Draft ETSI GS MEC 030 V3.0.15 (2023-01)

Multi-access Edge Computing (MEC); V2X Information Services API

Disclaimer: This DRAFT is a working document of ETSI ISG MEC. It is provided for information only and is still under development within ETSI ISG MEC. ETSI and its Members accept no liability for any further use/implementation of this Specification.

Non-published MEC drafts stored in the "Open Area" are working documents, these may be updated, replaced, or removed at any time

Do not use as reference material.

Disclaimer

Do not cite this document other than as "work in progress".

The present document has been produced and approved by the Multi-access Edge Computing (MEC) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Approved and published Specifications and reports for implementation of the MEC system shall be obtained via the ETSI Standards search page at: http://www.etsi.org/standards-search

Reference
RGS/MEC-0030v311V2XAPI

Keywords
API, MEC, service, V2X

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights 7
Foreword 7
Modal verbs terminology 7
1 Scope 8
2 References 8
2.1 Normative references 8
2.2 Informative references 9
3 Definition of terms, symbols and abbreviations 10
3.1 Terms 10
3.2 Symbols 10
3.3 Abbreviations 10
4 Overview 11
5 Description of the service (informative) 12
5.1 Reference scenarios for the VIS 12
5.2 Multi-operator scenarios and V2X services 12
5.3 V2X service continuity in multi-operator operation scenarios 13
5.3.1 Introduction 13
5.4 VIS functionalities 14
5.4.1 Overview 14
5.4.2 Communication between V2X Control Function (3GPP) and VIS (MEC) 15
5.4.3 Inter-MEC system V2X application communication 16
5.4.4 Inter-MEC system service exposure 16
5.4.5 The VIS and its role in producing journey-specific predictive QoS notifications 16
5.4.6 The VIS and its role in multi-domain end-to-end predictive QoS notifications 17
5.4.7 Interaction between the VIS and non-session based V2X services 19
5.4.8 The VIS and the provision of information about V2X Message Distribution Servers 20
5.5 Sequence diagrams 20
5.5.1 Sending a request for provisioning information for V2X communication over Uu unicast 20
5.5.2 Sending a request for provisioning information for V2X communication over Uu MBMS 21
5.5.3 Sending a request for provisioning information for V2X communication over PC5 21
5.5.4 Sending a request for provisioning information for direct communication with V2X Message Distribution Servers 22
5.5.5 Sending a request for journey-specific QoS predictions 22
5.5.6 REST based subscribe-notify model 23
5.5.6.1 Subscribing to event notifications 23
5.5.6.2 Receiving notification on expiry of V2X information event subscription 24
5.5.6.3 Updating subscription for V2X information event notifications 24
5.5.6.4 Unsubscribing from V2X information event notifications 25
5.5.7 Receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu unicast 25
5.5.8 Receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu MBMS 26
5.5.9 Receiving V2X information event notifications about the provisioning information changes for V2X communication over PC5 27
5.5.10 V2X message interoperability 27
5.5.10.1 V2X message subscribe 27
5.5.10.2 V2X message publication 28
5.5.10.3 V2X message notification 28
5.6 Conclusions on VIS 29
6 Data model 29
6.1 Introduction 29
6.2 Resource data types 29
6.2.1 Introduction 29

6.2.2 Type: UuUnicastProvisioningInfo 29
6.2.3 Type: UuMbmsProvisioningInfo 30
6.2.4 Type: Pc5ProvisioningInfo 30
6.2.5 Type: V2xMsgDistributionServerInfo 30
6.2.6 Type: PredictedQos 31
6.2.7 Type: V2xMsgPublication 32
6.3 Subscription data types 33
6.3.1 Introduction 33
6.3.2 Type: ProvChgUuUniSubscription 33
6.3.3 Type: ProvChgUuMbmsSubscription 34
6.3.4 Type: ProvChgPc5Subscription 34
6.3.5 Type: V2xMsgSubscription 35
6.3.6 Type: PredQosSubscription 36
6.3.7 Type: SubscriptionLinkList 36
6.4 Notifications data types 36
6.4.1 Introduction 36
6.4.2 Type: ProvChgUuUniNotification 37
6.4.3 Type: ProvChgUuMbmsNotification 37
6.4.4 Type: ProvChgPc5Notification 37
6.4.5 Type: V2xMsgNotification 38
6.4.6 Type: TestNotification 38
6.5 Referenced structured data types 38
6.5.1 Introduction 38
6.5.2 Type: TimeStamp 38
6.5.3 Type: LocationInfo 38
6.5.4 Type: Plmn 39
6.5.5 Type: Ecgi 39
6.5.6 Type: FddInfo 39
6.5.7 Type: TddInfo 39
6.5.8 Type: V2xApplicationServer 40
6.5.9 Type: UuUniNeighbourCellInfo 40
6.5.10 Type: V2xServerUsd 40
6.5.11 Type: UuMbmsNeighbourCellInfo 40
6.5.12 Type: Pc5NeighbourCellInfo 41
6.5.13 Type: LinkType 41
6.5.14 Type: V2xMsgPropertiesValues 41
6.5.15 Type: V2xMsgFilterCriteria 42
6.5.16 Type: V2xMsgDistributionServer 42
6.5.17 Type: QosPredFilterCriteria 42
6.5.18 Type: WebsockNotifConfig 43
6.6 Referenced simple data types and enumerations 43
6.6.1 Introduction 43
6.6.2 Type: CellId 43
6.6.3 Type: Earfcn 43
6.6.4 Type: TransmissionBandwidth 44

7 API definition 44
7.1 Introduction 44
7.2 Global definitions and resource structure 44
7.3 Resource: uu_unicast_provisioning_info 46
7.3.1 Description 46
7.3.2 Resource definition 46
7.3.3 Resource methods 47
7.3.3.1 GET 47
7.3.3.2 PUT 48
7.3.3.3 PATCH 48
7.3.3.4 POST 48
7.3.3.5 DELETE 48
7.4 Resource: uu_mbms_provisioning_info 49
7.4.1 Description 49
7.4.2 Resource definition 49
7.4.3 Resource methods 49

7.4.3.1 GET 49
7.4.3.2 PUT 50
7.4.3.3 PATCH 50
7.4.3.4 POST 50
7.4.3.5 DELETE 50
7.5 Resource: pc5_provisioning_info 51
7.5.1 Description 51
7.5.2 Resource definition 51
7.5.3 Resource methods 51
7.5.3.1 GET 51
7.5.3.2 PUT 52
7.5.3.3 PATCH 52
7.5.3.4 POST 52
7.5.3.5 DELETE 52
7.6 Resource: provide_v2x_msg_distribution_server_info 53
7.6.1 Description 53
7.6.2 Resource definition 53
7.6.3 Resource methods 53
7.6.3.1 GET 53
7.6.3.2 PUT 53
7.6.3.3 PATCH 53
7.6.3.4 POST 53
7.6.3.5 DELETE 54
7.7 Resource: provide_predicted_qos 54
7.7.1 Description 54
7.7.2 Resource definition 54
7.7.3 Resource methods 55
7.7.3.1 GET 55
7.7.3.2 PUT 55
7.7.3.3 PATCH 55
7.7.3.4 POST 55
7.7.3.5 DELETE 56
7.8 Resource: publish_v2x_message 56
7.8.1 Description 56
7.8.2 Resource definition 56
7.8.3 Resource methods 56
7.8.3.1 GET 56
7.8.3.2 PUT 56
7.8.3.3 PATCH 56
7.8.3.4 POST 56
7.8.3.5 DELETE 57
7.9 Resource: subscriptions 57
7.9.1 Description 57
7.9.2 Resource definition 57
7.9.3 Resource methods 58
7.9.3.1 GET 58
7.9.3.2 PUT 59
7.9.3.3 PATCH 59
7.9.3.4 POST 59
7.9.3.5 DELETE 61
7.10 Resource: existing subscription 61
7.10.1 Description 61
7.10.2 Resource definition 61
7.10.3 Resource methods 61
7.10.3.1 GET 61
7.10.3.2 PUT 62
7.10.3.3 PATCH 64
7.10.3.4 POST 64
7.10.3.5 DELETE 64

Annex A (informative): Mapping of permissions for RESTful API and topic based alternative transport 66

A.1 Overview 66
A.2 Mapping of permissions - RESTful and topic based alternative transport 66

Annex B (informative): Complementary material for API utilization 67

Annex C (informative): Radio access network scenario options for V2X communication 68

Annex D (informative): State-of-the-art of using a Message Broker as V2X Message Distribution Server for exchanging non-session based V2X messages 69

History 70

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) Multi-access Edge Computing (MEC).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document focuses on a MEC V2X Information services (VIS), in order to facilitate V2X interoperability in a multi-vendor, multi-network and multi-access environment, considering the relevant work of other industry bodies relating to V2X communication (e.g. ETSI ITS, 5GAA). It describes the V2X-related information flows, required information and operations. The present document also specifies the necessary API with the data model and data format.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] Void.

[2] Void.

[3] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

NOTE: Available at https://tools.ietf.org/html/rfc5246.

[4] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

NOTE: Available at https://tools.ietf.org/html/rfc6749.

[5] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

NOTE: Available at https://tools.ietf.org/html/rfc6750.

[6] ETSI TS 102 894-2: "Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary".

[7] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

NOTE: Available at https://tools.ietf.org/html/rfc8446.

[8] ETSI TS 133 210: "Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 5G; Network Domain Security (NDS); IP network layer security (3GPP TS 33.210)".

[9] ETSI GS MEC 009: "Multi-access Edge Computing (MEC); General principles, patterns and common aspects of MEC Service APIs".

[10] ETSI TS 123 285: "Universal Mobile Telecommunications System (UMTS); LTE; Architecture enhancements for V2X services (3GPP TS 23.285)".

[11] ETSI TS 136 300: "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall description; Stage 2 (3GPP TS 36.300)".

[12] ETSI TS 136 423: "LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); X2 Application Protocol (X2AP) (3GPP TS 36.423)".

[13] ETSI TS 136 413: "LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP) (3GPP TS 36.413)".

[14] ETSI TS 136 331: "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification (3GPP TS 36.331)".

[15] ETSI TS 136 321: "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification (3GPP TS 36.321)".

[16] Void.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] Void.

[i.2] Void.

[i.3] OpenAPI™ Specification.

NOTE: Available at https://github.com/OAI/OpenAPI-Specification.

[i.4] ETSI GR MEC 022: "Multi-access Edge Computing (MEC); Study on MEC Support for V2X Use Cases".

[i.5] Void.

[i.6] ETSI TS 129 388: "LTE; V2X Control Function to Home Subscriber Server (HSS) aspects (V4); Stage 3 (3GPP TS 29.388)".

[i.7] ETSI TS 129 389: "LTE; Inter-V2X Control Function Signalling aspects (V6); Stage 3 (3GPP TS 29.389)".

[i.8] Void.

[i.9] Void.

[i.10] Void.

[i.11] Void.

[i.12] Void.

[i.13] Void.

[i.14] ETSI GS MEC 003: "Multi-access Edge Computing (MEC); Framework and Reference Architecture".

[i.15] ETSI GS MEC 012: "Multi-access Edge Computing (MEC); Radio Network Information API".

[i.16] ETSI GS MEC 013: "Multi-access Edge Computing (MEC); Location API".

[i.17] ETSI GS MEC 028: "Multi-access Edge Computing (MEC); WLAN Access Information API".

[i.18] https://autopilot-project.eu/.

[i.19] MQTT: The Standard for IoT Messaging.

NOTE: Available at https://mqtt.org/.

[i.20] https://5gcroco.eu/.

[i.21] https://5gcarmen.eu/.

[i.22] Advanced Message Queuing Protocol - https://www.amqp.org/.

[i.23] https://5gmeta-project.eu/.

[i.24] C-ROADS - THE PLATFORM OF HARMONISED C-ITS DEPLOYMENT IN EUROPE.

NOTE: Available at https://www.c-roads.eu/platform.html.

[i.25] https://www.ict4cart.eu/.

[i.26] https://developer.here.com/documentation/traffic/dev_guide/topics/quick-start.html.

[i.27] ETSI TS 123 288: "Architecture enhancements for 5G System (5GS) to support network data analytics services".

[i.28] ETSI TS 123 502: "Procedures for the 5G System (5GS)".

[i.29] Void.

[i.30] 5GAA TR-200055: 5GS Enhancements for Providing Predictive QoS in C-V2X.

NOTE: Available at https://5gaa.org/wp-content/uploads/2020/05/5GAA_A-200055_eNESQO_TR_final.pdf.

[i.31] 5GAA White Paper, Making 5G Proactive and Predictive for the Automotive Industry.

NOTE: Available at https://5gaa.org/wp-content/uploads/2020/01/5GAA_White-Paper_Proactive-and-Predictive_v04_8-Jan.-2020-003.pdf.

[i.32] ETSI GR MEC 001: "Multi-access Edge Computing (MEC); Terminology".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GR MEC 001 [i.32] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in ETSI GR MEC 001 [i.32] and the following apply:

5GAA	5G Automotive Association
5QI	5G QoS Identifier
AMQP	Advanced Message Queuing Protocol
BTP	Basic Transport Protocol
CAM	Cooperative Awareness Message
CDF	Cumulative Distribution Function
C-ITS	Cooperative Intelligent Transport Systems

CN	Core Network
CQI	Channel Quality Indicator
C-V2X	Cellular V2X
DENM	Decentralized Environmental Notification Message
DL	Downlink
DN	Data Network
E2E	End-to-End
eNB	evolved Node B
E-UTRAN	Evolved UMTS Terrestrial Radio Access Network
FDD	Frequency Division Duplex
FOTA	Firmware Over The Air
gNB	5G Node B
HSS	Home Subscriber Server
IoT	Internet of Things
ITS	Intelligent Transport Systems
KPI	Key Performance Indicator
MaaS	Mobility-as-a-Service
MBMS	Multimedia Broadcast Multicast Services
MQTT	Message Queuing Telemetry Transport
NEF	Network Exposure Function
NF	Network Function
NWDAF	Network Data Analytics Function
OEM	Original Equipment Manufacturer
PF	Prediction Function
PoP	Point-of-Presence
RSU	Road Side Unit
SLA	Service-Level Agreement
ToD	Teleoperated Driving
SDP	Session Description Protocol
SOTA	Software Over The Air
TC	Technical Committee
TDD	Time Division Duplex
UDP	User Datagram Protocol
UL	Uplink
V2N2I	Vehicle-to-Network-to-Infrastructure
V2N2V	Vehicle-to-Network-to-Vehicle
VIS	V2X Information Services

4 Overview

The present document specifies the VIS API to facilitate V2X interoperability in a multi-vendor, multi-network and multi-access environment.

Clause 5 presents reference scenarios for the VIS service and lists the functionalities of the service. It also describes the information flows used for VIS.

The information that can be exchanged over the VIS API is described in clause 6 which provides detailed descriptions of all information elements that are used for VIS.

Clause 7 describes the actual VIS API providing detailed information of how information elements are mapped into a RESTful API design.

5 Description of the service (informative)

5.1 Reference scenarios for the VIS service

According to recommendations in ETSI GR MEC 022 [i.4], multi-access, multi-network and multi-operator scenarios are the reference assumptions motivating the need for MEC normative work on this area. Figure 5.1-1 shows all the scenarios applicable to V2X services. In particular:

- Some V2X services can be managed by OEMs (the so called "Vehicle OEMs scenario"), and, thus, it is reasonable to consider both single and multi-operator scenarios for such services. Note that V2X services are expected to be provided by different network operators in the same country and/or in different countries.
- Similarly, the same applies when the "ITS Operator scenario" is considered, that may additionally provide services for different vehicle OEMs. An ITS operator may need to provide a country-wide V2X service, by exploiting different operators' networks (deploying different MEC systems), and offering this service to vehicles belonging to different OEMs. Note that also in this case, V2X services are expected to be provided by different network operators in the same country and/or in different countries.

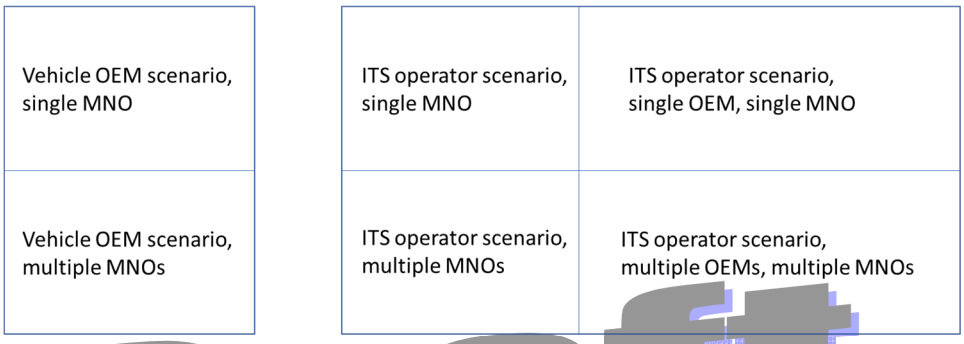


Figure 5.1-1: Reference scenarios relevant to the VIS service

As a consequence, in order to enable all use cases, the MEC V2X Information services (VIS) should support C-V2X systems implemented in the most general scenarios. In particular, these scenarios should assume the presence of multiple MEC vendors and the need to enable interoperable data exchange between them. Moreover, multi-operator interoperability is a key aspect for ensuring service continuity, and it is described in clause 5.2.

