functions implemented as vehicle OEM-controlled App (OEM App) (see Section 7.2.1). However, as application data are processed or routed via the OEM backend (OEM AS), such a deployment option is not recommended for UCs with high data load and/or stringent latency requirements.

The P1 deployment option enables V2X applications using the vehicle OEM-supported Apps (see Section 7.2.2) and OEM-independent SP App (see Section 7.2.3). This greatly increases the penetration rate of V2X application for end users. Similar to the O1 deployment option, application data are processed or routed via the SP backend (SP AS), such a deployment option is not recommended for UC with high data load and/or stringent latency requirements.

The P4 interface connects vehicle OEM App and SP AS, to enable V2X service provisioning by SP to vehicle OEM controlled Apps (OEM App). As P4 is a cross-stakeholder interface, special considerations are needed by the involved OEMs and SPs, regarding responsibility, interoperability, security, etc. The actual solutions depend on the agreements among involved stakeholders.

The V1 and V1' deployment solution allows communication between IOO AS and OEM App/SP App without additional the involvement of backend entities (OEM AS / SP AS). This is preferred by applications generating high data load or requiring low latency. Like the P4 interface, V1/ V1' are also cross-stakeholder interfaces. As an IOO stakeholder is involved, standardised messages and protocols are recommended to ensure interoperability. Furthermore, trust and security are equally important considerations for V1 and V1'.

Information Sharing Instances, as described in Section 6.4, enable scalable and interoperable E2E cross-stakeholder implementation of V2X applications. From the UC implementation examples discussed in this chapter, we see Information Sharing Instances can be used to share application data, e.g., traffic event and traffic signal information, when the traffic load and latency requirements are not stringent. In case the data load is high, or latency is critical for the application, e.g., object data detection and sharing UC or AVP/AVM UC, the Information Sharing Instances can be used for service discovery and initiation instead of conveying application data, which is also important for scalable deployment of the UC.

9 Architecture and UC conclusions

V2X services can be supported using existing cellular network communication in combination with interacting backend systems. Solutions described in this TR have been proven feasible and effective in accelerating the V2X service penetration by various deployments. Especially for UCs, which require interactions between road infrastructure and other road users, or UCs, where information needs to be delivered over long distances but with less stringent latency requirements. The solutions described in this TR are considered currently viable. With enhanced cellular network coverage, radio capacity and capabilities, and network features such as MEC, QoS and Network Slicing, it is foreseen that also more demanding UCs can be addressed by cellular communication.

10 Business perspectives on V2N2X deployments

Introduction

The emergence of cellular networks has facilitated myriad capabilities within the transportation ecosystem, particularly in the realm of data exchange. The following is an abstract of the Technical Report "Business Perspectives on Vehicle-to-Network-to-Everything (V2N2X) Deployments" [21] describing the V2N2X market from a business standpoint, encompassing market dynamics, stakeholder analysis, and business models deployed in various exemplary instances.

Market analysis

The V2X market, encompassing both cellular and direct communication technologies, is projected to grow substantially, with estimates indicating a market value surpassing USD 20 billion by 2030. The V2N2X market, a subset of this, holds significant potential, with cellular connectivity expected in the millions of vehicles and smart city installations by 2025. Market growth is propelled by various factors including societal challenges (safety, traffic flow, sustainability, urbanisation, etc.) digitalisation efforts, Euro NCAP directives, and, specifically in the EU, legislative mandates, with cellular coverage expansion playing a pivotal role.

Stakeholder analysis

Stakeholders in the V2N2X ecosystem, including road users, infrastructure operators, and vehicle OEMs, exhibit distinct roles, needs, and expectations. The report describes these needs in terms of "jobs to be done" and the "pains and gains" related to these jobs.

Infrastructure operators seek safer and more efficient transport systems, leveraging V2N2X for traffic management and operational efficiencies. Vehicle OEMs, currently less engaged in sharing, are driven by impending legislation and safety imperatives, emphasising data sharing and scalable solutions. However, it is noteworthy that the sharing of information "within the own brand" is already common.

These direct stakeholders are supported by Service Providers, Mobile Network Operators, Field Equipment Manufacturers and Technology providers who benefit indirectly from the implementation of V2N2X services by selling services to the direct stakeholders.

Business models in exemplary deployments

Examining (fairly) large-scale deployments in regions such as the Netherlands, Belgium, the US, and China reveals diverse business models underpinning V2N2X implementations. These models involve collaboration between Policymakers, Service Providers, Technology Vendors, and Mobile Network Operators, with revenue streams derived from information services, data monetisation, and infrastructure investments.

Conclusion

The V2N2X ecosystem presents lucrative revenue opportunities for stakeholders, albeit amid challenges such as data standardisation, privacy concerns, and cost uncertainties.

Collaborative efforts among stakeholders, coupled with education on existing showcases and technological capabilities, are imperative for overcoming these obstacles and realising the full potential of V2N2X deployments.

Annex A: Generic V2X application layer architecture

Figure 27 shows the generic V2X application layer reference architecture. All interfaces in Figure 27 are logical interfaces at the application layer. The implementation details of each interface depend on the deployment options, e.g., using Uu, PC5, or other communication technologies. System components and interfaces that apply to V2N2X systems and solutions are described with details in Chapter 4.

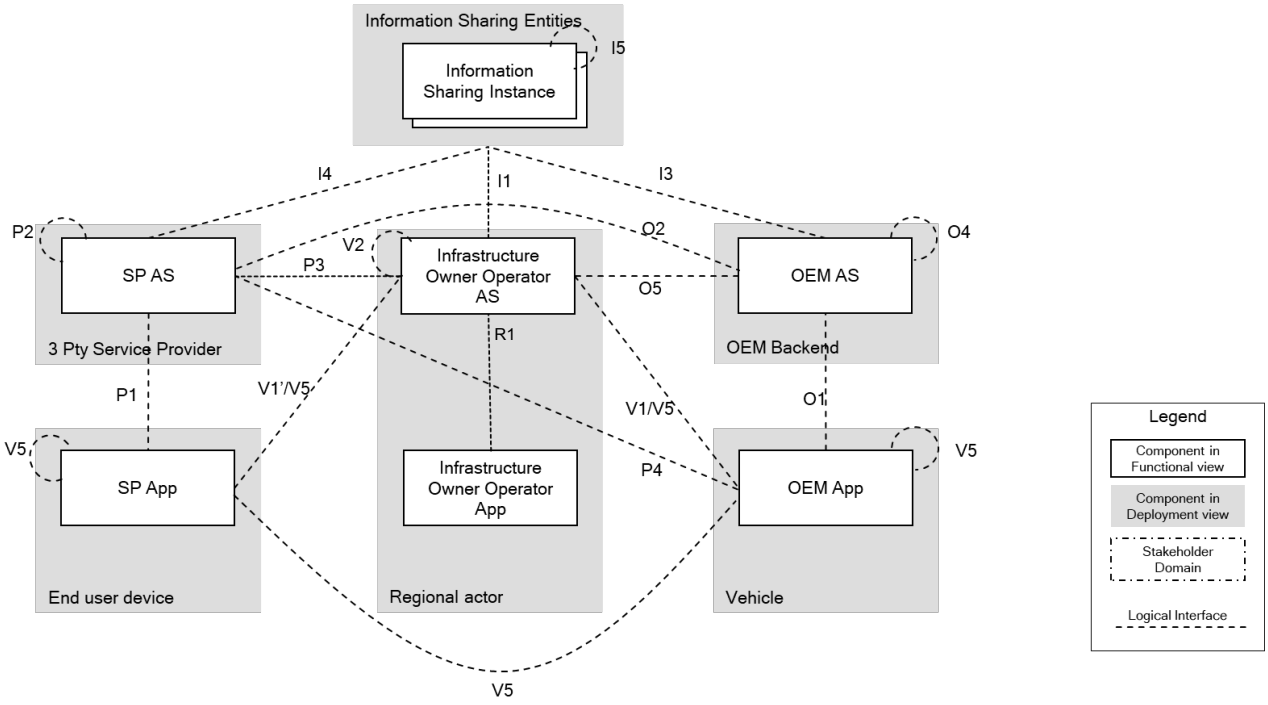


Figure 27: Generic V2X application layer reference architecture

Annex B: Examples of Information Sharing Instance

This annex collects existing implementations of Information Sharing Instance as supplementary information to Section 6.4 Information Sharing for Scalable and Interoperable Deployment

B.1 C-Roads Information Sharing Domain Principles

The model for information sharing specified by the European road authorities and member states in C-Roads is described in the project's specification for "IP based interface profile" [4]. This specification is intended for information sharing between backend systems and describes a publish/subscribe model using Advanced Message Queuing Protocol with metadata (AMQP application properties) to allow message filtering based on what an actor is interested in, e.g., location, type of message, etc.

AMQP is selected as the protocol because it is richer in capabilities (e.g., for filtering) and since communication in this domain is not bandwidth constrained. The other common publish/subscribe protocol Message Queuing Telemetry Transport is more intended for simple devices with limited capabilities and bandwidth constrained networks. MQTT is more applicable for communication between backend systems and end clients (e.g., vehicles and smartphones) and would, as such, add an additional scalability layer.