5.2 Multi-operator scenarios and V2X services

The left hand side of figure 5.2-1 shows a typical multi-operator scenario, highlighting the case of temporary absence of radio coverage, e.g. in roaming situations. As showed in the right-hand side of figure 5.2-1, in a traditional V2X system (without the VIS service) the interconnection between MNOs is terminated at the remote side, with clear disadvantages in terms of high E2E latency; on the other hand, thanks to the exploitation of the VIS service (enabling also a "horizontal communication" between MEC systems), the interconnection between MNOs can be realized with low E2E latency.

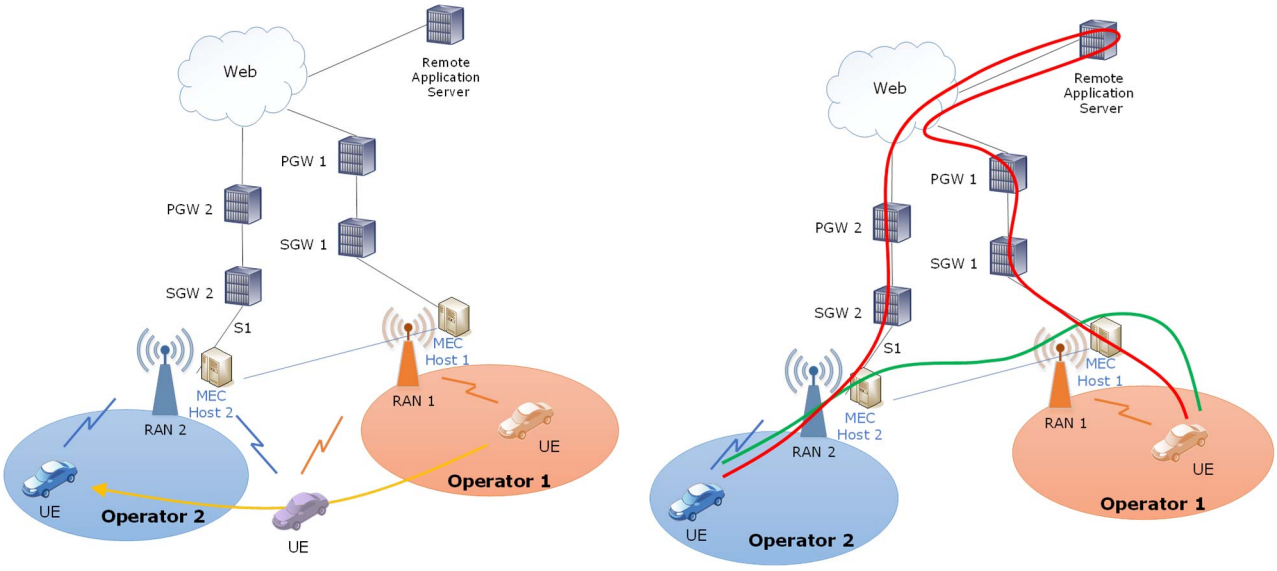


Figure 5.2-1: (left): Example of a multi-operator scenario for V2X services; (right): Example of path for data exchange without the VIS service (in red) and with the VIS service (in green)

V2X service needs to be provided across all the territory including both operators' coverage areas, as well as when leaving the coverage area of one operator and entering the coverage area of the other operator without any service disruption and guaranteeing E2E performance. For that purpose, VIS exposes information on PC5 configuration parameters and manages the multi-operator environment, especially when a UE is out of coverage.

5.3 V2X service continuity in multi-operator operation scenarios

5.3.1 Introduction

Wireless communication is a key enabling technology of co-operative intelligent transportation systems. Road users (including vehicles, cyclists, pedestrians) involved in the communication may use services provided by different operators.

A mobile operator network is typically region specific or country specific, which provides services directly to its own customers (subscribers), while providing communications to other operators' customers via the core network level interworking between two operators' networks. To maintain the V2X service continuity (often with low latency requirement) for road users becomes very challenging especially when such road users (e.g. vehicular UEs) move from one PLMN to another.

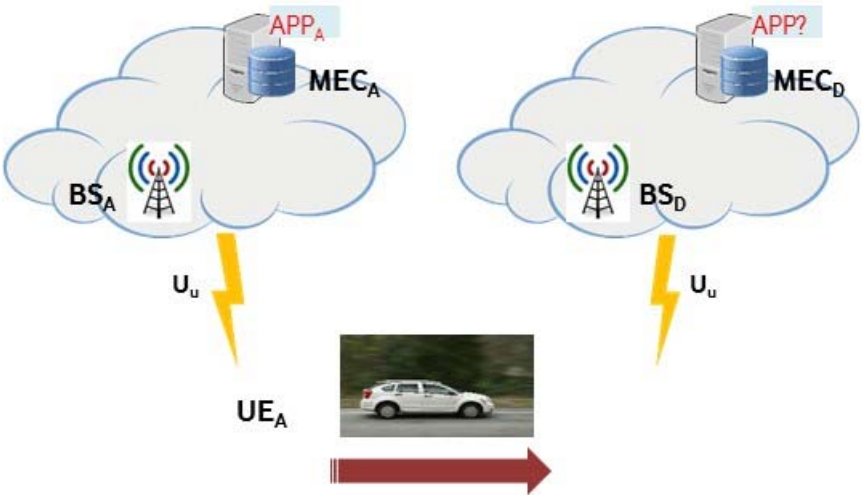


Figure 5.3.1-1: Example V2X use case: inter-PLMN service continuity

To enable service continuity in such use cases, mobile network level interworking among different PLMNs is necessary as specified in 3GPP specifications ETSI TS 123 285 [10], ETSI TS 129 388 [i.6] and ETSI TS 129 389 [i.7]. Furthermore, inter-MEC system coordination is also required to prepare in advance the UEs in transit (based on the agreements among operators, roam or handover to a new PLMN) and reduce the interruption time.

The service consumers communicate with VIS over the VIS API to get the necessary V2X service provisioning information for the visiting PLMN in support of inter-PLMN service continuity. Both the MEC applications and the MEC platform may consume the VIS; and both the MEC platform and the MEC applications may be the providers of the V2X information.

The VIS API supports both queries and subscriptions (pub/sub mechanism) that are used over the RESTful API or over alternative transports such as message bus. Alternative transports are not specified in detail in the present document. For RESTful architectural style, the present document defines the HTTP protocol bindings.

5.4 VIS functionalities

5.4.1 Overview

The MEC standards have been designed to facilitate V2X interoperability in a multi-vendor, multi-network and multi-access environment. The introduction of the VIS API is aimed at helping the ecosystem adopt MEC for automotive use cases. These use cases may involve different car makers, OEM suppliers, network infrastructure vendors, MEC vendors, application/content providers and other stakeholders. Therefore, it is critical that all MEC related interoperability reference points involving the potential stakeholders are fully specified.

In particular, the VIS defined in the present document will permit information exposure, pertinent to the support of automotive use cases, to MEC application instances. It will also permit a single ITS operator to offer a V2X service over a region that may span different countries and involve multiple network operators, MEC systems and MEC application providers.

For that purpose, the MEC VIS includes the following functionalities:

1) Gathering of PC5 V2X relevant information from the 3GPP network (e.g. the list of authorized UEs, the relevant information about the authorization based on the UE subscription and the relevant PC5 configuration parameters).

2) Exposure of this information to MEC apps (also potentially belonging to different MEC systems).

3) Enablement of MEC apps to communicate securely with the V2X-related 3GPP core network logical functions (e.g. V2X control function).

4) Enablement of MEC apps in different MEC systems to communicate securely with each other.

5) Possibly gathering and processing information available in other MEC APIs (e.g. RNI API, see ETSI GS MEC 012 [i.15], Location API, see ETSI GS MEC 013 [i.16], WLAN API, see ETSI GS MEC 028 [i.17], etc.) in order to predict radio network congestion and provide suitable notifications to the UE.

From that perspective, the VIS service is relevant to Mp1 and Mp3 reference points in the MEC architecture. In particular, the relevant information is exposed to MEC apps via the Mp1 reference point. Potential impacts on Mp3 reference point (e.g. enabling the possibility to transfer this information between different MEC platforms) are introduced in ETSI GS MEC 003 [i.14] and are out of the scope of the present document.

NOTE 1: The VIS API provides information to MEC applications in a standardized way; this is essential for interoperability in multi-vendor scenarios; nevertheless, it is acknowledged that MEC applications may communicate in a direct way (i.e. without the use of MEC platform).

NOTE 2: Inter-system communication may be realized between MEOs. As an alternative, or, in addition to that, possible Mp3 enhancements (or new reference points between MEC systems) may be defined. This is out of the scope of the present document.

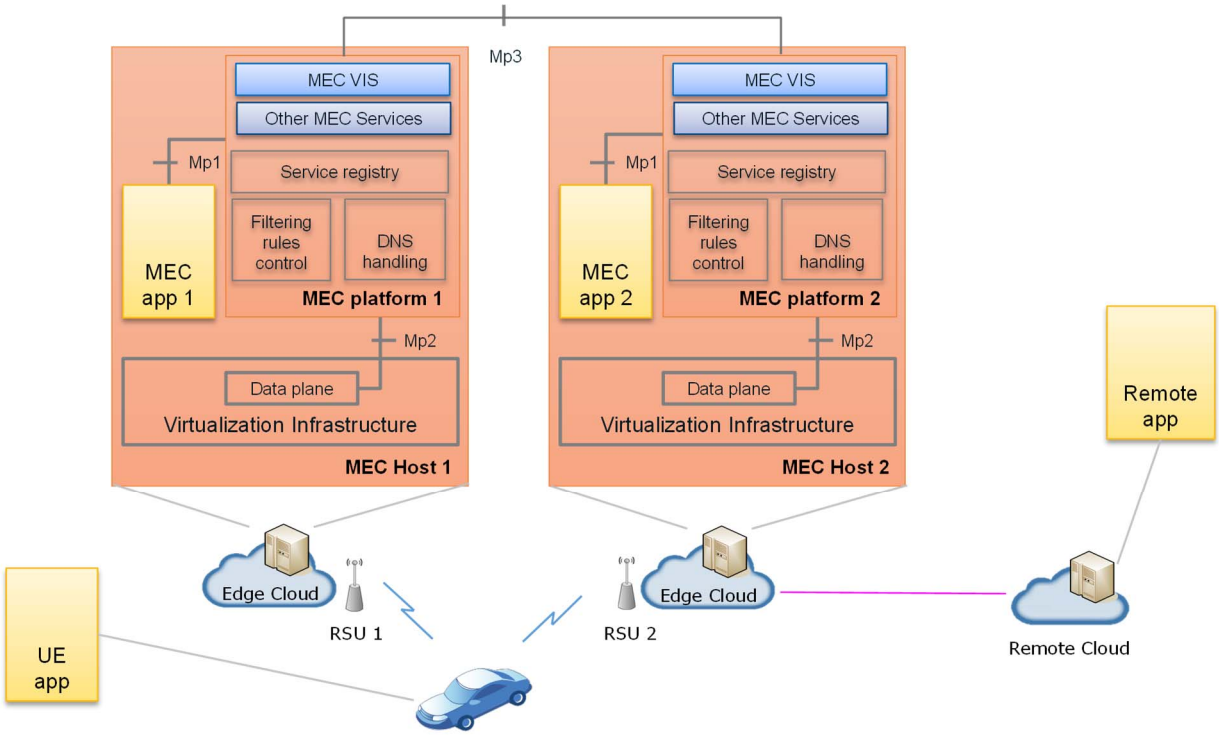


Figure 5.4.1-1: Example of application instances in a V2X service with VIS API

Figure 5.4.1-1 illustrates a typical V2X system involving multiple MEC hosts and the use of the VIS service. In the framework of V2X services, a car is hosting a client application, and is connected to a certain MEC host (and a related MEC application). In presence of multiple MEC hosts, the VIS permits to expose information between MEC applications running on different MEC hosts. In addition, other remote application server instances can be located somewhere else (e.g. private clouds owned by the operator or by the OEM). The VIS service may be produced by the MEC platform or by the MEC application.

5.4.2 Communication between V2X Control Function (3GPP) and VIS (MEC)

In a 3GPP network, V2X applications can be deployed on V2X Application Server. The V2X Control Function is the NF in core network part, which is used for network-related actions required for V2X. The HSS provides the list of the PLMNs, where the UE is authorized to perform V2X communication over PC5 reference point to the V2X Control Function, see ETSI TS 123 285 [10]. V2 is the reference point between the V2X Application Server and the V2X Control Function in the operator's network.

The VIS defined in MEC is used to facilitate V2X interoperability in a multi-vendor, multi-network and multi-access environment. Therefore, the VIS should obtain the UE's subscription data (e.g. PC5 based V2X communication allowed PLMN), from the V2X Control Function.

Because the V2X Application Server bears multiple V2X applications, it can, therefore, be deployed in MEC platform as a MEC application. The VIS can communicate with the V2X Application Server through Mp1, and it can obtain the UE's V2X subscription data from the V2X Control Function through the V2X Application Server.

NOTE: The VIS, or generic parts of it, can be deployed in the MEC Platform.

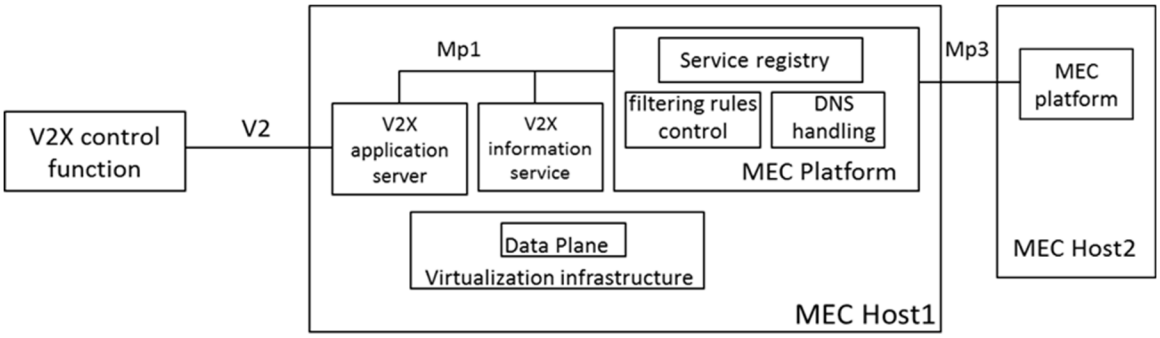


Figure 5.4.2-1: Example of architecture enabling the communication between the VIS and the V2X Control Function

5.4.3 Inter-MEC system V2X application communication

A V2X MEC application may be required to communicate with its peer applications in other MEC systems in order to fulfil the intended purpose of the application use case. The involved MEC systems need to enable the authorized applications in one MEC system to communicate with their peers in another MEC system.

The discovery of the application peers may be facilitated by the VIS API by exposing the available communication end point information for peer to peer connectivity. Alternatively, the configured traffic rules for the V2X MEC application together with the underlying inter-MEC system connectivity arrangements may support the application peers' communication. Lastly, the V2X MEC application may rely on non-MEC-specific means for its peer discovery and then rely on its authorized access to external interface for the communication.

The required arrangements between the involved MEC systems for realizing secure connectivity with the application specific requirements are deployment specific and beyond the scope of the present document.

5.4.4 Inter-MEC system service exposure

A V2X MEC application in one MEC system may be required to consume a service in another MEC system in order to fulfil the intended purpose of the application use case. The V2X MEC application discovers the service in question in the service registry in its local MEC host.

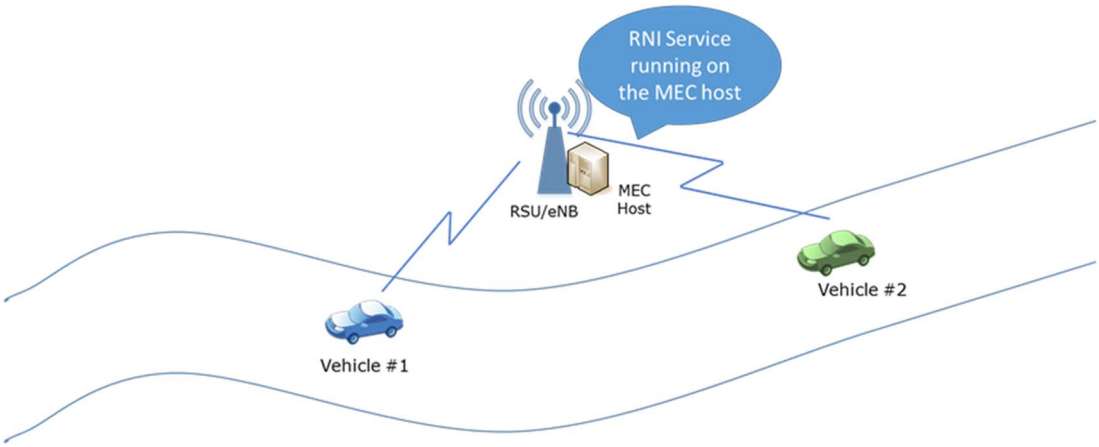
The required arrangements between the involved MEC systems for mapping a service produced in one MEC system to an endpoint in another MEC system are deployment specific and beyond the scope of the present document.

5.4.5 The VIS and its role in producing journey-specific predictive QoS notifications

Accurate and timely predictions of the radio environment at locations planned to be visited by vehicles can either trigger, modify or postpone:

i) the application of certain V2X functionalities; and/or

ii) the download of content delivery/ software packages.



NOTE: Figure 5.4.5-1 illustrates an exemplary V2X system scenario, where the MEC host is deployed in collocation with a Roadside Unit (RSU)/ eNB providing coverage (V2X communication); the RNI Service (RNIS) is running at the MEC host - it is assumed that the planned trajectory of vehicle #1 and vehicle #2 is not known at the RSU.

Figure 5.4.5-1: Exemplary V2X system scenario (see note)

However, focusing on V2X system scenarios characterized by high mobility and dynamic topology (as in figure 5.4.5-1), the accuracy and the timeliness of information (e.g. radio network, location information, etc.) may be hampered by:

- the environmental situation, e.g. the occurrence of network congestion events when, for example, many vehicles attempt to provide radio measurements to the connected eNB/gNB, which is collocated with a MEC host, as well as by
- the deployment density of the cellular network, together with the capabilities of the deployed MEC infrastructure.

An example illustrating the impact of the above mentioned limitations on system performance is the one of a vehicle planning to follow a trajectory from location A to location B and a related MEC application which would need to be informed of radio conditions "en route", ahead of the vehicle's passing time, before reaching a decision. Decisions may consist in e.g. enabling/disabling autonomous driving features, downloading infotainment content, scheduling Software/Firmware Over-the-Air (SOTA/FOTA) updates, etc.

To address such challenges, the VIS service may assist in implementing a framework for cooperative acquisition, partitioning and distribution of information for efficient, journey-specific QoS prediction. That is, the VIS service may be utilized to identify space/time correlations between radio quality data collected by different vehicles in a V2X system and a specific vehicle's planned journey for better prediction of the quality of the communication network along the designated route. As a consequence, the VIS may expose relevant (i.e. journey-specific) information about the QoS prediction to authorized UEs.

In addition, QoS predictions might be useful to V2X applications or service providers, e.g. a Mobility-as-a-Service (MaaS) provider, a fleet management company, a vehicle OEM, a MNO, or a third party at the infrastructure side. For example, the predictive QoS notifications can be exploited by a fleet management company, or an OEM as information useful to decide upon the means to reduce the incurred service consumption cost. This is because in some situations, radio or network bottlenecks may appear, including MEC Points-of-Presence (PoP) computing resource availability, if many vehicles converge to a certain geographical area, or because some specific road traffic event happened. In such cases, MEC PoP optimization might be required (e.g. relocating the VM of a V2X MEC application instance to another MEC host in the MEC system, or performing a VM scale-up of a V2X MEC application instance in the same MEC host). Furthermore, such QoS predictions could be a useful input in workload offloading decisions from a vehicle to a MEC host.

5.4.6 The VIS and its role in multi-domain end-to-end predictive QoS notifications

In addition to the data collected by the different vehicles, VIS may also enable support for predictive QoS notifications in multi-domain MEC deployments by collecting and distributing analytics from external Prediction Functions (PFs) located in the different network domains. The sourcing of analytics from external PFs, such as the Network Data Analytics Function (NWDAF) in the 3GPP network domains may help in providing more accurate QoS predictions to the service consumers.

In 5G Automotive Association (5GAA) activities, multiple scenarios are studied, where the service consumer may benefit from the information related to the QoS prediction of an E2E user plane link between two specific endpoints. These scenarios are ranging from Teleoperated Driving (ToD), towards coordinated maneuvers scenarios and use cases, such as lane merge, platooning, etc. as well as high definition map collection/sharing and in-vehicle infotainment, as described in [i.30] and [i.31]. In this context, a service consumer may be a MEC application, a MEC platform, a V2X application server or an "ultimate" QoS prediction consumer, such as a vehicle or an OEM application accessing the VIS via a MEC application. According to the different scenarios studied by 5GAA (V2N2V and V2N2I [i.31]), these two endpoints may be application instances deployed in two vehicles and exchanging V2X messages or an application instance in a vehicle exchanging V2X messages with the road infrastructure, in both cases with the support of the MEC infrastructure. Two exemplary scenarios are described in figures 5.4.6-1 and 5.4.6-2, where the QoS prediction of the E2E user plane link should consider up to 5 different network domains. In figure 5.4.6-1 (V2N2V), those domains are specifically: the 3GPP network domains of the two operators (MNO A an MNO B), the Data Networks (DN A) and (DN B) where the MEC host is located, as well as the IP controlled network that is interconnecting the networks of the PLMN A and PLMN B.

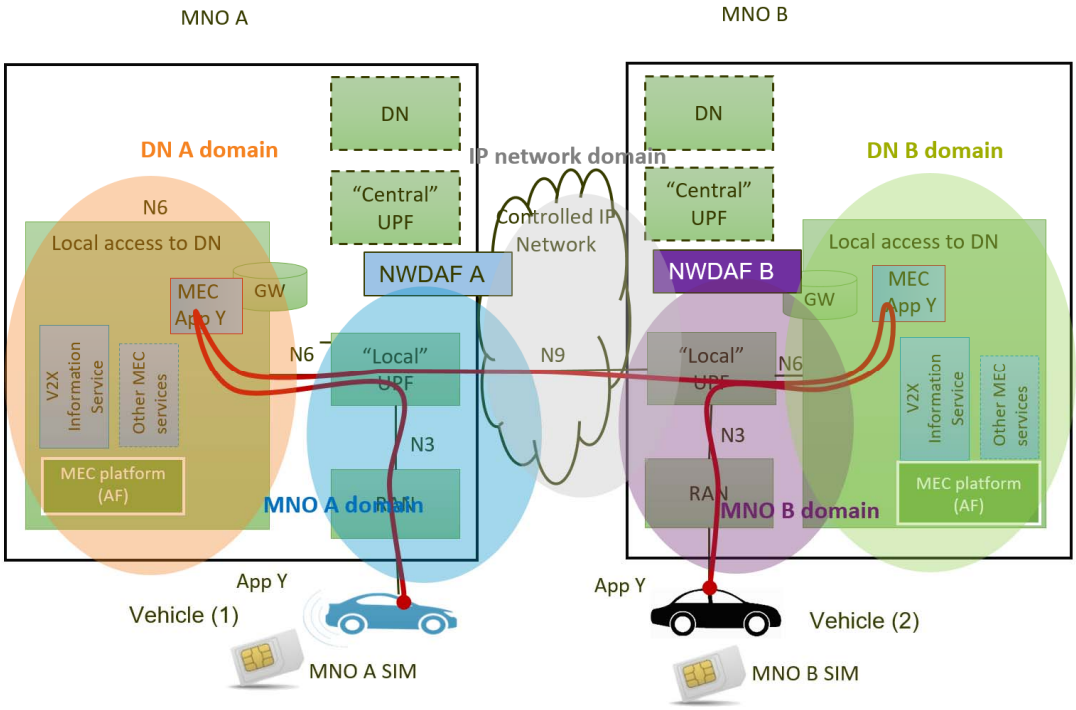


Figure 5.4.6-1: E2E scenario for multi-operator V2X deployment (case V2N2V)

In figure 5.4.6-2 (case V2N2I), those network domains for E2E user plane link are instead 4 distinct domains: the 3GPP network domain of the operator that provides 5G connectivity to the vehicle (MNO A for vehicle 1), the DN (DN A for vehicle 1) where the MEC host is located, the interconnection domain and the 3rd party domain where the road infrastructure is located.

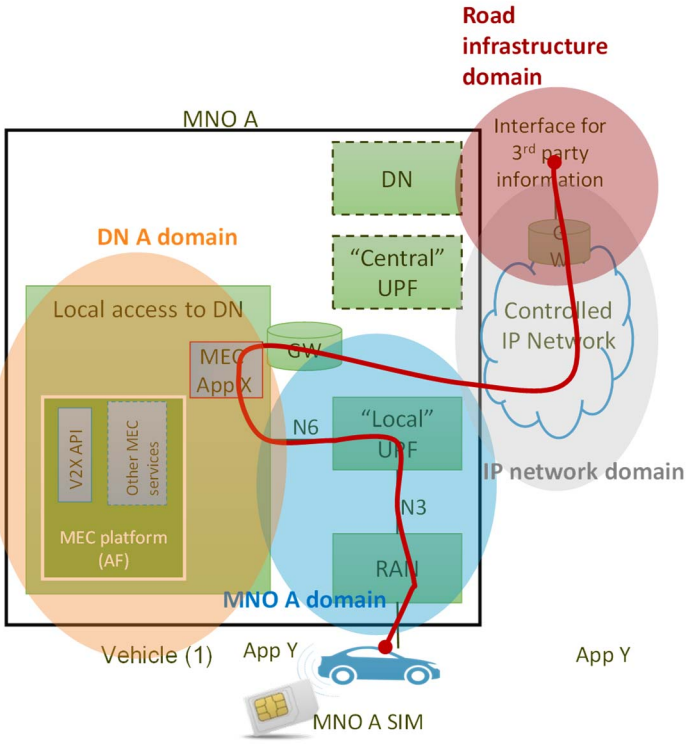


Figure 5.4.6-2: E2E scenario for multi-operator V2X deployment (case V2N2I)

In the case of 3GPP System domains, the VIS may cooperate with the analytics service provided by the NWDAF according to clause 6.9 of ETSI TS 123 288 [i.27]. Such analytics service can provide predictions for the 3GPP network domain (e.g. at the Radio Access Network - RAN level and Core Network segment between the UE and the User Plane Function - UPF node), such as the QoS Sustainability Analytics.

Other network domains, such as the IP interconnect domain or DNs in the Service Provider domain (including the MEC host) may also deploy a domain-specific PF that can provide analytics information to a calling service consumer via the VIS. These DN domain specific predictions may be provided by NWDAF, as described by the DN Performance Analytics described in clause 6.14 of ETSI TS 123 288 [i.27].

The 3GPP system domain specific predictions may include analytics for user plane performance (i.e. average/maximum traffic rate, average/maximum packet delay, average packet loss rate) in the form of statistics or predictions to a service consumer. VIS may aggregate such information to provide an E2E QoS prediction in the considered multi-domain scenario.

These user plane performance analytics may be obtained from the NWDAF directly or via the Network Exposure Function (NEF) according to clause 5.2.6.16 of ETSI TS 123 502 [i.28].

NOTE: In some cases, a V2X MEC application communicating with the VIS may also conveniently generate a prediction for one or more domains where the PF is not available based on collected information.

VIS may aggregate analytics sourced from external PFs, such as the NWDAF or other analytics provider in order to provide (predictions/estimations) of the QoS that cover the E2E user plane link between two V2X application instances.

It is out of scope how VIS can collect such information from the external PFs.

5.4.7 Interaction between the VIS and non-session based V2X services

Several V2X services, that are currently implemented, are non-session based services. These services concern the broadcasting of information related to events or warnings about potential danger (e.g. Road Status Notification, Warning Alert). This means that a single V2X message needs to be delivered to a set of locally relevant UEs or to a set of locally relevant V2X Application Servers.

As introduced in the Annex D of the 3GPP specification ETSI TS 123 285 [10], a possible deployment approach is based on the use of a V2X Message Distribution Server that is in charge to receive V2X messages from non-session based V2X services and to distribute them to the relevant destination entities.

This approach is widely adopted in several research projects and other initiatives for the implementation of the communication over the Uu interface for non-session based V2X services. In the specific, a Message Broker is typically deployed as V2X Message Distribution Server. Different application layer protocols, on which a Message Broker is based, can be used for the scope. Details about the existing implementations are provided in Annex D of the present document.

The VIS should implement the required interactions with the Message Brokers, that are acting as V2X Message Distribution Servers, to correctly provide to the MEC applications the interaction with the non-session based V2X services (i.e. transmit and receive V2X messages related to non-session based V2X services). The definition of which type of Message Broker the VIS should interact with and how this interaction should be performed are beyond the scope of the present document.

5.4.8 The VIS and the provision of information about V2X Message Distribution Servers

A V2X MEC application implementing a non-session based V2X service, may require very low latency in the publication or notification of V2X messages. This strict latency requirement can be due to the specificities of the implemented V2X service that can be related to safety critical aspects.

The MEC application may need to directly interact with V2X Message Distribution Servers used for exchanging non-session based V2X messages to avoid any additional latency that is introduced by the VIS. The MEC application should, therefore, be provided with provisioning information about which V2X Message Distribution Servers are available, their characteristics, and how it can connect with them.

This information can be provided by the VIS upon the registration of the V2X Message Distribution Servers to the MEC platform, e.g. when these V2X Message Distribution servers are implemented as service-producing MEC applications. Each V2X Message Distribution Server can be based on a specific application layer protocol (e.g. MQTT, AMQP) and manage the exchange of V2X messages of a given geographical area. The MEC application can provide some criteria to select only the relevant V2X Message Distribution Servers based on application layer protocols that the MEC application supports or geographical areas of interest to the MEC application.

The way that the V2X Message Distribution Servers are registered to a MEC platform (e.g. as service-producing MEC applications) and the interactions among the VIS and the V2X Message Distribution Servers are beyond the scope of the present document.

5.5 Sequence diagrams

5.5.1 Sending a request for provisioning information for V2X communication over Uu unicast

Figure 5.5.1-1 shows a scenario where the service consumer (e.g. a MEC application or a MEC platform) sends a request to receive the provisioning information for V2X communication over Uu unicast for a particular location. The response contains the required information.

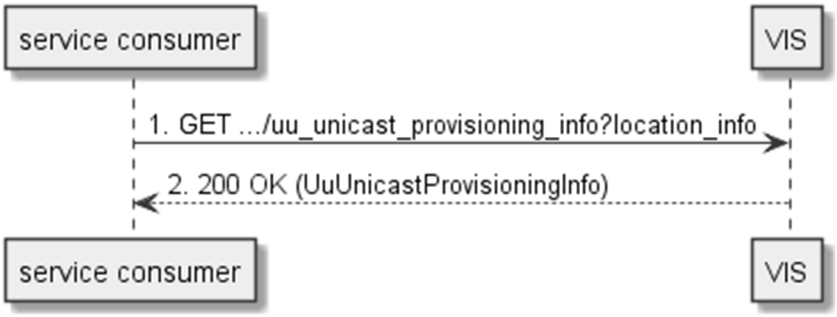


Figure 5.5.1-1: Flow of service consumer requesting the Uu unicast provisioning information

A service consumer requesting the Uu unicast provisioning information, as illustrated in figure 5.5.1-1, consists of the following steps:

1) Service consumer sends a GET request to the resource representing the Uu unicast provisioning information. The request contains the location information (e.g. the serving cell ID of or the geographical area information of the UE) as an input parameter.

2) VIS responds with "200 OK" with the message body containing the UuUnicastProvisioningInfo.

5.5.2 Sending a request for provisioning information for V2X communication over Uu MBMS

Figure 5.5.2-1 shows a scenario where the service consumer (e.g. a MEC application or a MEC platform) sends a request to receive the provisioning information for V2X communication over Uu MBMS for a particular location. The response contains the required information.

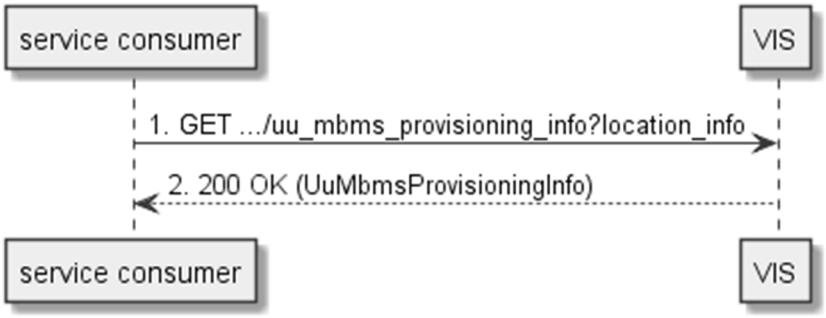


Figure 5.5.2-1: Flow of service consumer requesting the Uu MBMS provisioning information

A service consumer requesting the Uu MBMS provisioning information, as illustrated in figure 5.5.2-1, consists of the following steps:

1) Service consumer sends a GET request to the resource representing the Uu MBMS provisioning information. The request contains the location information (e.g. the serving cell ID of or the geographical area information of the UE) as an input parameter.