Figure 28 shows the C-Roads model for the Information Sharing Domain, interface names within brackets refer to C-Roads naming. The C-Roads profile also outlines governance, security, and discovery. (Note: The figure only shows C-Roads-related interfaces, there might also be additional interfaces between actors based on business agreements.)

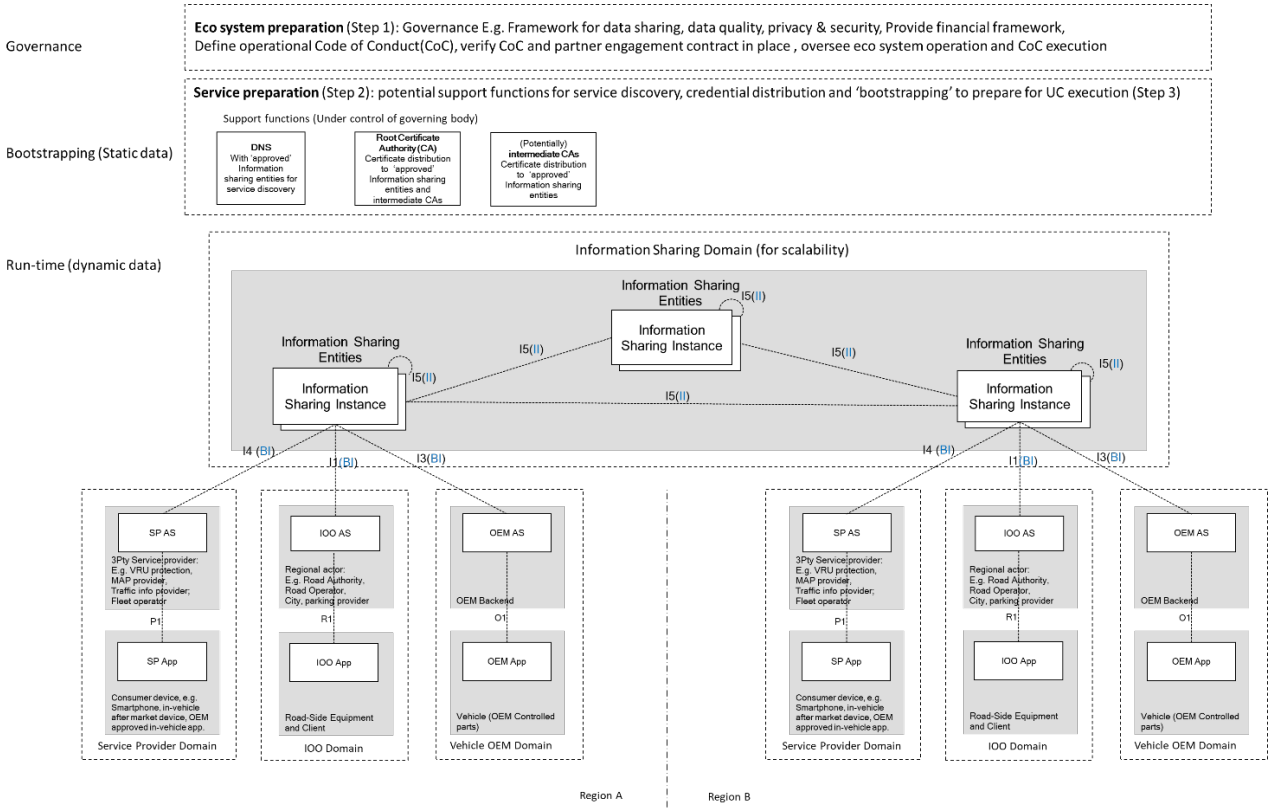


Figure 28: C-Roads model for information sharing

A reference model for this is the "Talking Traffic" solution that uses contracts and a governance model to ensure the system performance, integrity, and quality. This model is also being applied in other solutions like Mobilidata (more information about these solutions in Annex B.2 and B.3).

This solution has a preparatory phase where participating actors are approved – i.e., security credentials are distributed, service discovery is performed. Then connectivity is established between actors that have been approved; passed validation checks, signed agreements, etc. After the preparatory phase, information sharing and/or interaction can take place. Below is a short summary of procedures described in C-Roads' "IP based interface profile" [4].

Preparation phase consists of "ecosystem preparation" and "service preparation", as outlined in step 1 and 2 in **Figure 2:**

1. "Governing Body" sets the rules (e.g., framework for data sharing, data quality, privacy and security), provides the financial framework, defines the operational Code of Conduct (CoC), verifies the CoC and partner engagement contracts, etc. Once these are in place, actors can be approved.
2. "Governing Body" allows the distribution of certificates to actors for secure communication using TLS and for data signing (to ensure authenticity and proof of data origin). Intermediate CAs may be present in the certificate chain, e.g., operated by Information Sharing Entity operators to handle certificate distribution to their clients. It also initiates updates of the DNS

server(s) (standard DNS that is authoritative for this Information Sharing Domain) with information about Information Sharing Instances (to enable automatic discovery of commissioned Instances).

3. "Information Sharing Instances" perform a DNS query to obtain addresses of other Information Sharing Instances and initiate the establishment of TLS connections to create the federated Information Sharing Domain using the I5 interface (In C-Roads [4] this interface is called II). Once the connectivity is established, these Instances exchange "capabilities" using the HTTP-based control protocol on I5 interface. Capabilities refer to information about supported areas (countries & tiles), message sets, etc. that are supported and the URL where data set can be fetched. (A tile-based solution using Quadtree is used to indicate an area of arbitrary size, see Annex D.) The exchange of capabilities is needed so information available in one country is available to clients connected to an Information Sharing Instance in another country. An Information Sharing Instance performs such discovery at start-up and when capabilities have been enhanced, e.g., when a new message set is supported.
4. "Client Actors" (i.e., Sp ASs, IOo ASs and OEm ASs as clients) establish a connection using I1, I3 or I4 interfaces to the Information Sharing Instances of their preference (in C-Roads [4] this interface is called BI), to which they made an agreement, e.g., the local instance in their country of presence.

   Note: In C-Roads [4] the BI interface (in Figure 15, called I1, I3 or I4 interfaces) are using the same protocols and AMQP metadata to encapsulate payload information, but have variances in what payload is supported, what is published and subscribed to (e.g., an IOO may support publishing In-Vehicle Information Message or IVIM, but it would be OEMs and SPs that subscribe to such information).

   Note: A client can connect to multiple Information Sharing Instances for redundancy reason.
5. Client Actors provide information to the connected Information Sharing Instances about what information they can publish and in which areas, i.e., tiles according to the Quadtree concept, see Annex D. The clients also provide information about the location and type of information that they are interested in; this allows a subscription filter to be configured in the Information Sharing Instances.

Once preparation, connectivity, publishing agreements and subscription filters are established, information exchange can be performed, and this step is described per applicable use case in Chapter 8.

B.2 Talking Traffic Information Sharing Domain Principles

Solutions from Talking Traffic[29] are similar to the C-Roads principles as described in Annex B.1, but with different and additional interfaces, e.g. to Traffic Light Controllers. Full documentation is available at: National iVRI standar–s - CROW.[30] The descriptions include documentation needed for a complete system, e.g., contract documents, message content, message profiles, step-by-step plans and processes, checklists for acceptance test, HTTP and JSON REST APIs, Security (TLS), etc.

Talking Traffic is a successful innovation programme to bring digital infrastructure and connected vehicles to large-scale deployment in The Netherlands, leveraging the existing cellular networks. In the preparation phase of the programme, a group of authorities, led by the Ministry of Infrastructure and Waterworks, agreed on a set of use cases suitable for their country, mainly around signalled intersections. These use cases were Priority/pre-emption for designated road users, leveraging vehicle probe data for improved traffic flow efficiency, and GLOSA/TTG.

With the use cases selected, a public-private governance structure was created consisting of:

- A committee of senior policymakers responsible for authority alignment.
- A committee of subject matter experts from the authorities, involved in operational aspects.
- A joint body of senior representatives from the industry and representatives from the previous mentioned committees, called the Strategic Council (SC).
- The Change Advisory Board, a committee open for participation by all stakeholders.

This structure worked together to create the initial Common Code of Conduct[31], consisting of technical and non-technical arrangements:

- Examples of non-technical elements:
  - Standardised privacy (data processor) agreements
  - Long-term funding for the governance structure (small deposit by authorities for every smart intersection, fund controlled by the SC)
- Examples of technical elements:
  - Agreement on message types and usage (e.g., ETSI C-ITS messages)
  - An open standard for the exchange of real-time messages with field

[29] The Talking Traffic home page: https://dmi-ecosysteem.nl/en/theme-page-urban-traffic/talking-traffic/.

[30] Access to the documents at National iVRI standards - CROW is free of charge. But to have the access, one needs to create an account.

[31] Many elements of the CCoC can be found at https://www.crow.nl/thema-s/smart-mobility/landelijke-ivri-standaarden

equipment called the C-ITS subject interface (SI)[32], adopted by all suppliers of traffic signal controllers

- Quality levels/KPIs on uptime, connection quality (clock synchronisation, latency), message conformity and use-case quality
- Latency budgets, for the individual components as well as a target for the end-to-end latency
- Standards on interoperability (open interfacing only, no custom end-to-end solutions, no silos)
- Security arrangements: TLS, PKI, MFA etc.