2) VIS responds with "200 OK" with the message body containing the UuMbmsProvisioningInfo.

5.5.3 Sending a request for provisioning information for V2X communication over PC5

Figure 5.5.3-1 shows a scenario where the service consumer (e.g. a MEC application or a MEC platform) sends a request to receive the provisioning information for V2X communication over PC5 for a particular location. The response contains the required information.

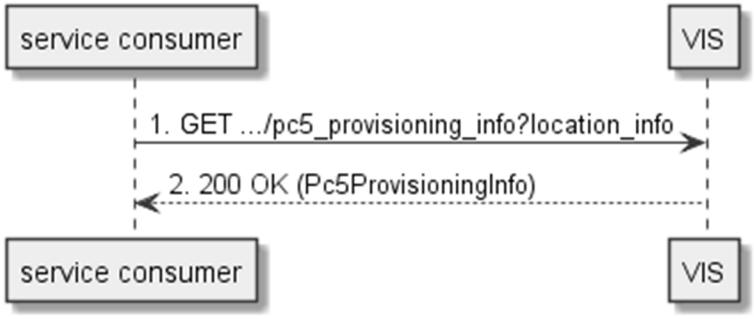


Figure 5.5.3-1: Flow of service consumer requesting the PC5 provisioning information

A service consumer requesting the PC5 provisioning information, as illustrated in figure 5.5.3-1, consists of the following steps:

1) Service consumer sends a GET request to the resource representing the PC5 provisioning information. The request contains the location information (e.g. the serving cell ID of or the geographical area information of the UE) as an input parameter.

2) VIS responds with "200 OK" with the message body containing the Pc5ProvisioningInfo.

5.5.4 Sending a request for provisioning information for direct communication with V2X Message Distribution Servers

Figure 5.5.4-1 shows a scenario where, the service consumer (e.g. a V2X application) sends a POST request to VIS to receive information about available V2X Message Distribution Servers. The response contains the required information.

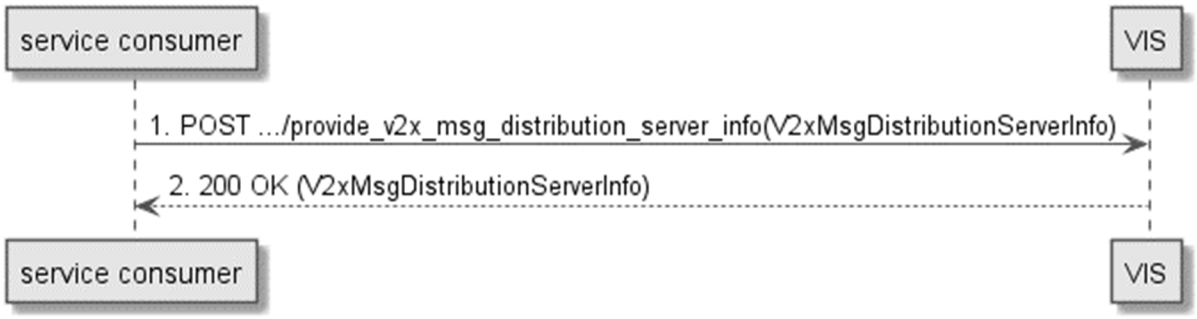


Figure 5.5.4-1: Flow of a V2X application requesting information about available V2X Message Distribution Servers

The procedure of a V2X application requesting information about available V2X Message Distribution Servers, as illustrated in figure 5.5.4-1, consists of the following steps:

1) The service consumer sends a POST request to VIS. The message body contains the data structure V2xMsgDistributionServerInfo containing the information of the V2X Message Distribution Servers that the V2X application uses as criteria for direct interaction with relevant V2X Message Distribution Servers.

The VIS returns the "200 OK" response to the service consumer with the message body containing the V2xMsgDistributionServerInfo data structure.

5.5.5 Sending a request for journey-specific QoS predictions

Figure 5.5.5-1 shows a scenario where, the service consumer (e.g. a V2X application) sends a POST request to VIS to receive the predicted QoS correspondent to potential routes of a vehicular UE. The response contains the required information.

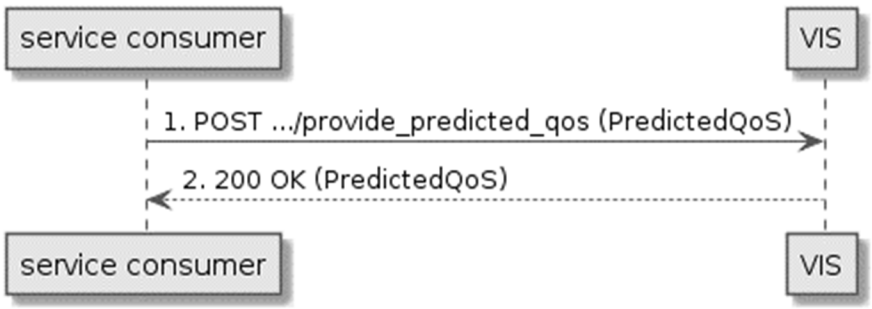


Figure 5.5.5-1: Flow of a V2X application requesting the predicted QoS of a UE with potential routes

The procedure of a V2X application requesting the predicted QoS for a vehicular UE with potential routes, as illustrated in figure 5.5.5-1, consists of the following steps:

1) The service consumer sends a POST request to VIS. The message body contains the data structure for the predicted QoS relevant to potential routes of the vehicular UE.

2) The VIS returns the "200 OK" response to the service consumer with the message body containing the predicted QoS data structure.

In addition to the request/response type of interaction for QoS predictions, VIS can use subscription/notification interactions with the service consumers that ensures more flexibility. In this context, the service consumer may subscribe to the journey-specific QoS prediction when the vehicle starts to drive towards a destination and VIS may continuously provide in-advance predictions on relevant events along the route, as such information becomes available.

Also in the case of multi-domain E2E predictive QoS notifications, VIS can use subscription/notification interactions with the service consumers that ensures more flexibility. In this context, the service consumer may subscribe to the multi-domain E2E predictive QoS notification when the vehicle enters a specific road traffic scenario (e.g. approaching an intersection, starting a lane merge operation, etc.) and VIS may continuously provide in-advance predictions on relevant events along the route, as such information becomes available.

5.5.6 REST based subscribe-notify model

5.5.6.1 Subscribing to event notifications

To receive notifications on selected V2X information events, the service consumer creates a subscription to certain specific V2X information event that is available at VIS. Figure 5.5.6.1-1 shows a scenario where the service consumer uses REST based procedures to create a subscription for V2X information event notifications.

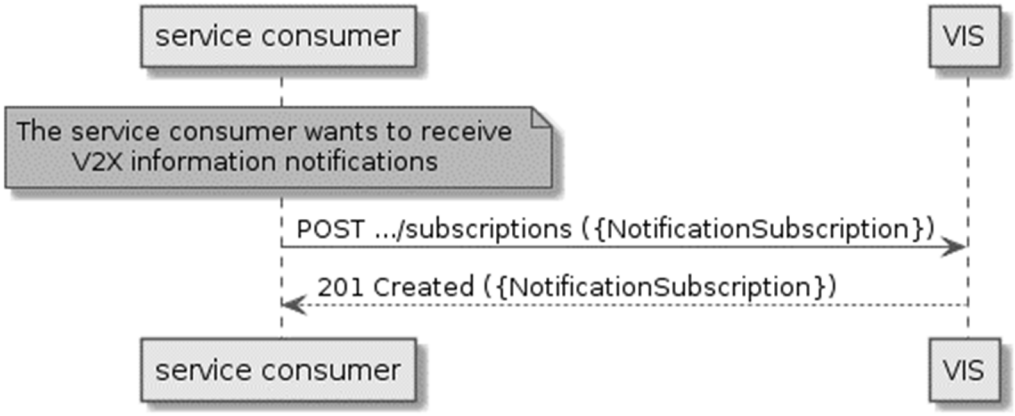


Figure 5.5.6.1-1: Flow of subscribing to the V2X information event notifications

Subscribing to the V2X information event notifications, as illustrated in figure 5.5.6.1-1, consists of the following steps.

When the service consumer wants to receive notifications about the V2X information events, it creates a subscription to the V2X information event notifications:

1) The service consumer sends a POST request with the message body containing the {NotificationSubscription} data structure to the resource representing V2X information subscription. The variable {NotificationSubscription} is replaced with the data type specified for different V2X information event subscriptions, and it defines the subscribed event, the filtering criteria and the address where the service consumer wishes to receive the V2X information event notifications.

2) VIS sends "201 Created" response with the message body containing the data structure specific to that V2X information event subscription. The data structure contains the address of the resource created and the subscribed V2X information event type. The address of the resource created is also contained in the message header.

5.5.6.2 Receiving notification on expiry of V2X information event subscription

VIS may define an expiry time for the V2X information event subscription. In case expiry time is used, the time will be included in the {NotificationSubscription} data structure that is included in the response message to the subscription. Prior to the expiry, VIS will also send a notification to the service consumer that owns the subscription.

Figure 5.5.6.2-1 shows a scenario, where the service consumer receives a subscription expiry notification for the existing subscription.

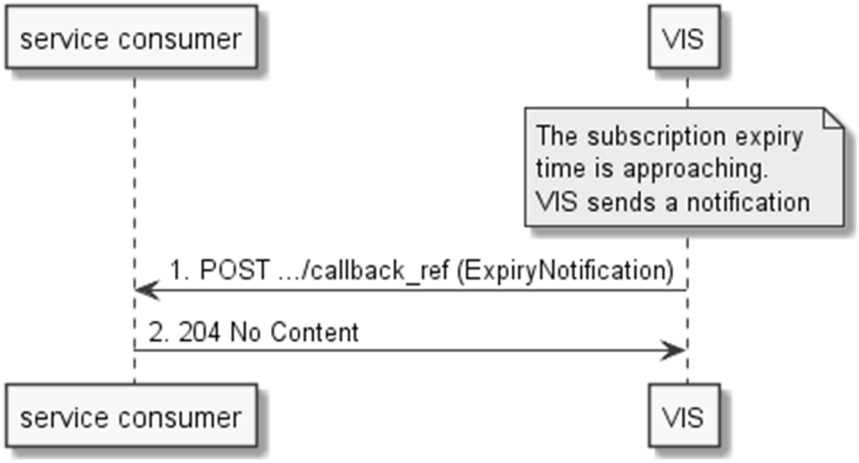


Figure 5.5.6.2-1: Flow of VIS sending a notification on expiry of the subscription

Sending a notification on expiry of the subscription, as illustrated in figure 5.5.6.2-1 consists of the following steps. If VIS has defined an expiry time for the subscription, VIS will send a notification prior the expiry:

1) VIS sends a POST request to the callback reference address included by the service consumer in the subscription request. The POST request contains a data structure ExpiryNotification.

2) Service consumer sends a "204 No Content" response to the VIS.

5.5.6.3 Updating subscription for V2X information event notifications

Figure 5.5.6.3-1 shows a scenario where the service consumer needs to update an existing subscription for a V2X information event notification. The subscription update is triggered e.g. by the need to change the existing subscription, or due to the expiry of the subscription.

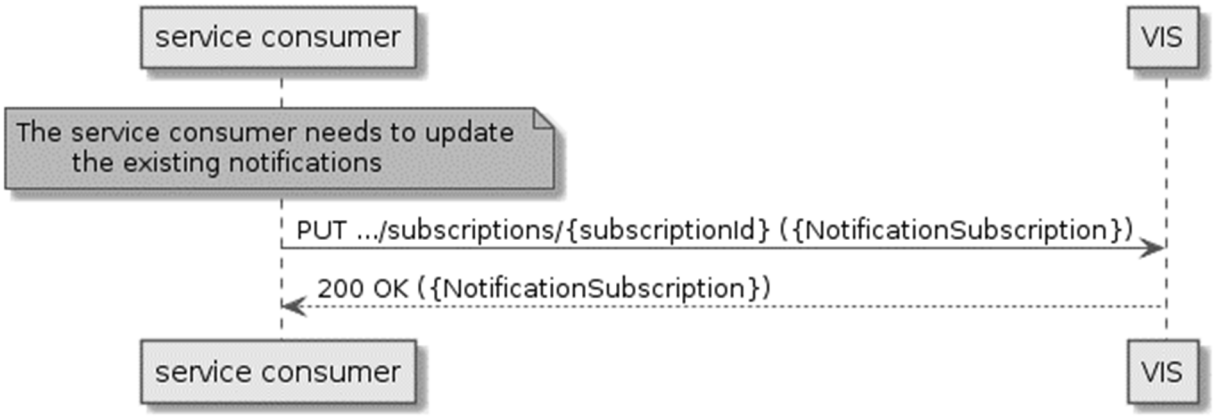


Figure 5.5.6.3-1: Flow of service consumer updating subscription for V2X information event notifications

Updating subscription for V2X information event notifications, as illustrated in figure 5.5.6.3-1, consists of the following steps.

When the service consumer needs to modify an existing subscription for V2X information event notifications, it can update the corresponding subscription as follows:

1) Service consumer updates the subscription resource by sending a PUT request to the resource representing the V2X information event subscription that was created with the modified data structure specific to that V2X information event subscription.

2) VIS returns "200 OK" with the message body containing the accepted data structure specific to that V2X information event subscription.

5.5.6.4 Unsubscribing from V2X information event notifications

When the service consumer does not want to receive notifications anymore after subscribing to V2X information events, the service consumer unsubscribes from the V2X information event notifications. Figure 5.5.6.4-1 shows a scenario where the service consumer uses REST based procedures to delete the subscription for V2X information event notifications.

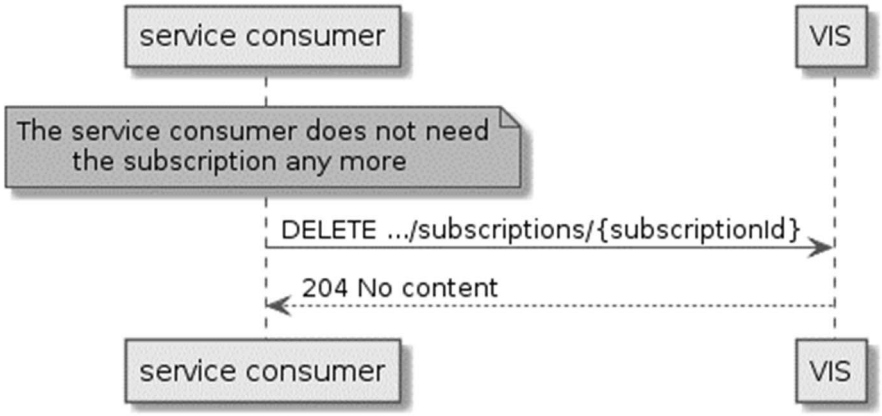


Figure 5.5.6.4-1: Flow of unsubscribing from the V2X information event notifications

Unsubscribing from the V2X information event notifications, as illustrated in figure 5.5.6.4-1, consists of the following steps.

When the service consumer does not want to receive the notifications anymore, it can unsubscribe from the V2X information notification events by deleting the subscription:

1) Service consumer sends a DELETE request to the resource representing the V2X information event subscription that was created.

2) VIS sends "204 No content" response.

5.5.7 Receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu unicast

Figure 5.5.7-1 presents the scenario where the VIS sends V2X information event notification to the service consumer about the provisioning information changes for V2X communication over Uu unicast.