During these processes the Ministry of Infrastructure and Waterworks procured a platform (Information Sharing Instance) for data exchange, data quality control, stakeholder dashboarding, governance, and the enforcement of security and privacy – open for use by all participating authorities.

After the initial development phase, a large-scale deployment of the services followed. By October 2023, this programme connected field equipment and traffic management from over 50 authorities with over 25% of motorised vehicles in The Netherlands. Data is shared bi-directionally leading to a daily exchange of over 1.3 billion messages.

During the deployment many lessons were learned, and significant changes and additions were made in the initial CCoC. With the foundation in place, a set of inter-urban use cases was selected for large-scale deployment, such as wrong way driver warning, emergency vehicle awareness, jam-tail warning, and road inspector vehicles/ shock absorbers in action. Also, a testbed was created and a process for certification of digital services with field equipment was realised. All these changes were initiated and supervised by the public-private governance structure.

Figure 29 shows the high-level overview of Talking Traffic architecture.

[32] C-ITS Subject Interface: https://www.citsinterface.org/

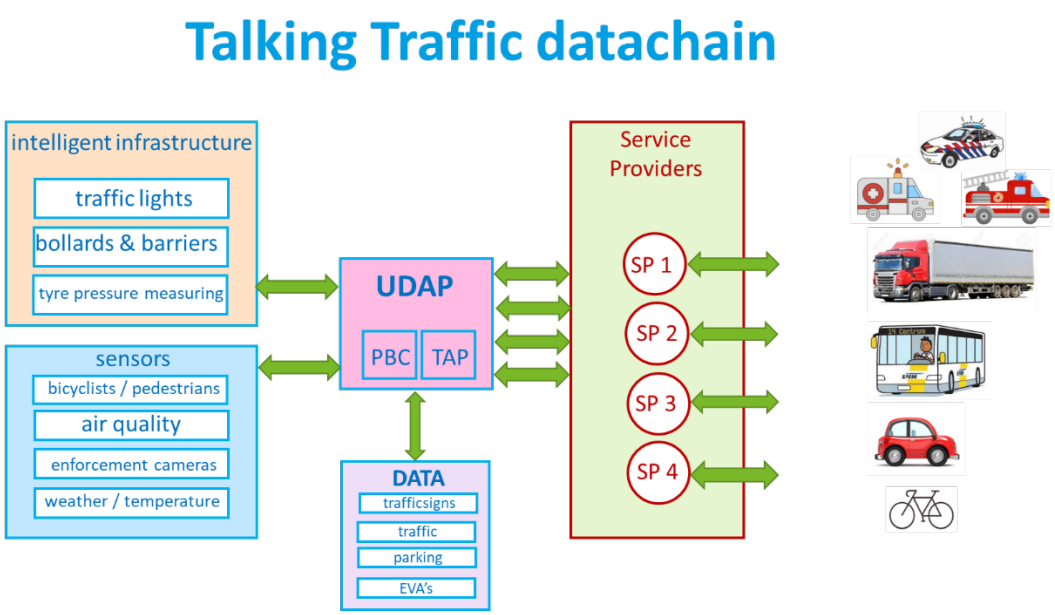


Figure 29: High-level overview of Talking Traffic

(Source: Dutch Ministry of Infrastructure and Waterworks)

The (public) intelligent infrastructure and the central Urban Data Access Platform (UDAP) is in the domain on the authorities. The Service Providers are commercial organisations that consume but also share data with the authorities through the UDAP platform. Road users are connected to the service providers mainly with 4G.

Data is exchanged bi-directionally and in real time at large scale. The currently supported message types are SPaT, MAP, CAM, SRM, SSM, IVI and DENM. Almost all cities, regional authorities and the national highway operator are connected to UDAP. Connecting service providers include Be-Mobile, Yunex Traffic, and TomTom. Other parties such as INRIX, KIA/Hyundai and the ANWB have announced they will also connect to UDAP.

As such, UDAP is a real word example of a public Information Sharing Instance. Currently UDAP exchanges around 1.4bn messages per day with an average end to end latency around 150ms.

Service providers connect to the UDAP entity using a national open standard named the "UDAP Service Provider Interface". Data is exchanged through TCP channels. These channels are managed with an orchestrator API which provides the functionality to fully manage the channel (setup, scope, change, terminate). This specific interface has been created because at the time Talking Traffic was launched the C-Roads BI was still under development. This interface also supports specific messages to manage the connection, e.g., uptime, clock sync and roundtrip latency. Another difference with C-Roads is that the messages are not signed. Trust is realised through a strict set of arrangements including organisation and product certification, privacy agreements, and the use of organisation and object tokens.

More information about the Talking Traffic solution can be found in the presentation from the Dutch Ministry of Infrastructure and Water Management made in a 5GAA event[33].

B.3 Mobilidata Information Sharing Domain Principles

Solutions MobiliData[34] are built on the C-Roads principles as described in Annex B.1, but they also use the additional interfaces from Talking Traffic, e.g., to Traffic Light Controllers.

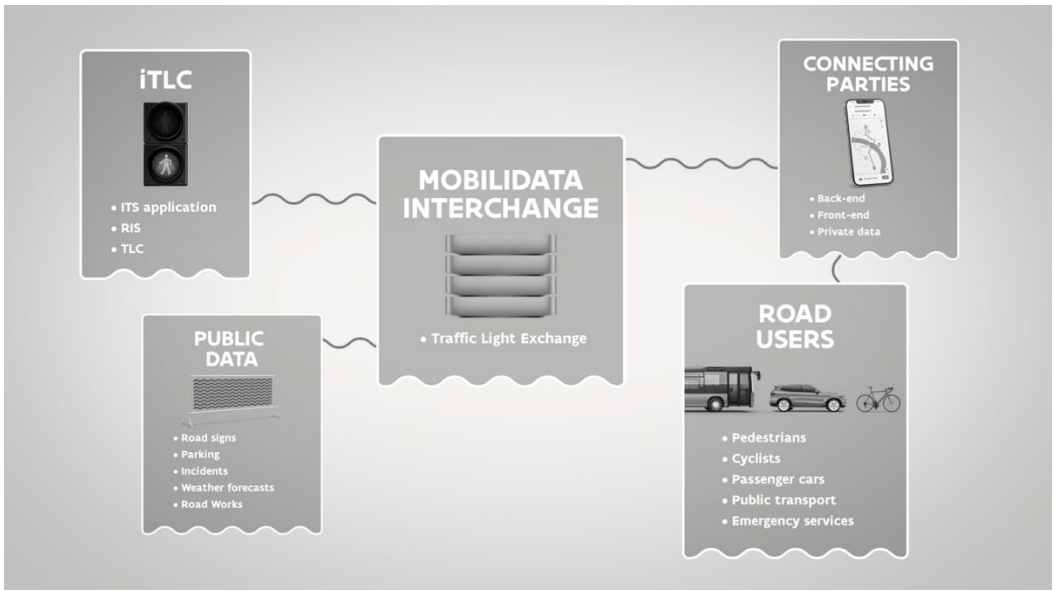


Figure 30: Mobilidata, high-level overview

(Source: Mobilidata/Agentschap Wegen en Verkeer, AWV)

Similar to Talking Traffic, the authorities have taken the responsibility for the public data and the central Information Sharing Instance (called the Mobilidata Interchange).

[33] https://5gaa.org/events/25th-5gaa-f2f-meeting-week-2/

[34] Mobilidata home page is Mobilidata targeted driving advice and intelligent mobility; a white paper providing an overview on Mobilidata can be found at mobilidata-eng-whitepaper-mob-architecture-june202.pdf

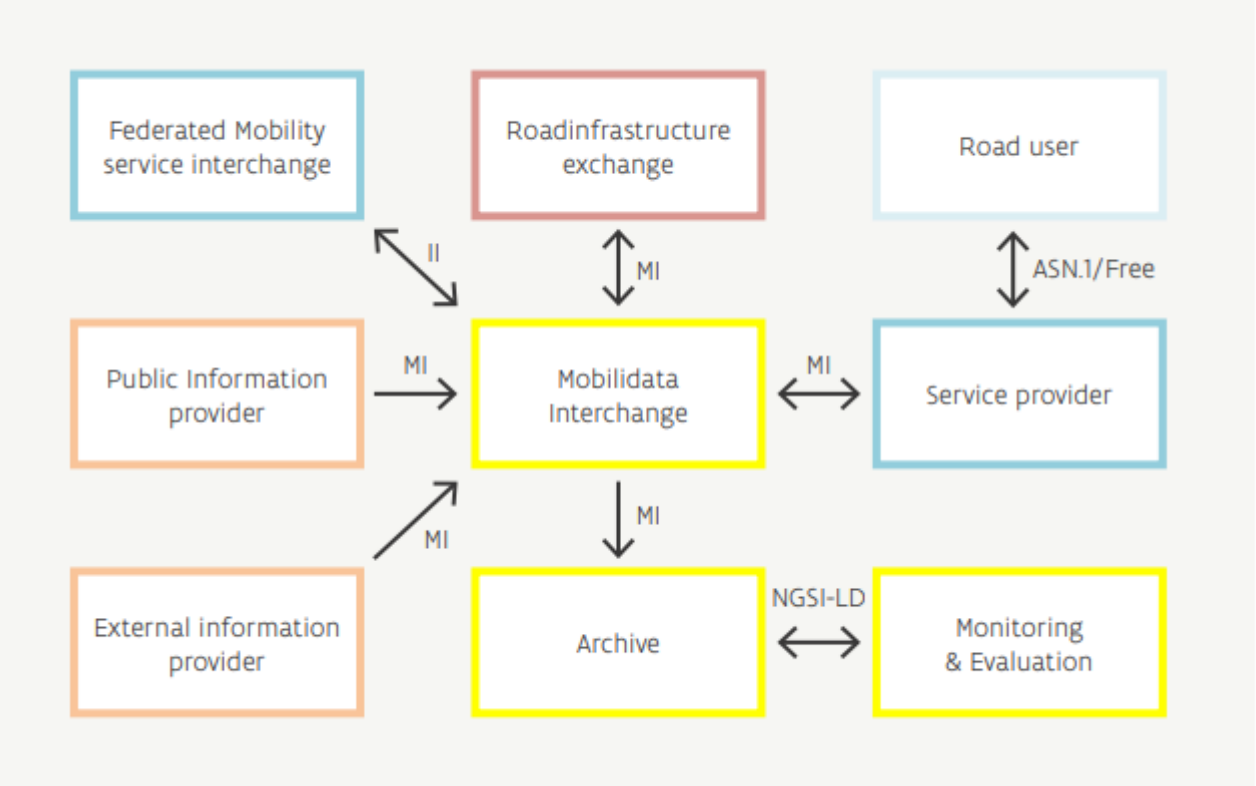


Figure 31: Mobilidata overview of components and interfaces

Data is exchanged with the Connecting Parties (Service Providers) using an interface called the MI. The MI is a superset of the C-Roads defined BI interface, meaning it is the BI with the additional possibility to exchange DatexII messages. Trust is realised in a similar way as in Talking Traffic.

Currently, the Service Providers Mobilidata are connected. Mobilidata has an ambitious roadmap in terms of use cases (see below for a summary) which is attracting the interest of more parties looking to connect to the environment.

The following UCs are part of the Mobilidata roadmap 'Use Case Functional Analysis, M0001':

- Static and Dynamic Speed Limits
- Static Road Signs
- Priority Vehicle Warning
- Slow Moving Vehicle Warning
- Accident/Vehicle Breakdown Warning
- Slow Emergency/Safeguarding Vehicle Warning
- Slippery Road Warning
- Person/Animal on the Road Warning
- Spilled Load Warning

- Traffic Jam Ahead Warning
- Road Works Warning
- Wrong-way Driving Warning
- iTLC Time-to-Green Information and Speed Advice
- iTLC Priority Emergency Vehicle
- iTLC Prioritising Public Transport
- iTLC Prioritising Vehicle Convoy
- iTLC Prioritising Truck (HGV)
- iTLC Traffic Signal Optimisation
- Recommended Routing
- Truck (HGV) Parking Information
- Park and Ride Facility Information

Note: iTLC stands for Intelligent Traffic Light Controller.

Annex C: 'Talking Traffic' message frequency profile

Below is an extract for the CAM, SPaT and MAP message frequency profiles used for TSI sharing using cellular mobile networks found at: CROW Kennisbank[35].

MAP

MAP data shall be transmitted:

- Upon connection
- On change
- At regular intervals (1-24 hours)

SPaT

SPaT data shall be transmitted:

- On change with a maximum frequency of 10Hz
- At least once every 10 seconds (i.e., retransmit in case SPaT data has not changed)

CAM

CAM data shall be transmitted:

- When the CAM data is relevant for the iTLC
- With a maximum frequency of 10Hz
- With a minimum frequency of 0.1Hz (i.e., once per 10 seconds)
- With a frequency of 1Hz for vehicles on the MAP

Relevant CAM data

- For (non-priority) vehicles (including cyclist and pedestrians) the CAM data is relevant
- If the vehicle is expected to reach intersection within 120 seconds (under free flow conditions)
- If the current location overlaps with a lane that is accessible to the vehicle type or is within the conflict area of the MAP

Under the following conditions the CAM data shall not be send to the iTLC:

- If the vehicle is stationary for more than 15 minutes
- If the vehicle can only be mapped on a lane(s) where the vehicle type is not allowed
- If the vehicle is marked invalid
- If CAM data is being simulated, unless explicitly authorised by the road authority

[35] Access to the documents at National iVRI standards - CROW is free of charge. But to have the access, one needs to create an account.

Stationary

If the positions of a vehicle overlap within the (varying) GPS accuracy, the vehicle is considered stationary until a displacement greater than the GPS accuracy is detected.

Invalid

Under the following conditions a vehicle shall be marked as invalid:

- If the timestamp is older than 2s, or more than 500ms in the future
- If the vehicle type changes during a trip
- If the instantaneous speed or acceleration is higher than plausible for the stated and/or known vehicle characteristics of the user (e.g., 50 km/h for a cyclist)
- If the average speed between two consecutive points of a vehicle is higher than plausible for the specified vehicle characteristics
- If the location data comes from the 2nd, 3rd, nth device in the same vehicle; if in this case one of the devices is a professional device to request priority, this device should always be marked as valid

A vehicle that is marked invalid shall remain invalid until the data is continuously valid for at least 2 minutes.

Prevention of spoofing

Cluster 2/Cluster 3 must prevent simulated data from being passed on to iTLCs within the production domain, unless it concerns simulated trips by road authorities. Examples of possible ways to detect spoofing are:

- Using functions in the platform (of the device) to detect or block simulated position data shall be used (e.g. standard on Android).
- If the location data is collected on a device in which data from an accelerometer, gyroscope and/or electronic compass is also available, then illogical data combinations shall be examined,
  - GNSS receiver indicates motion, while accelerometer does not indicate any vibration or acceleration.
  - The heading of the GNSS receiver deviates more than 30 degrees from the direction of the electronic compass.
  - Changes in the heading of the GNSS receiver do not match the heading changes registered by the gyroscope.

Accuracy

The data frame *positionConfidenceElipse* shall be used to convey inherent uncertainties in the data. It enables the entering of two values, while a GPS device usually only returns one value.

The *semiMinorConfidence* shall be used for conveying accuracy as provided by the GPS, whereby the value *unavailable* is not allowed.

If only the *semiMinorConfidence* is provided the *positionConfidenceEllipse* has the shape of a circle, as shown in the figure below.

In addition, the *semiMajorConfidence* and the *semiMajorOrientation* can be used to report the deviation estimated by the Service Provider. In this case, the *semiMinorConfidence* and *semiMajorConfidence* together create the ellipse shape as is intended by the standard.

Note that the major axis can be shorter than the minor axis. The *semiMajorOrientation* (indicated by Azimuth in the figure below) serves two purposes: one being to indicate the orientation (rotation) of the ellipse, whereas the other is to indicate the location as estimated by the service provider.

The delta between the service provider estimated location and the location of the GPS device is equal to the length of the major axis, and specifically in the direction indicated by the *semiMajorOrientation* (the green dot in the figure below). If the Service Provider cannot determine a good estimate of the deviation, these fields should be set to "unavailable".

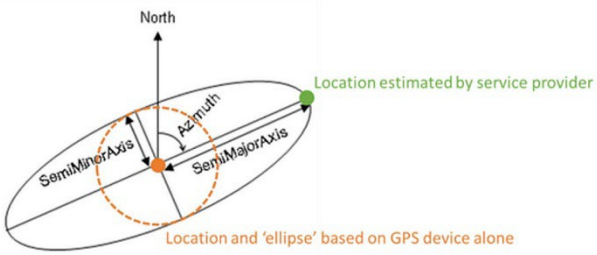


Figure 32: Schematic of usage of positionConfidenceEllipse

Note: Android provides 68% accuracy data by default (1 standard deviation) and must therefore be multiplied by a factor of 2 to meet the ETSI definition (95% corresponds to 2 standard deviations).

Stabilised heading

At very low speeds (5km/h) or standstill, speed and heading sometimes show random behaviour, making map matching difficult. In this situation a stabilised heading shall be delivered, appropriate to the assumed vehicle motion or heading.

Load reduction

The following logic shall be applied to reduce bandwidth and to save battery power:

- If constant velocity (+/- 5km/h) the frequency shall be halved to a minimum of 0.1Hz.
- If velocity changes >5km/h, the frequency shall be doubled to a maximum of 1Hz.
- If velocity changes >10km/h, updated CAM data shall be transmitted immediately, and the frequency shall be reset to 1Hz.
- If the heading changes >45 degrees, updated CAM data shall be transmitted immediately.
- If the in-vehicle device battery capacity is 33% and the device is not charging, all frequencies shall be halved to a minimum of 0.1Hz.

Annex D: Georeferencing Method – Quadtree

The principle is to calculate keys that represent tiles in a quadtree grid. This system is used by Bing Maps [7] and Here [6] under the name of "quadkeys" (short for quadtree keys) and also mentioned in the C-Roads Platform hybrid communication specifications [4]. A Java implementation is publicly available[36]. Quadtree is also explained in Wiki[37].

As shown in the following figure, each tile of the quadtree grid has a unique quadkey. The length of a quadkey corresponds to the zoom level, and the quadkey of a tile always starts with the quadkey of its parent tile. In Figure 33, tile 2 is the parent of tiles 20 to 23, and tile 21 is the parent of tiles 210 to 213.