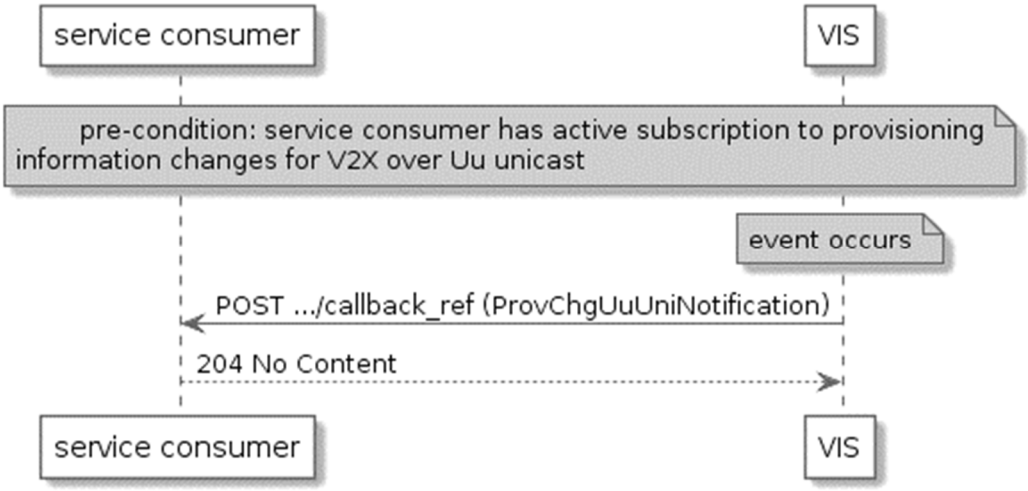


Figure 5.5.7-1: Flow of receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu unicast

Receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu unicast, as illustrated in figure 5.5.7-1, consists of the following steps:

1) VIS sends a POST request with the message body containing the ProvChgUuUniNotification data structure to the callback reference address included by the service consumer in the event subscription for the provisioning information changes for V2X communication over Uu unicast.

2) Service consumer sends a "204 No Content" response to the VIS.

5.5.8 Receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu MBMS

Figure 5.5.8-1 presents the scenario where the VIS sends V2X information event notification to the service consumer about the provisioning information changes for V2X communication over Uu MBMS.

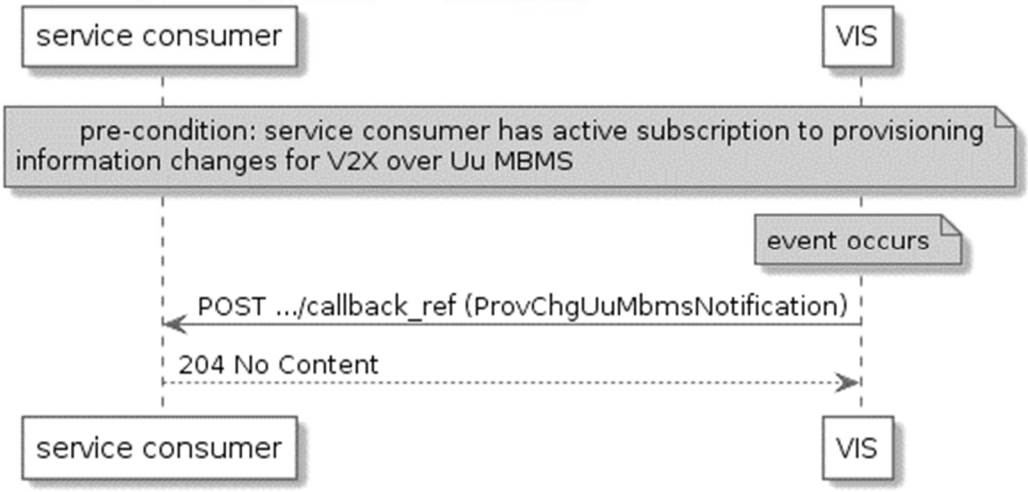


Figure 5.5.8-1: Flow of receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu MBMS

Receiving V2X information event notifications about the provisioning information changes for V2X communication over Uu MBMS, as illustrated in figure 5.5.8-1, consists of the following steps:

1) VIS sends a POST request with the message body containing the ProvChgUuMbmsNotification data structure to the callback reference address included by the service consumer in the event subscription for the provisioning information changes for V2X communication over Uu MBMS.

2) Service consumer sends a "204 No Content" response to the VIS.

5.5.9 Receiving V2X information event notifications about the provisioning information changes for V2X communication over PC5

Figure 5.5.9-1 presents the scenario where the VIS sends V2X information event notification to the service consumer about the provisioning information changes for V2X communication over PC5.

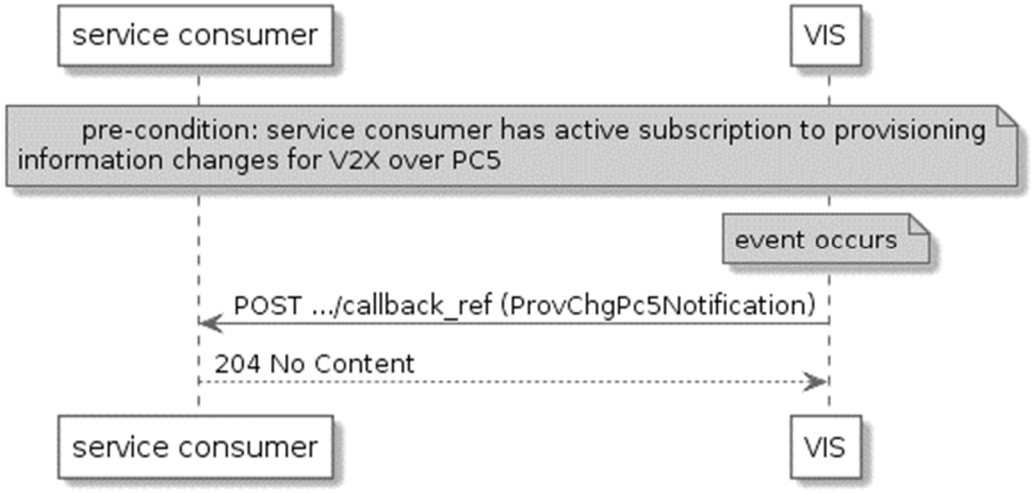


Figure 5.5.9-1: Flow of receiving V2X information event notifications about the provisioning information changes for V2X communication over PC5

Receiving V2X information event notifications about the provisioning information changes for V2X communication over PC5, as illustrated in figure 5.5.9-1, consists of the following steps:

1) VIS sends a POST request with the message body containing the ProvChgPc5Notification data structure to the callback reference address included by the service consumer in the event subscription for the provisioning information changes for V2X communication over PC5.

2) Service consumer sends a "204 No Content" response to the VIS.

5.5.10 V2X message interoperability

5.5.10.1 V2X message subscribe

The V2X message subscribe is the procedure for a service consumer (e.g. a MEC application or a MEC platform) to request to subscribe the V2X messages which come from different vehicle OEMs or operators.

Figure 5.5.10.1-1: Flow of V2X message subscribe

A service consumer requesting the subscription of V2X message from VIS, as illustrated in figure 5.5.10.1-1, consists of the following steps:

1) The service consumer sends a POST request with the message body containing the V2xMsgSubscription data structure to the resource representing V2X message subscription. The data structure defines the subscribed event and the address where the service consumer wishes to receive the event notifications.

2) VIS sends "201 Created" response with the message body containing the data structure specific to this subscription. The data structure contains the address of the resource created and the subscribed event type.

5.5.10.2 V2X message publication

The V2X message publication is the procedure for a service consumer (e.g. a MEC application or a MEC platform) to publish a V2X message to VIS who will then notify the subscribed service consumers.

Figure 5.5.10.2-1: Flow of V2X message publication

A service consumer requesting the publication of V2X message to VIS, as illustrated in figure 5.5.10.2-1, consists of the following steps:

1) Service consumer sends POST request to publish a V2X message to VIS. The POST request contains a data structure V2xMsgPublication.

2) VIS sends a "204 No Content" response.

5.5.10.3 V2X message notification

The V2X message notification is the procedure for the VIS to send a notification to service consumer (e.g. a MEC application or a MEC platform).

Figure 5.5.10.3-1: Flow of V2X message notification

VIS sends a notification to the subscribed service consumer, as illustrated in figure 5.5.10.3-1, with the following steps:

1) VIS sends a POST request with the message body containing the V2xMsgNotification data structure to the callback reference address included by the service consumer in the V2xMsgSubscription event subscription.

2) Service consumer sends a "204 No Content" response to the VIS.

5.6 Conclusions on VIS

MEC facilitates V2X interoperability in a multi-vendor, multi-network and multi-access environment. As described in previous clauses, the VIS is a required service for a MEC platform for it to enable V2X applications in a distributed edge cloud potentially spanning multiple networks and operators' systems.

The VIS is a MEC service that exposes selected V2X related information, including information available on the V2 reference point in 3GPP network. The VIS may be produced by a MEC application, e.g. the V2X application server, or by MEC platform.

The VIS facilitates the interaction between the MEC applications and the V2X Message Distribution Server (e.g. Message Broker) which is needed to distribute V2X messages related to non-session based V2X services. Alternatively, the VIS can also provide to MEC applications upon request the details to connect to V2X Message Distribution Servers for a direct interaction between a MEC application and a V2X Message Distribution Server.

The MEC platform gathers information through VIS in order to determine the required remote services and communication end points for the authorized V2X MEC applications. The MEC platform makes the required service and communication end points discoverable in the local service registry of the MEC platform.

6 Data model

6.1 Introduction

The following clauses provide the description of the data model.

6.2 Resource data types

6.2.1 Introduction

This clause defines data structures that shall be used in the resource representations.

6.2.2 Type: UuUnicastProvisioningInfo

This type represents the provisioning information required for V2X communication over Uu unicast. This information is per location (e.g. a cell of a base station or a geographical area) based.

The attributes of the UuUnicastProvisioningInfo shall follow the notations provided in table 6.2.2-1, as defined in ETSI TS 123 285 [10], ETSI TS 136 300 [11] and ETSI TS 136 423 [12].

Table 6.2.2-1: Attributes of the UuUnicastProvisioningInfo

Attribute name	Data type	Cardinality	Description
timeStamp	TimeStamp	0..1	Time stamp.
proInfoUuUnicast	Structure (inlined)	1..N	The provisioning information per location as defined below.
>locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
>v2xApplicationServer	V2xApplicationServer	1	V2X Application Server address (consisting of IP address and UDP port) for unicast.
>neighbourCellInfo	UuUniNeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over Uu unicast.

6.2.3 Type: UuMbmsProvisioningInfo

This type represents the provisioning information required for V2X communication over Uu MBMS. This information is per location (e.g. a cell of a base station or a geographical area) based.

The attributes of the UuMbmsProvisioningInfo shall follow the notations provided in table 6.2.3-1, as defined in ETSI TS 123 285 [10], ETSI TS 136 300 [11] and ETSI TS 136 423 [i.9].

Table 6.2.3-1: Attributes of the UuMbmsProvisioningInfo

Attribute name	Data type	Cardinality	Description
timeStamp	TimeStamp	0..1	Time stamp.
proInfoUuMbms	Structure (inlined)	1..N	The provisioning information per location as defined below.
>locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
>v2xServerUsd	V2xServerUsd	1	User Service Description for V2X Application Server is used to configure the UE for receiving local V2X Application Server information when it is provided over MBMS.
>neighbourCellInfo	UuMbmsNeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over Uu MBMS.

6.2.4 Type: Pc5ProvisioningInfo

This type represents the provisioning information required for V2X communication over PC5. This information is per location (e.g. a cell of a base station or a geographical area) based.

The attributes of the Pc5ProvisioningInfo shall follow the notations provided in table 6.2.4-1, as defined in ETSI TS 123 285 [10], ETSI TS 136 300 [11], ETSI TS 136 331 [14] and ETSI TS 136 321 [15].

Table 6.2.4-1: Attributes of the Pc5ProvisioningInfo

Attribute name	Data type	Cardinality	Description
timeStamp	TimeStamp	0..1	Time stamp.
proInfoPc5	Structure (inlined)	1..N	The provisioning information per location as defined below.
>locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
>dstLayer2Id	String	1	For sidelink communication, the Destination Layer-2 ID is set to the ProSe Layer-2 Group ID or Prose UE ID, see ETSI TS 136 321 [15]. PLMN operators coordinate to make sure Destination Layer-2 ID(s) for different V2X services are configured in a consistent manner.
>neighbourCellInfo	Pc5NeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over PC5.

6.2.5 Type: V2xMsgDistributionServerInfo

This type represents the information of one or more V2X Message Distribution Servers that is exchanged between the VIS and a service consumer (e.g. a MEC application) to enable a direct interaction among the service consumer and one or multiple V2X Message Distribution Servers.

The attributes of the V2xMsgDistributionServerInfo shall follow the notations provided in table 6.2.5-1.

Table 6.2.5-1: Attributes of the V2xMsgDistributionServerInfo

Attribute name	Data type	Cardinality	Description
v2xMsgDistributionServer	V2xMsgDistributionServer	1..N	Describes the information of the V2X Message Distribution Servers supported by the service consumer for direct communication.
locationInfo	LocationInfo	0..N	Location information to identify a particular geographical area of interest to the service consumer for receiving V2X messages.

6.2.6 Type: PredictedQos

This type represents the predicted QoS of a vehicular UE. This information is per UE potential route based.

The attributes of the PredictedQos shall follow the notations provided in table 6.2.6-1.

Table 6.2.6-1: Attributes of the PredictedQos

Name	Data type	Cardinality	Remarks
predictionTarget	Enum (inlined)	1	Indicates target of QoS prediction. Valid values: 1. SINGLE_UE_PREDICTION: The predicted QoS is to be intended as journey-specific for a requesting vehicular UE. 2. E2E_APPLICATION_INSTANCE_PREDICTION: The E2E user plane link between two V2X application instances, where one instance relates to a single vehicular UE and the other instance to an application instance within another network, i.e. either another vehicular UE as in the V2N2V case, or an infrastructure element as in the V2N2I case. Shall only be included in the request.
timeGranularity	TimeStamp	0..1	Time granularity of visiting a location.
locationGranularity	String	1	Granularity of visited location. Measured in meters.
noticePeriod	TimeStamp	0..1	Information on when the predicted QoS is needed at the service consumer interface. The value of the notice period depends on the application reaction that has to be triggered by the service consumer. The value of the notice period shall be equal or a multiple of the timeGranularity, if it is present. If present, it shall only be included in the request.
predictionArea	Structure (inlined)	0..1	Geographical area including the two ends of the user plane link between two V2X application instances. It shall only be present when "predictionTarget" = "E2E_APPLICATION_INSTANCE_PREDICTION".
>center	LocationInfo	1	Center of geographical area including the two ends of the user plane link between two V2X application instances.
>radius	String	1	Radius of geographical area including the two ends of the user plane link between two V2X application instances. Measured in meters.
routes	Structure (inlined)	0..N	Information relating to the potential routes of a vehicular UE. Shall only be present when "predictionTarget" = "SINGLE_UE_PREDICTION".
>routeInfo	Structure (inlined)	1..N	Information relating to a specific route. The first structure shall relate to the route origin and the last to the route destination. Intermediate waypoint locations may also be provided.
>>location	LocationInfo	1	Vehicular UE location.
>>time	TimeStamp	0..1	Estimated time at the location.

qos	Structure (inlined)	1	Predicted QoS at the related time and vehicular UE location. Shall only be included in the response.
>stream	Structure (inlined)	1..N	Predicted QoS at the related time and vehicular UE location for the specific data stream. In case of 3GPP network, this is mapped to a QoS flow. Stream needs to also contain the stream ID that, in case of the 3GPP network, can be mapped on to the 5QI or QCI.
>>qosKpi	Structure (inlined)	1..N	This structure contains the prediction for a specific QoS KPI related to a given data stream.
>>>kpiName	String	1	The name of the KPI (e.g. latency, UL bitrate, etc). It can be included in the request and in the response.
>>>kpiValue	String	1	Information on the predicted value for the specific QoS KPI. It can be in different forms, such as upper bound and lower bound, CDF, actual value, etc. Shall only be included in the response.
>>>confidence	String	0..1	Confidence of the prediction, as returned by the relevant domain PF. The value and the measurement of the confidence depends on the SLA. Shall only be included in the response.
NOTE: The data type of locationGranularity is a string which indicates the granularity of a visited location by means of latitudinal and longitudinal margins.			

NOTE: It is possible to use the PredictedQoS data type when the target of the QoS prediction is a certain area of the territory (in addition to the existing route), e.g. by defining a certain point in the space and a radius. Such case may be used by the service consumer to request a QoS prediction for the area related to the execution of a specific use case (e.g. lane merge, lane change, cross of an intersection, etc.). Such QoS prediction may refer to the collection of all the possible E2E user plane links between the vehicular UE in question and all the potential vehicular UEs and/or road infrastructure elements in the area.

6.2.7 Type: V2xMsgPublication

This type represents the V2X message that a service consumer publishes to VIS.

The attributes of the V2xMsgPublication shall follow the notations provided in table 6.2.7-1.

Table 6.2.7-1: Attributes of the V2xMsgPublication

Attribute name	Data type	Cardinality	Description
msgPropertiesValues	V2xMsgPropertiesValues	1	List of message properties that is associated to the V2X message publication that can be used for filtering messages at the reception of the V2X message.
msgRepresentationFormat	String	1	The representation format of the binary V2X message, for example base64 or hexadecimal representation.
msgContent	String	1	Published V2X message content. Its format is defined by the standardization organization indicated by the attribute stdOrganization of the msgPropertiesValues attribute.

6.3 Subscription data types

6.3.1 Introduction

This clause defines data structures for subscriptions.