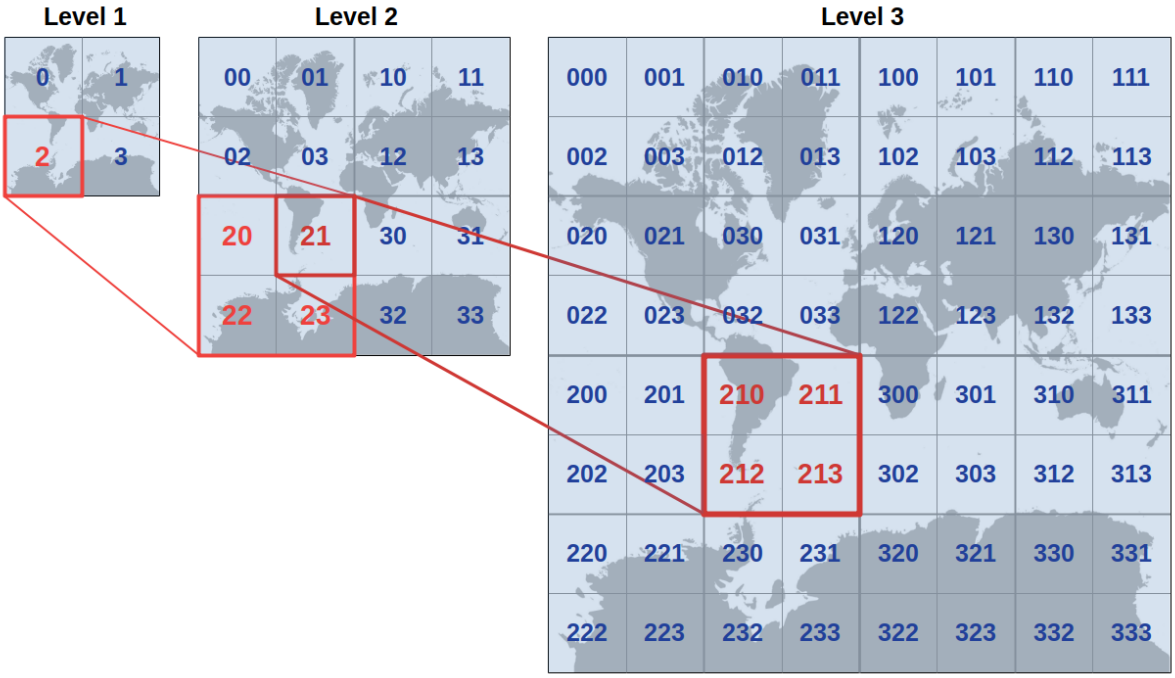


Figure 33: Quadkey numbering system [5]

These properties are compatible with the hierarchical pattern of message queueing protocol topics, e.g., in MQTT, which renders the integration of quadkeys directly into MQTT topics very easy, allowing publishing and subscribing of specific tiles. For AMQP "Application properties" (metadata) is used to indicate quadtree tile(s) when publishing and for filtering when subscribing. For example, to subscribe to all DENM messages in the tile with quadkey number 12022, the following topic extension can be used: … / … / 1 / 2 / 0 / 2 / 2

If a DENM is published in the tile with the quadkey 120220 (one of the child tiles of tile

[36] At https://github.com/passchieri/Hybrid-IF2

[37] QuadTiles - OpenStreetMap Wiki

12022), then it will be received by users that have subscribed to it and all its parent tiles as well (12022, 1202, 120, 12 and 1). For AMQP the DENM would thus be published with an "Application property" (e.g., named *quadTree*) that is equal to 120220, which would be delivered to users that have a subscription to *quadTree* 120220.

Annex E: 3GPP QoS assurance and Network Slicing mechanisms

Given the demands on QoS support and Network Slicing driven by the current and emerging advanced V2X services, there are on-going efforts in the telecom industry, e.g., the CAMARA [15] initiative, for making the standardised 3GPP features easily accessible by different industry segments. This Annex provides further technical details about 3GPP defined QoS mechanisms (E.1) and Network Slicing (E.2).

E.1 Overview of 3GPP QoS assurance mechanisms in 4G and 5G systems

Figure 34 illustrates the different 3GPP-defined QoS assurance mechanisms [8]:

- Network Slicing is defined in 3GPP as a logical network that provides specific capabilities and network characteristics. It is a tool to separate resources and provide defined network characteristic, for example an industry vertical which facilitates use-case differentiation and secures the necessary capacity and performance to meet Service Level Agreements (SLA) even in high-demand situations (heavy network load).
  - Note: Unless 4G QoS Class Identifier (QCI) or 5G QoS Identifier (5QI) values standardised in 3GPP [8] are used without modifications, the same QCI or 5QI value may have different behaviours in different Network Slices. The sub-sections on "Network Slicing" below provide more details about how a UE can use Network Slicing for V2X applications like Automated Valet Parking.
- A Packet Data Unit (PDU) session needs to be established when the UE has packets to transmit. One or more PDU sessions can be established within one Network Slice.
- For one PDU session, multiple QoS Flows can be defined. The number of simultaneously active QoS Flows is typically limited.
- One or more Applications Flows[38] can be contained within one QoS Flow. Application Flows based on separation and prioritisation allow traffic to be differentiated by characteristics like priority, Packet Error Rates (PER), Packet Delay Budgets (PDB), Guaranteed Bitrate (GBR), Delay Critical GBR, non-GBR, etc.

[38] 'Application Flow' refers to data traffic of an application that certain QoS policy can be applied. Application Flow can be described using descriptors e.g. IP 5-Tuple.

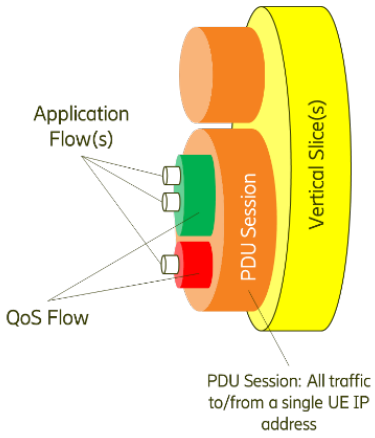


Figure 34: 3GPP QoS assurance mechanisms

With respect to Quality on Demand (QoD)/Quality of Service (QoS) APIs, these should be radio-access technology agnostic. Therefore, depending on the local deployments of the MNOs, the QoD API might be available in 4G, 5G, or both.

It is important to note that all described QoS mechanisms are **working on an application level, and not device level.** So, different applications might make use of different Network Slices, and some applications might use a QoD API while others may not. This also addresses the needs of automotive applications with different QoS requirements because they are operated in parallel (e.g., an AVP application is executed while at the same time Mobile Broad-Band (MBB) data traffic and status information is transmitted to the vehicle backend, or a map download is performed).

Even when the network is delivering the requested QoS, the actual QoS performance may change due to the RAN being temporary unable to fulfil it. The network has mechanisms to handle such events, e.g., Alternative QoS Profile, QoS Sustainability analytics, and QoS monitoring. Additional proper network planning and QoS/priority assignment can also reduce the probability of such events.

E.2 Network Slicing

A cellular network architecture comprises a number of function-providing network nodes, with different node configurations and purpose, deployed at potential different physical instances and geographic locations throughout an MNO national network structure. With the evolution of system and network technologies and with the large-scale introduction of hardware virtualisation technologies and (cloud native) deployment options, many of such network nodes were re-implemented with virtualisation technology underneath. This allows flexible and even dynamic node (software) deployments and a multitude of network configurations, without requiring function-specific hardware (re)installations.

Different such network deployment structures inherit different embedded network (performance) characteristics. For example, having certain network nodes deployed closer to the connecting UEs and using IP network under the control of the MNO may reduce the latency experienced for those UEs. A MEC deployment is one such

example. On the other hand, fewer and more centralised node deployments may reduce deployment and operation cost for the MNO in question.

The introduction of virtualisation technologies at the MNO core network started already with 4G LTE core networks, resulting in the virtual evolved packet core (vEPC) network deployment concept. With this evolutionary technology step it became possible to have more than one vEPC structure deployed with a single MNO network, and to separate data traffic and network usage per different vEPCs and between the corresponding internal MNO network structures. One can consider such vEPC deployments, a 4G network design concept, as an early version of a cellular Network Slice.

A Network Slice, as considered today, refers to a certain cellular network node deployment structure. Each structure (alias "Network Slice") constitutes on the user-plane a fully functional network architecture. Few network nodes provide shared internal services to several Network Slices. Examples are network nodes handling the user- and subscription administration, or network operation tasks. User-plane deployment structures of virtualised network nodes, with different node configurations and deployment locations, can exist in parallel. Thanks to the virtualisation technology underneath, these different virtual networks (slices) operate in full logical separation to each other. Within each such virtual network (slice) different PDU sessions with different QoS Flow characteristics, carrying different application flows therein, can be established. An extreme configuration of a Network Slice structure would be if it is configured to handle all its internal dataflows and user-plane sessions in the same way. Figure 34 illustrates a sample structure of different flows, established within one such Network Slice.

It shall be noted that a cellular Network Slice as such is referring to a specific network node deployment structure. This implies that a Network Slice itself is not providing any end-to-end connection for a UE or for UE applications. This in turn leads to the question if and how a UE (modem) can attach to a given Network Slice, or to multiple Network Slices simultaneously, and how certain applications, residing at the UE, could establish their communication flow(s) within one or within another Network Slice, available to the connected UE. In turn, this carries the question of how to address a certain Network Slice at a given MNO network, and how to know which Network Slices, with which embedded characteristics, are available at a given MNO network, and to the connecting UE, based on the UE type and SIM subscription.[39]

Different mechanisms have been standardised, or can be utilised, to address Network Slices in a dynamic or in a static way. The following subsections introduce those mechanisms on conceptual levels.

E.2.1 Slice selection with URSP rules

UE Route Selection Policy (URSP) provides a foundation to deliver dynamic Network Slice selection, enabling traffic steering and the separation of end-to-end services for devices and for client software components (client services or applications) deployed at a given device. When devices are being provided with URSP capabilities, the UE is able to use Network Slices according to the policies defined for that subscription. This concept links, in fact, URSP rules to the Network Slices of the connecting UE with the user subscription of that UE.

[39] GSMA: TS.62 UE Requirements Related to Network Slicing using Requirements URSP. Version 1.0, 9th November 2023, https://www.gsma.com/newsroom/wp-content/uploads//TS.62-V1.0-UE-Requirements-related-to-network-slicing-using-URSP-1.pdf

The network offers the information about available slice types to the device via URSPs, so the URSP adds further details regarding which Network Slices the device's underlying applications should use when activated. URSP rules thereby abstract from the technical details of the connectivity proving MNO network and from the particular deployment structures of the MNO Network Slices. See [9] for a further description of URSP. Therefore, the device knows in advance of a certain application process which slice types are available, and how to get access to the relevant slice type for the client application. Applicable slice(s) to be used need to be discussed with the corresponding UE connectivity providing MNO and be activated as "allowed Network Slices" for the UE's SIM profile.

If the UE is in a roaming context the Network Slice selection via URSP rules becomes somewhat more complicated because the UE subscription and its attached permissions is bound to the home network (home MNO). The network that provides the cellular connectivity to the roaming UE is in fact the visited network (by the visited MNO). At run-time, the visited network would provide the various Network Slices with their corresponding deployment structures. The relationship between the visited network and the home network, and the UE's SIM, is via a roaming agreement signed between the home MNO and the visited MNO(s).

In an automotive V2X context there are typically many different client services, active simultaneously, at the same UE device e.g., Telematic Control Unit (TCU), sharing the same cellular network modem (UE) and the same physical cellular network connection. If different client software components should attach to different Network Slices, available to the UE, the URSP rules would provide the information about the available Network Slices. A devices operating system at the UE would map the client "application identifier" (App-ID)[40] to the corresponding URSP rule, and thereby indirectly to the corresponding Network Slice at the connectivity providing MNO network.

This concept assumes that the UE device in fact has an operating system, or a similar function, that can map the App-IDs to the URSP rules, available to the UE. And it assumes that an App-ID expresses the network characteristics as required by the corresponding client software. This URSP concept, and its prerequisites, are assumed to be available for the smartphone segment with its ecosystems of apps. GSMA TS.62 (Nov. 2023) and 3GPP TS 24.526 provide more details on the UE requirements related to Network Slicing using URSP rules.

E.2.2 Slice selection with SIM profile

In a very simplified Internet of Things (IoT) ecosystem structure, all UE software-clients, or the one IoT device (hardware) function, has a static mapping to a best fitting Network Slices structure, with a corresponding URSP rule or a corresponding Network Slice ID, called Single – Network Slice Selection Assistance Information (S-NSSAI) in 3GPP terms, assigned. The linkage between the Network Slice ID (or URSP rule) and the UE is configured at the UE's SIM profile.

Whenever such an IoT configured UE connects to a cellular network, the corresponding SIM profile is consulted, and the corresponding Network Slice gets attached to the UE (modem). All software (or hardware) client functions activated at such an IoT device would utilise the same physical and virtual Network Slice configuration. Meaning that

[40] Application identifiers for URSP rules are defined at 3GPP TS 24.526 (stage 3, Rel. 18, Dec. 2023)

all client services would experience the same cellular network characteristics for the time the UE stays connected.

E.2.3 Slice selection with S-NSSAI requests

The 3GPP TS 23.501 describes how a 5G system supports Network Slicing. A Network Slice, according to TS 23.501, is identified by an S-NSSAI, which is comprised of a slice/service type (SST) and a slice differentiator (SD). The inclusion of an SD in an S-NSSAI is optional. A set of one or more S-NSSAIs is called the NSSAI. TS.62 (GSMA, Nov. 2023) provides more details on the UE requirements related to Network Slicing and the NSSAI concept for addressing standardised and non-standardised cellular Network Slices.

In a nutshell, a given cellular Network Slice can be associated with an S-NSSAI as unique identifier. The USRP concept, outlined at E.2.1, maps traffic descriptors to route selection descriptors, where the latter may contain the S-NSSAI values.

Even in cases when such a URSP rule-mapping function is not available at the UE, a PDU session, within a given cellular Network Slice at the connectivity providing MNO, can still be established. End-user communication flows can be established thereafter within the provided PDU session, including QoS requirements.

The direct establishment of a PDU session within a given Network Slice requires knowledge, at the UE (modem), of the network slide identifier (S-NSSAI). With this knowledge AT commands[41] can be used to implement such a request. An example of how to apply the 'CGDCONT' command for requesting a PDU session establishment within a given S-NSSAI is provided in Table 2.

[41] How to use AT-commands for requesting a certain Network Slice by its S-NSSAI number. Source: Tech-invite, a 3GPP and IETF space; https://www.tech-invite.com/3m27/toc/tinv-3gpp-27-007_x.html

Command	Possible response(s)
+CGDCONT=<cid>[, <PDP_type>[, <APN>[, <PDP_addr>[, <d_comp>[, <h_comp>[, <IPv4AddrAlloc>[, <request_type>[, <P-CSCF_discovery>[, <IM_CN_Signalling_Flag_Ind>[, <NSLPI>[, <securePCO>[, <IPv4_MTU_discovery>[, <Local_Addr_Ind>[, <Non-IP_MTU_discovery>[, <Reliable_Data_Service>[, <SSC_mode>[, <S-NSSAI>[, <Pref_access_type>[, <RQoS_ind>[, <MH6-PDU>[, <Always-on_req>[, <old-cid>[, <ATSSS-ST>[, <LADN-DNN_ind>[, <MA-PDU-session-information>[, <Ethernet_MTU_discovery>[, <Unstructured_Link_MTU_discovery>[, <PDU_Pair_ID>[, <RSN>[, <ECSConf_info_ind>[, <EDC_support>[, <SDNAEPC_support>[, <EAS_redisc_supp_indEDC_support>[, <SDNAEPC_support>]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]	
+CGDCONT?	[+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[, <IPv4AddrAlloc>[, <request_type>[, <P-CSCF_discovery>[, <IM_CN_Signalling_Flag_Ind>[, <NSLPI>[, <securePCO>[, <IPv4_MTU_discovery>[, <Local_Addr_Ind>[, <Non-IP_MTU_discovery>[, <Reliable_Data_Service>[, <SSC_mode>[, <S-NSSAI>[, <Pref_access_type>[, <RQoS_ind>[, <MH6-PDU>[, <Always-on_req>[, <old-cid>[, <ATSSS-ST>[, <LADN-DNN_ind>[, <MA-PDU-session-information>[, <Ethernet_MTU_discovery>[, <Unstructured_Link_MTU_discovery>[, <PDU_Pair_ID>[, <RSN>[,<ECSConf_info_ind>>[, <EDC_support>[, <SDNAEPC_support>[, <EAS_redisc_supp_ind>]]]]]]]]]]]]]]]]]]]]]]]]]]]] [<CR><LF>+CGDCONT: <cid>,<PDP_type>,<APN>,<PDP_addr>,<d_comp>,<h_comp>[, <IPv4AddrAlloc>[, <request_type>[, <P-CSCF_discovery>[, <IM_CN_Signalling_Flag_Ind>[, <NSLPI>[, <securePCO>[, <IPv4_MTU_discovery>[, <Local_Addr_Ind>[, <Non-IP_MTU_discovery>[, <Reliable_Data_Service>[, <SSC_mode>[, <S-NSSAI>[, <Pref_access_type>[, <RQoS_ind>[, <MH6-PDU>[, <Always-on_req>[, <old-cid>[, <ATSSS-ST>[,<LADN-DNN_ind>[, <MA-PDU-session-information>[, <Ethernet_MTU_discovery>[, <Unstructured_Link_MTU_discovery>[, <PDU_Pair_ID>[, <RSN>[, <ECSConf_info_ind>>[, <EDC_support>[, <SDNAEPC_support>[, <EAS_redisc_supp_ind>]]]]]]]]]]]]]]]]]]]]]]]]]]] [...]]]