6.3.2 Type: ProvChgUuUniSubscription

This type represents a subscription to the notifications from VIS about the provisioning information changes for V2X communication over Uu unicast.

The attributes of the ProvChgUuUniSubscription shall follow the indications provided in table 6.3.2-1.

Table 6.3.2-1: Attributes of the ProvChgUuUniSubscription

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "ProvChgUuUniSubscription".
callbackReference	Uri	0..1	URI exposed by the client on which to receive notifications via HTTP. See note.
requestTestNotification	Boolean	0..1	Shall be set to TRUE by the service consumer to request a test notification via HTTP on the callbackReference URI, as described in ETSI GS MEC 009 [9], clause 6.12a. Default: FALSE.
websockNotifConfig	WebsockNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between VIS and the service consumer for notifications. See note.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the VIS API as it acts as an ID for the subscription.
filterCriteria	Structure (inlined)	1	List of filtering criteria for the subscription. Any filtering criteria from below, which is included in the request, shall also be included in the response.
>locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
>v2xApplicationServer	V2xApplicationServer	1	V2X Application Server address (consisting of IP address and UDP port) for unicast.
>neighbourCellInfo	UuUniNeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over Uu unicast.
expiryDeadline	TimeStamp	0..1	Time stamp.
NOTE: At least one of callbackReference and websockNotifConfig shall be provided by the service consumer. If both are provided, it is up to VIS to choose an alternative and return only that alternative in the response, as described in ETSI GS MEC 009 [9], clause 6.12a.			

6.3.3 Type: ProvChgUuMbmsSubscription

This type represents a subscription to the notifications from VIS about the provisioning information changes for V2X communication over Uu MBMS.

The attributes of the ProvChgUuMbmsSubscription shall follow the indications provided in table 6.3.3-1.

Table 6.3.3-1: Attributes of the ProvChgUuMbmsSubscription

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "ProvChgUuMbmsSubscription".
callbackReference	Uri	0..1	URI exposed by the client on which to receive notifications via HTTP. See note.
requestTestNotification	Boolean	0..1	Shall be set to TRUE by the service consumer to request a test notification via HTTP on the callbackReference URI, as described in ETSI GS MEC 009 [9], clause 6.12a. Default: FALSE.
websockNotifConfig	WebsockNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between VIS and the service consumer for notifications. See note.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the V2X API as it acts as an ID for the subscription.
filterCriteria	Structure (inlined)	1	List of filtering criteria for the subscription. Any filtering criteria from below, which is included in the request, shall also be included in the response.
>locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
>v2xServerUsd	V2xServerUsd	1	User Service Description for V2X Application Server is used to configure the UE for receiving local V2X Application Server information when it is provided over MBMS.
>neighbourCellInfo	UuMbmsNeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over Uu MBMS.
expiryDeadline	TimeStamp	0..1	Time stamp.
NOTE: At least one of callbackReference and websockNotifConfig shall be provided by the service consumer. If both are provided, it is up to VIS to choose an alternative and return only that alternative in the response, as described in ETSI GS MEC 009 [9], clause 6.12a.			

6.3.4 Type: ProvChgPc5Subscription

This type represents a subscription to the notifications from VIS about the provisioning information changes for V2X communication over PC5.

The attributes of the ProvChgPc5Subscription shall follow the indications provided in table 6.3.4-1.

Table 6.3.4-1: Attributes of the ProvChgPc5Subscription

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "ProvChgPc5Subscription".
callbackReference	Uri	0..1	URI exposed by the client on which to receive notifications via HTTP. See note.
requestTestNotification	Boolean	0..1	Shall be set to TRUE by the service consumer to request a test notification via HTTP on the callbackReference URI, as described in ETSI GS MEC 009 [9], clause 6.12a. Default: FALSE.
websockNotifConfig	WebsockNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between VIS and the service consumer for notifications. See note.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the V2X API as it acts as an ID for the subscription.
filterCriteria	Structure (inlined)	1	List of filtering criteria for the subscription. Any filtering criteria from below, which is included in the request, shall also be included in the response.
>locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
>dstLayer2Id	String	1	For sidelink communication, the Destination Layer-2 ID is set to the ProSe Layer-2 Group ID or Prose UE ID, see ETSI TS 136 321 [15].
>neighbourCellInfo	Pc5NeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over PC5.
expiryDeadline	TimeStamp	0..1	Time stamp.
NOTE: At least one of callbackReference and websockNotifConfig shall be provided by the service consumer. If both are provided, it is up to VIS to choose an alternative and return only that alternative in the response, as described in ETSI GS MEC 009 [9], clause 6.12a.			

6.3.5 Type: V2xMsgSubscription

This type represents a subscription to notification of V2X message.

The attributes of the V2xMsgSubscription shall follow the notations provided in table 6.3.5-1.

Table 6.3.5-1: Attributes of the V2xMsgSubscription

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "V2xMsgSubscription".
callbackReference	Uri	0..1	URI exposed by the client on which to receive notifications via HTTP. See note.
requestTestNotification	Boolean	0..1	Shall be set to TRUE by the service consumer to request a test notification via HTTP on the callbackReference URI, as described in ETSI GS MEC 009 [9], clause 6.12a. Default: FALSE.
websockNotifConfig	WebsockNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between VIS and the service consumer for notifications. See note.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the VIS API as it acts as an ID for the subscription.
filterCriteria	V2xMsgFilterCriteria	1	List of filtering criteria aimed to filter V2X messages at the reception. Any filtering criteria which is included in the request, shall also be included in the response.
expiryDeadline	TimeStamp	0..1	Time stamp.
NOTE: At least one of callbackReference and websockNotifConfig shall be provided by the service consumer. If both are provided, it is up to VIS to choose an alternative and return only that alternative in the response, as described in ETSI GS MEC 009 [9], clause 6.12a.			

6.3.6 Type: PredQosSubscription

This type represents a subscription to notification of predicted QoS information.

The attributes of the PredQosSubscription shall follow the notations provided in table 6.3.6-1.

Table 6.3.6-1: Attributes of the PredQosSubscription

Attribute name	Data type	Cardinality	Description
subscriptionType	String	1	Shall be set to "PredQosSubscription".
callbackReference	Uri	0..1	URI exposed by the client on which to receive notifications via HTTP. See note.
requestTestNotification	Boolean	0..1	Shall be set to TRUE by the service consumer to request a test notification via HTTP on the callbackReference URI, as described in ETSI GS MEC 009 [9], clause 6.12a. Default: FALSE.
websockNotifConfig	WebsockNotifConfig	0..1	Provides details to negotiate and signal the use of a Websocket connection between VIS and the service consumer for notifications. See note.
_links	Structure (inlined)	0..1	Hyperlink related to the resource. This shall be only included in the HTTP responses and in HTTP PUT requests.
>self	LinkType	1	Self-referring URI. The URI shall be unique within the V2X API as it acts as an ID for the subscription.
filterCriteria	QosPredFilterCriteria	1	List of filtering criteria aimed to filter QoS predictions at the reception. Any filtering criteria included in the request, shall also be included in the response.
expiryDeadline	TimeStamp	0..1	Time stamp.
NOTE: At least one of callbackReference and websockNotifConfig shall be provided by the service consumer. If both are provided, it is up to VIS to choose an alternative and return only that alternative in the response, as described in ETSI GS MEC 009 [9], clause 6.12a.			

6.3.7 Type: SubscriptionLinkList

This type represents a list of links related to currently existing subscriptions for the service consumer. This information is returned when sending a request to receive current subscriptions.

Table 6.3.7-1: Attributes of the SubscriptionLinkList

Attribute name	Data type	Cardinality	Description
_links	Structure (inlined)	1	List of hyperlinks related to the resource.
>self	LinkType	1	URI of this resource.
>subscriptions	Structure (inlined)	0..N	The service consumer's subscriptions.
>>href	Uri	1	The URI referring to the subscription.
>>subscriptionType	String	1	Type of the subscription. The values are as defined in the "subscriptionType" attribute for each different V2X information event subscription data type.

6.4 Notifications data types

6.4.1 Introduction

This clause defines data structures that define notifications.

6.4.2 Type: ProvChgUuUniNotification

This type represents a notification from VIS with regards to the provisioning information changes for V2X communication over Uu unicast. The notification is sent by the VIS to inform about the changes.

The attributes of the ProvChgUuUniNotification shall follow the indications provided in table 6.4.2-1.

Table 6.4.2-1: Attributes of the ProvChgUuUniNotification

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "ProvChgUuUniNotification".
timeStamp	TimeStamp	0..1	Time stamp.
locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
v2xApplicationServer	V2xApplicationServer	0..1	V2X Application Server address (consisting of IP address and UDP port) for unicast.
neighbourCellInfo	UuUniNeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over Uu unicast.

6.4.3 Type: ProvChgUuMbmsNotification

This type represents a notification from VIS with regards to the provisioning information changes for V2X communication over Uu MBMS. The notification is sent by the VIS to inform about the changes.

The attributes of the ProvChgUuMbmsNotification shall follow the indications provided in table 6.4.3-1.

Table 6.4.3-1: Attributes of the ProvChgUuMbmsNotification

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "ProvChgUuMbmsNotification".
timeStamp	TimeStamp	0..1	Time stamp.
locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
v2xServerUsd	V2xServerUsd	0..1	User Service Description for V2X Application Server is used to configure the UE for receiving local V2X Application Server information when it is provided over MBMS.
neighbourCellInfo	UuMbmsNeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over Uu MBMS.

6.4.4 Type: ProvChgPc5Notification

This type represents a notification from VIS with regards to the provisioning information changes for V2X communication over PC5. The notification is sent by the VIS to inform about the changes.

The attributes of the ProvChgPc5Notification shall follow the indications provided in table 6.4.4-1.

Table 6.4.4-1: Attributes of the ProvChgPc5Notification

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "ProvChgPc5Notification".
timeStamp	TimeStamp	0..1	Time stamp.
locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area.
dstLayer2Id	String	0..1	For sidelink communication, the Destination Layer-2 ID is set to the ProSe Layer-2 Group ID or Prose UE ID, see ETSI TS 136 321 [15].
neighbourCellInfo	Pc5NeighbourCellInfo	0..N	The information of the neighbour cells in a visiting PLMN that support V2X communication over PC5.

6.4.5 Type: V2xMsgNotification

This type represents a notification for informing the subscribers about the V2X message.

The attributes of the V2xMsgNotification shall follow the notations provided in table 6.4.5-1.

Table 6.4.5-1: Attributes of the V2xMsgNotification

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "V2xMsgNotification".
timeStamp	TimeStamp	1	Date and time of the notification generation.
msgPropertiesValues	V2xMsgPropertiesValues	1	List of message properties that is associated to the V2X message.
msgRepresentationFormat	String	1	The representation format of the binary V2X message, for example base64 or hexadecimal representation.
msgContent	String	1	Published V2X message content. The format of the string is defined by the standardization organization indicated by the attribute stdOrganization of the msgPropertiesValues attribute.
_links	Structure (inlined)	1	Links to resources related to this notification.
>subscription	LinkType	1	A link to the related subscription.

6.4.6 Type: TestNotification

This type represents a test notification from VIS to determine if the Websocket method is to be utilized for the VIS to issue notifications for a subscription, as defined in clause 6.12a of ETSI GS MEC 009 [9].

Table 6.4.6-1: Attributes of the TestNotification

Attribute name	Data type	Cardinality	Description
notificationType	String	1	Shall be set to "TestNotification".
_links	Structure (inlined)	1	Hyperlink related to the resource.
>subscription	LinkType	1	URI identifying the subscription for the test notification.

6.5 Referenced structured data types

6.5.1 Introduction

This clause defines data structures that are referenced from data structures defined in the previous clauses, but are neither resource representations nor bound to any pub/sub mechanism.

6.5.2 Type: TimeStamp

This type represents a time stamp.

Table 6.5.2-1: Attributes of the TimeStamp

Attribute name	Data type	Cardinality	Description
seconds	Uint32	1	The seconds part of the time. Time is defined as Unix-time since January 1, 1970, 00:00:00 UTC.
nanoSeconds	Uint32	1	The nanoseconds part of the time. Time is defined as Unix-time since January 1, 1970, 00:00:00 UTC.

6.5.3 Type: LocationInfo

This type represents the location information.

The attributes of the LocationInfo shall follow the notations provided in table 6.5.3-1.

Table 6.5.3-1: Attributes of the LocationInfo

Attribute name	Data type	Cardinality	Description
ecgi	Ecgi	0..1	E-UTRAN Cell Global Identifier of the serving cell.
geoArea	Structure (inlined)	0..1	Information of a geographical area.
>latitude	Float	1	Latitude (DATUM = WGS84) -90 to 90 in decimal degree format DDD.ddd
>longitude	Float	1	Longitude (DATUM = WGS84) -180 to 180 in decimal degree format DDD.ddd
NOTE: Either ecgi or geoArea shall be present, but not both.			

6.5.4 Type: Plmn

This type represents a PLMN Identity.

The attributes of the Plmn shall follow the notations provided in table 6.5.4-1, as defined in ETSI TS 136 413 [13].

Table 6.5.4-1: Attributes of the Plmn

Attribute name	Data type	Cardinality	Description
mcc	String	1	The Mobile Country Code part of PLMN Identity.
mnc	String	1	The Mobile Network Code part of PLMN Identity.

6.5.5 Type: Ecgi

This type represents an E-UTRAN Cell Global Identifier.

The attributes of the Ecgi shall follow the notations provided in table 6.5.5-1, as defined in ETSI TS 136 423 [12] and ETSI TS 136 413 [13].

Table 6.5.5-1: Attributes of the Ecgi

Attribute name	Data type	Cardinality	Description
plmn	Plmn	1	Public Land Mobile Network Identity.
cellId	CellId	1	The E-UTRAN Cell Identity.

6.5.6 Type: FddInfo

This type represents an FDD information.

The attributes of the FddInfo shall follow the notations provided in table 6.5.6-1, as defined in ETSI TS 136 423 [12].

Table 6.5.6-1: Attributes of the FddInfo

Attribute name	Data type	Cardinality	Description
ulEarfcn	Earfcn	1	
dlEarfcn	Earfcn	1	
ulTransmissionBandwidth	TransmissionBandwidth	1	
dlTransmissionBandwidth	TransmissionBandwidth	1	

6.5.7 Type: TddInfo

This type represents a TDD information.

The attributes of the TddInfo shall follow the notations provided in table 6.5.7-1, as defined in ETSI TS 136 423 [12].

Table 6.5.7-1: Attributes of the TddInfo

Attribute name	Data type	Cardinality	Description
earfcn	Earfcn	1	
transmissionBandwidth	TransmissionBandwidth	1	
subframeAssignment	String	1	Uplink-downlink subframe configuration information.

6.5.8 Type: V2xApplicationServer

This type represents a V2X Application Server address (consisting of IP address and UDP port) for unicast.

Table 6.5.8-1: Attributes of the V2xApplicationServer

Attribute name	Data type	Cardinality	Description
ipAddress	String	1	
udpPort	String	1	

6.5.9 Type: UuUniNeighbourCellInfo

This type represents a neighbour cell information in a visiting PLMN that supports V2X communication over Uu unicast.

Table 6.5.9-1: Attributes of the UuUniNeighbourCellInfo

Attribute name	Data type	Cardinality	Description
plmn	Plmn	1	Public Land Mobile Network Identity.
ecgi	Ecgi	1	E-UTRAN Cell Global Identifier.
pci	Integer	1	Physical Cell Identifier.
fddInfo	FddInfo	1	Information for FDD operation.
tddInfo	TddInfo	1	Information for TDD operation.

6.5.10 Type: V2xServerUsd

This type represents a User Service Description for V2X Application Server.

Table 6.5.10-1: Attributes of the V2xServerUsd

Attribute name	Data type	Cardinality	Description
tmgi	Structure (inlined)		Temporary Mobile Group Identity (TMGI), which is used within MBMS to uniquely identify Multicast and Broadcast bearer services.
>mbmsServiceId	String	1	MBMS Service ID consisting of three octets.
>mcc	String	1	The Mobile Country Code part of PLMN Identity.
>mnc	String	1	The Mobile Network Code part of PLMN Identity.
serviceAreaIdentifier	String	1..N	A list of service area identifier for the applicable MBMS broadcast area.
sdpInfo	Structure (inlined)	1	SDP with IP multicast address and port number used for V2X communication via MBMS.
>ipMulticastAddress	String	1	
>portNumber	String	1	

6.5.11 Type: UuMbmsNeighbourCellInfo

This type represents a neighbour cell information in a visiting PLMN that supports V2X communication over Uu MBMS.