Table 2. AT command CGDCONT used to request a certain Network Slice by its S-NSSAI number
(Source 3GPP TS 27.007 V18.8.0)

The 'CGACT' command can also be used to activate a bearer resource for 4G evolved packet system (EPS). According to TS 23.501[42] and TS 24.501[43], a one-to-one mapping between a 5G system (5GS) PDU session and an EPS PDN connection exists. A 5GS PDU session is a set of QoS Flows consisting of one QoS flow of the default QoS rule, e.g., MBB traffic, and optionally one or more QoS Flows of the non-default QoS rule. A Packet Data Network (PDN) connection is set of EPS bearer contexts and consists of at least one default EPS bearer context and optionally one or more dedicated EPS bearer contexts. A PDU session can be mapped to one default EPS bearer context and zero or more dedicated bearer EPS bearer contexts. An EPS bearer context can be mapped to one or more QoS flows. The mapping between a QoS flow and an EPS bearer context is not always one to one.

E.2.4 Slice selection with APN names

Every cellular network deployment structure is carrying (at least) one specific gateway node that carries the end-to-end data traffic (user-plane) from/to a connecting UE to application servers outside of the MNO network domain, e.g., to the public internet or to a specific AS at a given enterprise. In a 4G (LTE) network such gateway node is called P-GW (packet gateway). The corresponding node in a 5G core network is called User Plane Function (UPF)[44].

[42] TS 23.501 https://www.tech-invite.com/3m23/tinv-3gpp-23-501.html

[43] TS 24.501 https://www.tech-invite.com/3m24/tinv-3gpp-24-501.html

[44] "What is the 5G User Plane Function (UPF)?" (source)

This relationship between a given network deployment structure, including a virtual network deployment structure (Network Slice), to the corresponding use-plane expose gateway node or function embeds another method of identifying a given (virtual) network deployment structure. Namely, by addressing its corresponding exposure gateway (P-GW or UPF).

A well-established schema for addressing different P-GWs is by using so call Access Point Names (APN). This 4G network concept has been specified in 3GPP Rel. 8 and has seen widespread usage thereafter. The same schema can also be used for addressing a certain UPF, using a 5G Data Network Name (DNN). Hence this concept embeds another schema for addressing Network Slices, albeit not based on S-NSSAIs, e.g., by directing UE data traffic, or UE data traffic of some kind, to a corresponding APN or DNN.

An APN Network ID typically has a format similar to an universal resource locator (URL), e.g., data.my-MNO.se. It constitutes as so called Fully Qualified Domain Name (FQDN). See Figure 35.

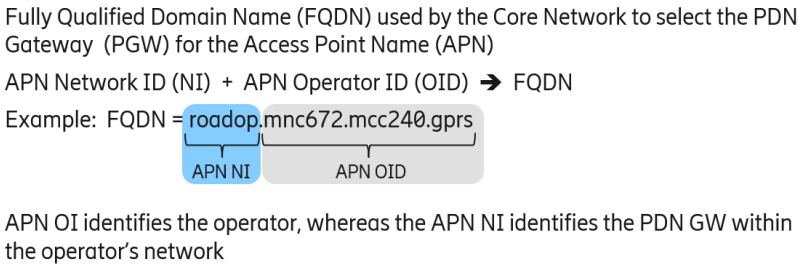


Figure 35: Fully Qualified Domain Names (FQDN) and APN Names

In contrast to using URSP rules for the mapping of UE client software services to Network Slices, requiring an operating system or mapping function at the UE, corresponding data traffic can be routed in a similar way via above APN name conventions. Additional tools, such as the 3GPP QoS framework, may be applied for traffic flows within a given Network Slice. (See Annex E.1.)

E.2.5 Global mobility aspects

When more than ordinary MBB connectivity are required, additional aspects need to be considered. The global automotive and V2X segment differs quite a bit from established MBB ecosystems and usage patterns of cellular network technology. The differences are rooted in, on the one hand, OEMs who operate connected vehicle services from their centralised application servers – sometimes in cross-country structures – and, on the other hand, vehicles are produced, sold, and operated in many countries and global regions. The very long lifecycle of deployed vehicles and the nature of vehicles being sold and re-sold and driving across borders leads to a high mobility pattern of UEs (vehicles), usually set in a roaming constellation, connecting and re-connecting to different visited MNO networks for home-routed connected vehicle services.

When it comes to vehicle services and service experiences, the overall expectations are that no matter where a given vehicle drives, and which MNO currently provides the vehicle's cellular connection, the resulting end-user service experience should still be satisfying and persistent. This raises the need for harmonised network capabilities and configurations cross MNOs that are made available through the above-described mechanisms, which need to be adopted and utilised by vehicle OEMs. This also points to the need for designing a "network-aware" vehicle (software) architecture for providing the wanted end-user experiences benefiting the network features and capabilities, e.g., QoS and local/regional breakout in visited MNO networks. The deployment solutions need joint efforts from the MNOs and vehicle OEMs.

Annex F: Logical interfaces in V2N2X application layer reference architecture

Table 3 describes logical interfaces in Figure 1, also known as reference points. For each interface Table 3 provides the following information:

- The system components that are connected via the interface.
- The type of services and information exchanged using the interface.
- Characteristic of the interface, e.g., intra-, or inter-stakeholder domain interface. (Stakeholder domains for vehicle OEM, IOO, and SP, are shown in the system architecture Figure 1.)
  - Note: Implementation of interfaces that cross different stakeholder domains, also known as inter-stakeholder domain interface, require agreed implementation profiles by the relevant stakeholders, to ensure interoperability of the V2X service.

The corresponding message formats and protocols as well as communication technologies used in the implementation of the interface depend on deployment options and the use case. Chapter 6 describes such details in the V2N2X solution blueprint. Chapter 8 provides further details for specific use cases based on the blueprint solution in Chapter 6.

Note: The protocols and messages to be used for the interfaces in Table 3 depend on use case and implementation solutions. For inter-stakeholder domain interfaces, the protocols and messages need to be negotiated and agreed between the connected parties. The “Example” column of the table provides the links to example implementations in Chapter 8 for inter-stakeholder interfaces.

Table 3: Description of logical interfaces in the V2N2X application layersytem architecture

Logical Interface	System Component 1	System Component 2	Services and information exchanged using the interface	Characteristics	Example implementation for inter-stakeholder interface
O1	OEM AS	OEM App	User data of the V2X application. Control data for the operation of the V2X application, e.g., application configuration, permission, security information, etc.	Intra OEM stakeholder domain interface	
O2	OEM AS	SP AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	**Inter stakeholder domain interface**	**Section 8.1 "Traffic event information sharing"; Section 8.2 "Traffic signal information sharing"; Section 8.3 "Traffic signal priority request sharing"; Section 8.4 "Emergency Vehicle Approaching"; Section 8.5 "HD MAP handling"; Section 8.7 "Object Detection and Sharing"; Section 8.8 "Vulnerable Road User protection";**
O4	OEM AS	OEM AS	User data of the V2X application among OEM AS(es) from the same or different OEMs. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	Intra or **Inter** OEM stakeholder domain interface	**Section 8.8 "Vulnerable Road User protection (VRU)";**

O5	OEM AS	Infrastructure Owner Operator AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	**Inter stakeholder domain interface**	**Section 8.1 "Traffic event information sharing"; Section 8.6 "Automated Valet Parking / Automated Vehicle Marshalling"; Section 8.7 "Object Detection and Sharing"; Section 8.8 "Vulnerable Road User protection";**
R1	Infrastructure Owner Operator AS	Infrastructure Owner Operator App	User data of the V2X application, e.g., sensor data, traffic signal data, etc. Control data for the operation of the IOO infrastructure.	**Intra IOO stakeholder domain interface**	
V1	Infrastructure Owner Operator AS	OEM App	User data of the V2X application. (Note, data communication of OEM App using V1 interface is usually under the control or with the permission of the OEM AS, e.g., via the O1 interface.)	**Inter stakeholder domain interface**	**Section 8.6 "Automated Valet Parking / Automated Vehicle Marshalling"; Section 8.7 "Object Detection and Sharing";**
V1'	Infrastructure Owner Operator AS	SP App	User data of the V2X application. (Note, data communication of SP App using V1' interface is usually under the control or with the permission of the SP AS, e.g., via the P1 interface.)	**Inter stakeholder domain interface**	**Section 8.7 "Object Detection and Sharing";**
V2	Infrastructure Owner Operator AS	Infrastructure Owner Operator AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	Intra or **Inter** IOO stakeholder domain interface	

P1	SP AS	SP App	User data of the V2X application. Control data for the operation of the V2X application, e.g., application configuration, permission, security information, etc.	Intra SP stakeholder domain interface	
P2	SP AS	SP AS	User data of the V2X application among SP AS(es) from the same or different SP(s). Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	Intra or **Inter** SP stakeholder domain interface	**Section 8.4 "Emergency Vehicle Approaching"; Section 8.8 "Vulnerable Road User protection";**
P3	SP AS	Infrastructure Owner Operator AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	**Inter stakeholder domain interface**	**Section 8.1 "Traffic event information sharing"; Section 8.2 "Traffic signal information sharing"; Section 8.3 "Traffic signal priority request sharing"; Section 8.4 "Emergency Vehicle Approaching"; Section 8.7 "Object Detection and Sharing";**
P4	SP AS	OEM App	User data of the V2X application. (Note, data communication of OEM App using P4 interface is usually under the control or with the permission of the OEM AS, e.g., via the O1 interface.)	**Inter stakeholder domain interface**	**Section 8.2 "Traffic signal information sharing"; Section 8.5 "HD MAP handling"; Section 8.8 "Vulnerable Road User protection";**