Table 6.5.11-1: Attributes of the UuMbmsNeighbourCellInfo

Attribute name	Data type	Cardinality	Description
plmn	Plmn	1	Public Land Mobile Network Identity.
ecgi	Ecgi	1	E-UTRAN Cell Global Identifier.
pci	Integer	1	Physical Cell Identifier.
fddInfo	FddInfo	1	Information for FDD operation.
tddInfo	TddInfo	1	Information for TDD operation.
mbmsServiceAreaIdentity	String	1..N	Supported MBMS Service Area Identities in the cell.

6.5.12 Type: Pc5NeighbourCellInfo

This type represents a neighbour cell information in a visiting PLMN that supports V2X communication over PC5.

Table 6.5.12-1: Attributes of the Pc5NeighbourCellInfo

Attribute name	Data type	Cardinality	Description
plmn	Plmn	1	Public Land Mobile Network Identity.
ecgi	Ecgi	1	E-UTRAN Cell Global Identifier.
siV2xConfig	SystemInformationBlockType21	1	V2X sidelink communication configuration, as defined in ETSI TS 136 331 [14].

6.5.13 Type: LinkType

This type represents a type of link.

Table 6.5.13-1: Attributes of the LinkType

Attribute name	Data type	Cardinality	Description
href	Uri	1	URI referring to a resource

6.5.14 Type: V2xMsgPropertiesValues

This type represents the properties that can be associated to a V2X message when it is published, or a notification in relation to the V2X message is issued. These properties are also used to match the filter criteria defined in the message subscription.

Table 6.5.14-1: Attributes of the V2xMsgPropertiesValues

Attribute name	Data type	Cardinality	Description
stdOrganization	Enum	1	Standardization organization which defines the published V2X message type: • ETSI: European Telecommunications Standards Institute. See note 1.
msgType	Enum	1	Published V2X message type. Its value is defined by the standardization organization indicated by the attribute stdOrganization. See note 2.
msgProtocolVersion	Integer	1	Protocol version of the V2X message (0..255), See note 3.
locationInfo	LocationInfo	1	Location information to identify a cell of a base station or a particular geographical area to which the V2X message can be associated.
NOTE 1: Other standardization organizations could be added as needed. NOTE 2: The V2X message types of ETSI shall be used as specified in ETSI TS 102 894-2 [6], clause A.114. NOTE 3: The message protocol version of ETSI shall be used as specified in ETSI TS 102 894-2 [6], clause A.114.			

6.5.15 Type: V2xMsgFilterCriteria

This type represents the filter criteria that can be used when performing a subscription to notification of V2X messages.

Table 6.5.15-1: Attributes of the V2xMsgFilterCriteria

Attribute name	Data type	Cardinality	Description
stdOrganization	Enum	1	Standardization organization which defines the published V2X message type: • ETSI: European Telecommunications Standards Institute. See note 1.
msgType	Enum	0..N	Published V2X message type. Its value is defined by the standardization organization indicated by the attribute stdOrganization. See note 2.
msgProtocolVersion	Integer	0..N	Protocol version of the V2X message (0..255), See note 3.
locationInfo	LocationInfo	0..N	Location information to identify a cell of a base station or a particular geographical area to which the V2X message can be associated.
NOTE 1: Other standardization organizations could be added as needed. NOTE 2: The V2X message types of ETSI shall be used as specified in ETSI TS 102 894-2 [6], clause A.114. NOTE 3: The message protocol version of ETSI shall be used as specified in ETSI TS 102 894-2 [6], clause A.114.			

6.5.16 Type: V2xMsgDistributionServer

This type represents the information about a V2X Message Distribution Server that can be exchanged between a service consumer (e.g. a MEC application) and the VIS. The service consumer uses this data structure for providing the criteria the V2X Message Distribution Server needs to fulfil for direct interaction with it. The VIS uses this data structure to provide to the service consumer the information needed to directly connect to V2X Message Distribution Servers fulfilling these criteria.

Table 6.5.16-1: Attributes of the V2xMsgDistributionServer

Attribute name	Data type	Cardinality	Description
infoProtocol	Structure (inlined)	1	Specifics of the application layer protocol of V2X Message Distribution Server.
>msgProtocol	Enum	1..N	Numerical value corresponding to the application layer protocol supported by the service consumer. For the msgProtocol, the following values are currently defined (see note): 0 = MQTT v3.1.0 1 = MQTT v3.1.1 2 = MQTT v5 3 = MQTT-SN 4 = AMQP 1.0
>protImplementation	String	1	Implementation specifics of application layer protocol, e.g. programming language.
infoConnection	Structure (inlined)	0..1	Connection information of the V2X Message Distribution Server the service consumer can use for direct connection. Shall only be included in the response.
>ipAddress	String	1	IP address of the V2X Message Distribution Server.
>port_number	Integer (0..65535)	1	Port number of the V2X Message Distribution Server.
NOTE: Other application layer protocols (and versions thereof) may be added as needed.			

6.5.17 Type: QosPredFilterCriteria

This type represents the filter criteria that can be used when performing a subscription to notification of QoS prediction messages.

Table 6.5.17-1: Attributes of the QosPredFilterCriteria

Attribute name	Data type	Cardinality	Description
streamId	String	0..1	The predicted QoS at the related time and vehicular UE location is provided for a specific data stream. In case of 3GPP network, this is mapped to a QoS flow. Stream needs to also contain the stream ID which, in case of the 3GPP network, can be mapped on to the 5QI or QCI. See note.
NOTE: Other identifiers could be added as needed.			

6.5.18 Type: WebsockNotifConfig

This type represents configuration for the delivery of subscription notifications over Websockets per the pattern defined in clause 6.12a of ETSI GS MEC 009 [9].

Table 6.5.18-1: Attributes of the WebsockNotifConfig

Attribute name	Data type	Cardinality	Description
websocketUri	Uri	0..1	Set by VIS to indicate to the service consumer the Websocket URI to be used for delivering notifications.
requestWebsocketUri	Boolean	0..1	Set to TRUE by the service consumer to indicate that Websocket delivery is requested.

6.6 Referenced simple data types and enumerations

6.6.1 Introduction

This clause defines simple data types and enumerations that can be referenced from data structures defined in the previous clauses.

6.6.2 Type: CellId

This type represents an E-UTRAN Cell Identifier.

The attributes of the CellId shall follow the notations provided in table 6.6.2-1, as defined in ETSI TS 136 413 [13].

Table 6.6.2-1: Attributes of the CellId

Attribute name	Data type	Cardinality	Description
cellId	String	1	E-UTRAN Cell Identity as a bit string (size (28)).

6.6.3 Type: Earfcn

This type represents an E-UTRA Absolute Radio Frequency Channel Number, which defines the carrier frequency used in a cell for a given direction (UL or DL) in FDD or for both UL and DL directions in TDD.

The attributes of the EARFCN shall follow the notations provided in table 6.6.3-1, as defined in ETSI TS 136 413 [13].

Table 6.6.3-1: Attributes of the Earfcn

Attribute name	Data type	Cardinality	Description
earfcn	Integer	1	E-UTRA Absolute Radio Frequency Channel Number, range (0... 65535)

6.6.4 Type: TransmissionBandwidth

This type represents a Transmission Bandwidth, which is used to indicate the UL or DL transmission bandwidth expressed in units of resource blocks. The values bw6, bw15, bw25, bw50, bw75, bw100 correspond to the number of resource blocks 6, 15, 25, 50, 75, 100.

The attributes of the TransmissionBandwidth shall follow the notations provided in table 6.6.4-1, as defined in ETSI TS 136 413 [13].

Table 6.6.4-1: Attributes of the TransmissionBandwidth

Attribute name	Data type	Cardinality	Description
transmissionBandwidth	Enum	1	Numeric value corresponding to the transmission bandwidth expressed in units of resource blocks as follows: 1 = bw6 (6 resource blocks) 2 = bw15 (15 resource blocks) 3 = bw25 (25 resource blocks) 4 = bw50 (50 resource blocks) 5 = bw75 (75 resource blocks) 6 = bw100 (100 resource blocks)

7 API definition

7.1 Introduction

This clause defines the resources and operations of the VIS API.

7.2 Global definitions and resource structure

All resource URIs of this API shall have the following root:

{apiRoot}/{apiName}/{apiVersion}/

"apiRoot" and "apiName" are discovered using the service registry. It includes the scheme ("https"), host and optional port, and an optional prefix string. The "apiName" shall be set to "vis" and "apiVersion" shall be set to "v2" for the present document.

The API shall support HTTP over TLS (also known as HTTPS) using TLS version 1.2 (as defined by IETF RFC 5246 [3]). TLS 1.3 (including the new specific requirements for TLS 1.2 implementations) defined by IETF RFC 8446 [7] should be supported. HTTP without TLS shall not be used. Versions of TLS earlier than 1.2 shall neither be supported nor used.

TLS implementations should meet or exceed the security algorithm, key length and strength requirements specified in clause 6.2.3 of ETSI GS MEC 009 [9] (if TLS version 1.2 as defined by IETF RFC 5246 [3] is used) or clause 6.2.2 of ETSI GS MEC 009 [9] (if TLS version 1.3 as defined by IETF RFC 8446 [7] is used) of ETSI TS 133 210 [8] (3GPP Release 16 or later).

The content format of JSON shall be supported. The JSON format is signalled by the content type "application/json".

This API shall require the use of the OAuth 2.0 client credentials grant type according to IETF RFC 6749 [4] with bearer tokens according to IETF RFC 6750 [5]. See clause 6.16 of ETSI GS MEC 009 [9] for more information. How the token endpoint and client credentials are provisioned into the MEC applications is out of scope of the present document.

This API supports additional application-related error information to be provided in the HTTP response when an error occurs. See clause 6.15 of ETSI GS MEC 009 [9] for more information.

Figure 7.2-1 illustrates the resource URI structure of this API. Table 7.2-1 provides an overview of the resources defined by the present document, and the applicable HTTP methods.

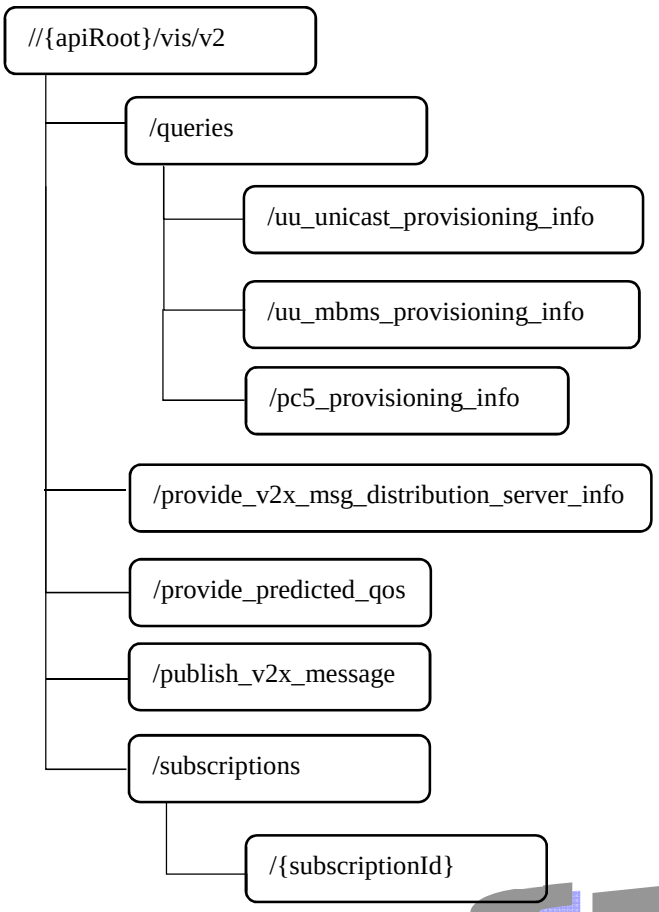


Figure 7.2-1: Resource URI structure of the VIS API

Table 7.2-1: Resources and methods overview

Resource name	Resource URI	HTTP method	Meaning
Uu unicast provisioning information	/queries/uu_unicast_provisioning_info	GET	Retrieve provisioning information required for V2X communication over Uu unicast.
Uu MBMS provisioning information	/queries/uu_mbms_provisioning_info	GET	Retrieve provisioning information required for V2X communication over Uu MBMS.
PC5 provisioning information	/queries/pc5_provisioning_info	GET	Retrieve provisioning information required for V2X communication over PC5.
Provide V2X Message Distribution Server information	/provide_v2x_msg_distribution_server_info	POST	Provide V2X Message Distribution Servers information required for direct connection with a service consumer.
Provide predicted QoS task	/provide_predicted_qos	POST	Provide predicted QoS based on route information.
Publish V2X message task	/publish_v2x_message	POST	Publish a V2X message to VIS.
All subscriptions for a subscriber	/subscriptions	GET	Retrieve a list of active subscriptions for this subscriber.
		POST	Create a new subscription.
Existing subscription	/subscriptions/{subscriptionId}	GET	Retrieve information on current specific subscription.
		PUT	Modify existing subscription by sending a new data structure.
		DELETE	Cancel the existing subscription.
Notification callback	Client provided callback reference	POST	Send a notification.

NOTE: The present version of the document supports a single VIS consumer requesting QoS predictions based on either alternative planned routes of the corresponding single vehicular UE or based on a prediction area involving possibly multiple E2E data paths between application instances.

7.3 Resource: uu_unicast_provisioning_info

7.3.1 Description

This resource is queried to retrieve information required for V2X communication over Uu unicast.

7.3.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/queries/uu_unicast_provisioning_info**

This resource shall support the resource URI variables defined in table 7.3.2-1.

Table 7.3.2-1: Resource URI variables for resource "uu_unicast_provisioning_info"

Name	Definition
apiRoot	See clause 7.2

7.3.3 Resource methods

7.3.3.1 GET

The GET method is used to query provisioning information for V2X communication over Uu unicast.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.3.3.1-1 and 7.3.3.1-2.

Table 7.3.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
location_info	String	1	Comma separated list of locations to identify a cell of a base station or a particular geographical area, formatted as follows for the two cases: …/uu_unicast_provisioning_info?location_info=ecgi,{String} Where the String is made up of 1 to N comma separated ecgi values (reference clause 6.5.5). …/uu_unicast_provisioning_info?location_info=latitude,{String},longitude,{String} Where the two strings are made up of 1 to N comma separated latitude and longitude values respectively (reference clause 6.5.3), such that the number of latitude and longitude values shall be equal. Examples of query formats when N=2 locations are provided below: …/uu_unicast_provisioning_info?location_info=ecgi,1357924680,1357924681 …/uu_unicast_provisioning_info?location_info=latitude,000.000,001.000,longitude,000.000,001.000

Table 7.3.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	UuUnicastProvisioningInfo	1	200 OK	Upon success, a response body containing the Uu unicast provisioning information is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.3.3.2 PUT

Not applicable.

7.3.3.3 PATCH

Not applicable.

7.3.3.4 POST

Not applicable.

7.3.3.5 DELETE

Not applicable.

7.4 Resource: uu_mbms_provisioning_info

7.4.1 Description

This resource is queried to retrieve information required for V2X communication over Uu MBMS.

7.4.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/queries/uu_mbms_provisioning_info**

This resource shall support the resource URI variables defined in table 7.4.2-1.

Table 7.4.2-1: Resource URI variables for resource "uu_mbms_provisioning_info"

Name	Definition
apiRoot	See clause 7.2

7.4.3 Resource methods

7.4.3.1 GET

The GET method is used to query provisioning information for V2X communication over Uu MBMS.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.4.3.1-1 and 7.4.3.1-2.

Table 7.4.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
location_info	String	1	Comma separated list of locations to identify a cell of a base station or a particular geographical area, formatted as follows for the two cases: …/uu_mbms_provisioning_info?location_info=ecgi,{String} Where the String is made up of 1 to N comma separated ecgi values (reference clause 6.5.5). …/uu_mbms_provisioning_info?location_info=latitude,{String},longitude,{String} Where the two strings are made up of 1 to N comma separated latitude and longitude values respectively (reference clause 6.5.3), such that the number of latitude and longitude values shall be equal. Examples of query formats when N=2 locations are provided below: …/uu_mbms_provisioning_info?location_info=ecgi,1357924680,1357924681 …/uu_mbms_provisioning_info?location_info=latitude,000.000,001.000,longitude,000.000,001.000

Table 7.4.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	UuMbmsProvisioningInfo	1	200 OK	Upon success, a response body containing the Uu MBMS provisioning information is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.4.3.2 PUT

Not applicable.

7.4.3.3 PATCH

Not applicable.

7.4.3.4 POST

Not applicable.

7.4.3.5 DELETE

Not applicable.

7.5 Resource: pc5_provisioning_info

7.5.1 Description

This resource is queried to retrieve information required for V2X communication over PC5.

7.5.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/queries/pc5_provisioning_info**

This resource shall support the resource URI variables defined in table 7.5.2-1.