I1	Information Sharing Instance	Infrastructure Owner Operator AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	**Inter stakeholder domain interface**	**Section 8.1 "Traffic event information sharing"; Section 8.3 "Traffic signal priority request sharing"; Section 8.7 "Object Detection and Sharing";**
I3	Information Sharing Instance	OEM AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	**Inter stakeholder domain interface**	**Section 8.1 "Traffic event information sharing"; Section 8.4 "Emergency Vehicle Approaching"; Section 8.7 "Object Detection and Sharing";**
I4	Information Sharing Instance	SP AS	User data of the V2X application. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	**Inter stakeholder domain interface**	**Section 8.1 "Traffic event information sharing"; Section 8.3 "Traffic signal priority request sharing"; Section 8.4 "Emergency Vehicle Approaching"; Section 8.7 "Object Detection and Sharing";**
I5	Information Sharing Instance	Information Sharing Instance	User data of the V2X application, e.g. DENM. Control data for the operation of the V2X application, e.g., service discovery, application configuration, security information, billing information, etc.	Intra or **Inter** SP stakeholder domain interface	**Section 8.1 "Traffic event information sharing"; Section 8.3 "Traffic signal priority request sharing"; Section 8.4 "Emergency Vehicle Approaching"; Section 8.7 "Object Detection and Sharing";**

Annex G: Software system and operation design principles

- Design for flexibility, automation and IT best-practices:
  - Provide the system foundation for a growing set of use cases. Facilitate data and information flows between private and public entities, cross-industry, cross-service providers, and between cross-jurisdictional stakeholders.
  - Extend data elements for cross-domain communication with descriptive metadata (see Annex H). This facilitates machine-readable and automated processing with protocol conversions on the application level. It also helps to decouple software lifecycles and versioning between the various stakeholder systems and domains.
  - Allow proprietary protocols and data formats within a stakeholder domain (see P1 or O1 in Figure 1: e.g., for commercial or for stakeholders' client-server interactions.
  - Encourage a state-less and event-driven software-design pattern. Avoid period message repetitions and timeout dependencies.
- Utilise best-practices for communication protocols and application programming interface (API) technologies:
  - The V2N2X communication protocols should be IP-based and use standard IT technologies for security, e.g., TLS (for TCP) or DTLS (for UDP).
  - Between the V2N2X information-sharing instances, use HTTP REST APIs for federation of information and for process automation.
- Design for large-scale operation and cross-country/cross-state/cross-stakeholder interactions:
  - Avoid the need for many-to-many system integration efforts and stakeholder contract relations. A stakeholder that aligns with the V2N2X information-sharing domain would have indirect access and reach all networked stakeholders, without further integration effort.
  - For stakeholders to have their IT systems interacting with the V2N2X information-sharing domain, which constitutes a dedicated trust domain, they must provide confirmation/proof that they will adhere to the data-sharing governance model, superseding the V2N2X Information Sharing Domain (e.g., by signing a CCoC). The proof or evidence may trigger the appropriate authority (CA) to issue a digital certificate (permission) for the stakeholder to communicate with a V2N2X Data Sharing Instance.
  - Support functions for automation, system and information resilience, security and trust in exchanging data should all be based on interactions

via standard DNS for discovery of “approved” actors and on a CA for handing out standard X509 certificates to approved actors.

  - For scalability within a the V2N2X Information Sharing Domain use a “message queuing protocol” with a publish/subscribe mechanism for data-sharing, filtering or forwarding of data elements or queries; e.g. the standardised advanced message queuing protocol (AMQP).

- Keep the additional standardisation efforts minimal:
  - Allow use case specific data formats to travel via generic and well-established application-level communication protocols. Provide metadata with suitable data elements to facilitate the transcoding of data formats and interaction protocols.

For more information about AMQP, metadata and interoperability, see Annex H.

Annex H: AMQP, metadata and interoperability

For network communication, interoperability is on the application-level, not on the radio-level, so mobile users on cellular networks can use different radio technologies (e.g., 4G, 5G, and beyond), and fixed assets operated by IOOs can be connected by different wired communication technologies or via cellular. This means that a road user connected to a 4G cellular network, provided by one Communication Service Provider (CSP), can communicate with other road users on a 5G cellular network, provided by another CSP. Application servers provide the bridge between users on different CSP networks, using different generations of cellular networks. In fact, it is the application data (IP packets) passed from the user (device or vehicle) on the mobile network to an AS. The radio-specific parts of the protocols are only used within the mobile networks. The AS can then provide service-level interoperability, i.e., pass the application-level information on to other actors, such as external service providers and road operators, or convert the application-level information to an agreed format before passing it on.

The application itself should make use of well-defined ITS message sets, as they are standardised by SAE or ETSI on the application level. For example, hazard warnings messages can be described in DENM or TIM[45] format, signalised intersections conditions by SPaT/MAP messages, or traffic signal pre-emption by SREM/SSEM messages. Note: The message format on an application level can be re-used, however message frequency should be used in an adapted way.

To facilitate information filtering and/or data format conversions, the actual application-level information is tagged with metadata, which provides information about the actual application-level input. The suggested ISO standard advanced messaging queuing protocol 1.0 (AMQP)[46] is available from a number of vendors, including Linux distributions; AMQP refer to metadata as “application properties”.

Below is an example from the C-Road “IP-based interface profile”[47] of what such metadata can indicate. Left-most column “Name” is the metadata (application property).

[45] Traveller Information Message, as defined by SAE/J2735 Message Set Dictionary

[46] More information on the ISO standard advanced messaging queuing protocol can be found at https://www.amqp.org/

[47] C-Roads: “IP based interface profile”, which is part of Release 2.0.x of the C-Roads Harmonised C-ITS Specifications: https://www.c-roads.eu/fileadmin/user_upload/media/Dokumente/Harmonised_text_v2.pdf

Name	Value and type	Description	Mandatory/ Optional
publisherId	string A two-letter country code (based on ISO 3166-1 alpha-2) and a numerical identifier (value between 0 and 16383 including leading zeroes) based on ISO 14816:2005 (same as used for providerIdentifier in IVIM), e.g. «AT00001», «DE15608»	Unique ID of the publisher. It is linked to the country where the provider wants to register. It could be in one country or several.	M
publicationId	String Concatenation of publisherId and a unique identifier for the dataset/publication with a ":" in between, e.g. "DE15608:IVIM_BERLIN_067" or "NO73944:679ABX92"	Each dataset/publication identifier needs to be unique for the given publisher.	O
originatingCountry	string Country code (based on ISO 3166-1 alpha-2)	Country code where the C-ITS message is created	M
protocolVersion	string E.g. "DENM:1.3.1" or "IVIM:1.2.1"	Represent the version of standard used to create the message, i.e. for DENM the version of ETSI EN 302 637-3, for IVIM, SPATEM the version of ETSI TS 103 301	M
serviceType	string E.g. "HLN-RLX"	Acronym defined in latest version of Common C-ITS Service and Use Case Definitions	O
messageType	string DENM, IVIM, SPATEM, MAPEM, SREM, SSEM, CAM	For this version of the specification the string shall be one of the following: DENM, IVIM, SPATEM, MAPEM, SREM, SSEM, and CAM. The list may be subject to changes in future versions of the specification	M
longitude	float Decimal degrees According to WGS84/EPSG:4326	Longitude of the event published; for DENM (eventPosition) and for IVI and SPATEM/ MAPEM/SSEM/SREM (referencePosition)	O
latitude	float Decimal degrees According to WGS84/EPSG:4326	Latitude of the event published; for DENM (eventPosition) and for IVI and SPATEM/ MAPEM/SSEM/SREM (referencePosition)	O
quadTree	string Comma separated list of quadtree tiles starting and ending with a comma, e.g. ",2023201202321201 01," (single value) or ", 2023201202321201 01,2023201202321201 02,2023201202321201 03," (multiple values chained)	Relevant spatial index location of the C-ITS message	M

Table 4: Metadata (AMQP application property) example

Metadata (application properties) can be user defined for AMQP and thus tailored to the needed applications and operation. In this example the metadata is tailored for C-Roads use with ETSI-type messages, as can be seen in the row *messageType*. An actor publishing information to an information-sharing instance thus includes these "application properties". An actor subscribing to an information-sharing instance provides a filter[48] of what information it is interested in. For example, if only ETSI DENM messages of a certain revision are supported by an actor, the filter would indicate that if that actor publishes information matches the filter, this information is pushed to the subscribing actor. Further filters and subscription properties could, for example, be using quadTree[49] (bottom metadata in Figure 37) to provide only road traffic information with relevance to a certain geographic area. Quadtree is further explained in Annex D.

[48] AMQP uses 'Structured Query Language' (SQL) for filter expressions, this mean that powerful conditions can be expressed, e.g. including 'And', 'Or', 'If', Comparison operators etc.

[49] QuadTiles - OpenStreetMap Wiki

Annex I: Document history

Date	Version	Subject/Comment
2024-04	V1.0	First public release of this TR.
2025-05	V2.0	Updated architecture and corresponding figures and text to include IOO App and the R1 interface.

5GAA is a multi-industry association to develop, test and promote communications solutions, initiate their standardisation and accelerate their commercial availability and global market penetration to address societal need. For more information such as a complete mission statement and a list of members please see https:/5gaa.org


5GAA
Automotive Association