Table 7.5.2-1: Resource URI variables for resource "pc5_provisioning_info"

Name	Definition
apiRoot	See clause 7.2

7.5.3 Resource methods

7.5.3.1 GET

The GET method is used to query provisioning information for V2X communication over PC5.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.5.3.1-1 and 7.5.3.1-2.

Table 7.5.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
location_info	String	1	Comma separated list of locations to identify a cell of a base station or a particular geographical area, formatted as follows for the two cases: …/pc5_provisioning_info?location_info=ecgi,{String} Where the String is made up of 1 to N comma separated ecgi values (reference clause 6.5.5). …/pc5_provisioning_info?location_info=latitude,{String},longitude,{String} Where the two strings are made up of 1 to N comma separated latitude and longitude values respectively (reference clause 6.5.3), such that the number of latitude and longitude values shall be equal. Examples of query formats when N=2 locations are provided below: …/pc5_provisioning_info?location_info=ecgi,1357924680,1357924681 …/pc5_provisioning_info?location_info=latitude,000.000,001.000,longitude,000.000,001.000

Table 7.5.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	Pc5ProvisioningInfo	1	200 OK	Upon success, a response body containing the PC5 provisioning information is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.5.3.2 PUT

Not applicable.

7.5.3.3 PATCH

Not applicable.

7.5.3.4 POST

Not applicable.

7.5.3.5 DELETE

Not applicable.

7.6 Resource: provide_v2x_msg_distribution_server_info

7.6.1 Description

This task resource represents V2X Message Distribution Servers information required for the direct connection of a service consumer to V2X Message Distribution Servers.

7.6.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/provide_v2x_msg_distribution_server_info**

This resource shall support the resource URI variables defined in table 7.6.2-1.

Table 7.6.2-1: Resource URI variables for resource "provide_v2x_msg_distribution_server_info"

Name	Definition
apiRoot	See clause 7.2

7.6.3 Resource methods

7.6.3.1 GET

Not applicable.

7.6.3.2 PUT

Not applicable.

7.6.3.3 PATCH

Not applicable.

7.6.3.4 POST

The POST method is used to request the information about available V2X Message Distribution Servers that can be supported by the service consumer (e.g. a MEC application).

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.6.3.4-1 and 7.6.3.4-2.

Table 7.6.3.4-1: URI query parameters supported by the POST method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.6.3.4-2: Data structures supported by the POST request/response on this resource

	Data type	Cardinality	Remarks	
Request body	V2xMsgDistributionServerInfo	1	Entity body in the request contains the information of the V2X Message Distribution Servers supported by the service consumer for direct communication.	
	Data type	**Cardinality**	**Response Codes**	**Remarks**
Response body	V2xMsgDistributionServerInfo	1	200 OK	The response body shall contain the connection information of the V2X Message Distribution Servers that the service consumer can use for direct connection.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.6.3.5 DELETE

Not applicable.

7.7 Resource: provide_predicted_qos

7.7.1 Description

This task resource represents predicted QoS information for a vehicular UE with potential routes.

7.7.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/provide_predicted_qos**

This resource shall support the resource URI variables defined in table 7.7.2-1.

Table 7.7.2-1: Resource URI variables for resource "provide_predicted_qos"

Name	Definition
apiRoot	See clause 7.2

7.7.3 Resource methods

7.7.3.1 GET

Not applicable.

7.7.3.2 PUT

Not applicable.

7.7.3.3 PATCH

Not applicable.

7.7.3.4 POST

The POST method is used to request the predicted QoS correspondent to potential routes of a vehicular UE.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.7.3.4-1 and 7.7.3.4-2.

Table 7.7.3.4-1: URI query parameters supported by the POST method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.7.3.4-2: Data structures supported by the POST request/response on this resource

Request body	**Data type**	**Cardinality**	**Remarks**	
	PredictedQos	1	Entity body in the request contains the predicted QoS as requested by the VIS service consumer.	
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	PredictedQos	1	200 OK	The response body shall contain the predicted QoS corresponding to potential routes of a vehicular UE.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.7.3.5 DELETE

Not applicable.

7.8 Resource: publish_v2x_message

7.8.1 Description

This resource represents a V2X message that is published by a service consumer (e.g. a MEC application or a MEC platform) to VIS.

7.8.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/publish_v2x_message**

This resource shall support the resource URI variables defined in table 7.8.2-1.

Table 7.8.2-1: Resource URI variables for resource "publish_v2x_message"

Name	Definition
apiRoot	See clause 7.2

7.8.3 Resource methods

7.8.3.1 GET

Not applicable.

7.8.3.2 PUT

Not applicable.

7.8.3.3 PATCH

Not applicable.

7.8.3.4 POST

The POST method is used to publish a V2X message.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.8.3.4-1 and 7.8.3.4-2.

Table 7.8.3.4-1: URI query parameters supported by the POST method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.8.3.4-2: Data structures supported by the POST request/response on this resource

Request body	**Data type**	**Cardinality**	**Remarks**	
	V2xMsgPublication	1	Entity body in the request contains the V2X message to be published.	
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	n/a		204 No Content	Upon success, a response 204 No Content without any response body is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered.

7.8.3.5 DELETE

Not applicable.

7.9 Resource: subscriptions

7.9.1 Description

This resource contains various resources related to subscriptions for notifications.

7.9.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/subscriptions**

This resource shall support the resource URI variables defined in table 7.9.2-1.

Table 7.9.2-1: Resource URI variables for resource "subscriptions"

Name	Definition
apiRoot	See clause 7.2

7.9.3 Resource methods

7.9.3.1 GET

The GET method is used to request information about the subscriptions for this requestor. Upon success, the response contains an entity body with the list of links to the subscriptions that are present for the requestor.

This method shall support the URI query parameters, request and response data structures and response codes, as specified in tables 7.9.3.1-1 and 7.9.3.1-2.

Table 7.9.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
subscription_type	String	0..1	Query parameter to filter on a specific subscription type. Permitted values: prov_chg_uu_uni: provisioning information change for V2X communication over Uu unicast prov_chg_uu_mbms: provisioning information change for V2X communication over Uu MBMS prov_chg_pc5: provisioning information change for V2X communication over PC5 v2x_msg: V2X interoperability message pred_qos: information on the predicted QoS

Table 7.9.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	Data type	Cardinality	Response Codes	Remarks
	SubscriptionLinkList	1	200 OK	Upon success, a response body containing the list of links to requestor's subscriptions is returned.
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.9.3.2 PUT

Not applicable.

7.9.3.3 PATCH

Not applicable.

7.9.3.4 POST

The POST method is used to create a new subscription to VIS notifications. Upon success, the response contains an entity body describing the created subscription.

This method shall support the request and response data structures and response codes, as specified in table 7.9.3.4-1.

Table 7.9.3.4-1: Data structures supported by the POST request/response on this resource

	Data type	Cardinality	Response Codes	Remarks
Request body	{NotificationSubscription}	1		The entity body in the request contains the data type of the specific VIS event subscription that is to be created, where the data type options are listed below and defined in clause 6.3: • ProvChgUuUniSubscription • ProvChgUuMbmsSubscription • ProvChgPc5Subscription • V2xMsgSubscription • PredQosSubscription
Response body	{NotificationSubscription}	1	201 Created	Indicates successful resource creation, where the resource URI shall be returned in the HTTP Location header field. In the returned NotificationSubscription structure, the created subscription is described using the appropriate data type from the list below and as defined in clause 6.3: • ProvChgUuUniSubscription • ProvChgUuMbmsSubscription • ProvChgPc5Subscription • V2xMsgSubscription • PredQosSubscription
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	415 Unsupported Media Type	It is used to indicate that the server or the client does not support the content type of the entity body. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

	ProblemDetails	0..1	422 Unprocessable Entity	It is used to indicate that the server understands the content type of the request entity and that the syntax of the request entity is correct but that the server is unable to process the contained instructions. This error condition can occur if an JSON request body is syntactically correct but semantically incorrect, for example if the target area for the request is considered too large. This error condition can also occur if the capabilities required by the request are not supported. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.9.3.5 DELETE

Not applicable.

7.10 Resource: existing subscription

7.10.1 Description

This resource represents a subscription that the client has created to receive VIS event notifications.

7.10.2 Resource definition

Resource URI: **{apiRoot}/vis/v2/subscriptions/{subscriptionId}**

This resource shall support the resource URI variables defined in table 7.10.2-1.

Table 7.10.2-1: Resource URI variables for resource "existing subscription"

Name	Definition
apiRoot	See clause 7.2.
subscriptionId	Refers to created subscription, where the VIS API allocates a unique resource name for this subscription. The resource name can be also used to identify the resource.

7.10.3 Resource methods

7.10.3.1 GET

The GET method is used to retrieve information about this subscription. Upon success, the response contains an entity body with the data type describing the subscription.

This method shall support the URI query parameters, request and response data structures, and response codes, as specified in tables 7.10.3.1-1 and 7.10.3.1-2.

Table 7.10.3.1-1: URI query parameters supported by the GET method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.10.3.1-2: Data structures supported by the GET request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	{NotificationSubscription}	1	200 OK	Upon success, a response body containing the data type describing the specific RNI event subscription is returned. The allowed data types for subscriptions are defined in clause 6.3 and are as follows: • ProvChgUuUniSubscription • ProvChgUuMbmsSubscription • ProvChgPc5Subscription • V2xMsgSubscription • PredQosSubscription
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.10.3.2 PUT

The PUT method is used to update the existing subscription. PUT method in this case has "replace" semantics. Upon successful operation, the target resource is updated with a new data type received within the message body of the PUT request.

This method shall support the URI query parameters, request and response data structures and response codes, as specified in tables 7.10.3.2-1 and 7.10.3.2-2.

Table 7.10.3.2-1: URI query parameters supported by the PUT method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.10.3.2-2: Data structures supported by the PUT request/response on this resource

	Data type	Cardinality	Response Codes	Remarks
Request body	{NotificationSubscription}	1		New NotificationSubscription is included as entity body of the request. The allowed data types for subscriptions are defined in clause 6.3 and are as follows: • ProvChgUuUniSubscription • ProvChgUuMbmsSubscription • ProvChgPc5Subscription • V2xMsgSubscription • PredQosSubscription
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	{NotificationSubscription}	1	200 OK	Upon success, a response body containing the data type describing the updated subscription is returned. The allowed data types for subscriptions are defined in clause 6.3 and are as follows: • ProvChgUuUniSubscription • ProvChgUuMbmsSubscription • ProvChgPc5Subscription • V2xMsgSubscription • PredQosSubscription
	ProblemDetails	0..1	400 Bad Request	It is used to indicate that incorrect parameters were passed to the request. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

	Data type	Cardinality	Response Codes	Remarks
	ProblemDetails	0..1	406 Not Acceptable	It is used to indicate that the server cannot provide any of the content formats supported by the client. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	412 Precondition Failed	It is used when a condition has failed during conditional requests, e.g. when using ETags to avoid write conflicts when using PUT. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	422 Unprocessable Entity	It is used to indicate that the server understands the content type of the request entity and that the syntax of the request entity is correct but that the server is unable to process the contained instructions. This error condition can occur if an JSON request body is syntactically correct but semantically incorrect, for example if the target area for the request is considered too large. This error condition can also occur if the capabilities required by the request are not supported. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.

7.10.3.3 PATCH

Not applicable.

7.10.3.4 POST

Not applicable.

7.10.3.5 DELETE

The DELETE method is used to cancel the existing subscription. Cancellation can be made by deleting the resource that represents the existing subscription.

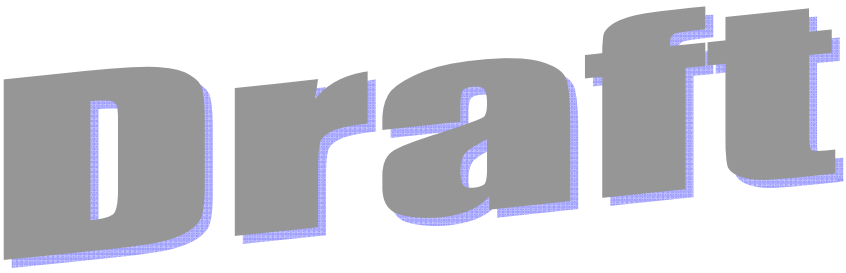
This method shall support the URI query parameters, request and response data structures and response codes, as specified in tables 7.10.3.5-1 and 7.10.3.5-2.

Table 7.10.3.5-1: URI query parameters supported by the DELETE method on this resource

Name	Data type	Cardinality	Remarks
n/a			

Table 7.10.3.5-2: Data structures supported by the DELETE request/response on this resource

Request body	Data type	Cardinality	Remarks	
	n/a			
Response body	**Data type**	**Cardinality**	**Response Codes**	**Remarks**
	n/a		204 No Content	Upon success, a response 204 No Content without any response body is returned.
	ProblemDetails	0..1	401 Unauthorized	It is used when the client did not submit credentials. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	1	403 Forbidden	The operation is not allowed given the current status of the resource. More information shall be provided in the "detail" attribute of the "ProblemDetails" structure.
	ProblemDetails	0..1	404 Not Found	It is used when a client provided a URI that cannot be mapped to a valid resource URI. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.
	ProblemDetails	0..1	429 Too Many Requests	It is used when a rate limiter has triggered. In the returned ProblemDetails structure, the "detail" attribute should convey more information about the error.



Annex A (informative): Mapping of permissions for RESTful API and topic based alternative transport

A.1 Overview

This annex provides mappings of permissions for topics between RESTful API and topic based alternative transport. ETSI GS MEC 009 [9] describes how permissions for topics between RESTful API and alternative transport can be mapped with each other. This annex uses the template for permissions mapping as defined in that group specification.

A.2 Mapping of permissions - RESTful and topic based alternative transport

Table A.2-1 lists the permission categories for each topic currently included in VIS API specification.

Table A.2-1: Definition of permissions for VIS

Permission identifier	Display name	Remarks
uu_unicast_provisioning_info	Uu Unicast Provisioning Info	Query
uu_mbms_provisioning_info	Uu MBMS Provisioning Info	Query
pc5_provisioning_info	PC5 Provisioning Info	Query
provide_v2x_msg_distribution_server_info	Provide V2X Message Distribution Server information	Task
provide_predicted_qos	Provide Predicted QoS	Task
publish_v2x_message	Publish V2X Message	Task
prov_chg_uu_uni	provisioning information change for V2X communication over Uu unicast	Subscribe-Notify
prov_chg_uu_mbms	provisioning information change for V2X communication over Uu MBMS	Subscribe-Notify
prov_chg_pc5	provisioning information change for V2X communication over PC5	Subscribe-Notify
v2x_msg	V2X interoperability message	Subscribe-Notify
pred_qos	information on the predicted QoS	Subscribe-Notify

Table A.2-2 describes how permission identifiers can be mapped to resources in the VIS RESTful API as defined in the present document.

Table A.2-2: Permission identifiers mapping for transport "REST"

Permission identifier	Specification
uu_unicast_provisioning_info	Resource: .../vis/v2/queries/uu_unicast_provisioning_info
uu_mbms_provisioning_info	Resource: .../vis/v2/queries/uu_mbms_provisioning_info
pc5_provisioning_info	Resource: .../vis/v2/queries/pc5_provisioning_info
provide_v2x_msg_distribution_server_info	Resource: .../vis/v2/provide_v2x_msg_distribution_server_info
provide_predicted_qos	Resource: .../vis/v2/provide_predicted_qos
publish_v2x_message	Resource: .../vis/v2/publish_v2x_message
prov_chg_uu_uni	Resource: .../vis/v2/subscriptions/prov_chg_uu_uni
prov_chg_uu_mbms	Resource: .../vis/v2/subscriptions/prov_chg_uu_mbms
prov_chg_pc5	Resource: .../vis/v2/subscriptions/prov_chg_pc5
v2x_msg	Resource: .../vis/v2/subscriptions/v2x_msg
pred_qos	Resource: .../vis/v2/subscriptions/pred_qos

Table A.2-3 describes how the permission identifiers can be mapped to topics offered over topic-based message bus.

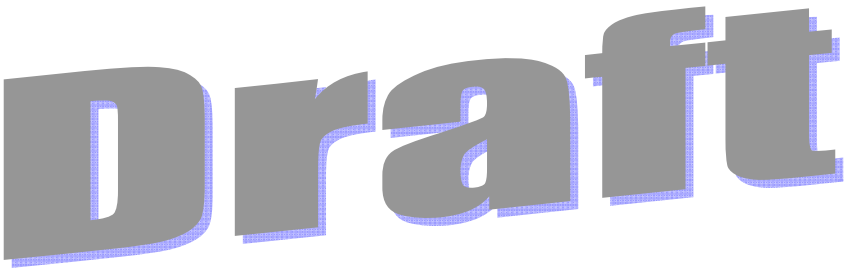
Table A.2-3: Permission identifiers mapping for transport "Topic-based message bus"

Permission identifier	Specification
uu_unicast_provisioning_info	Topic: /vis/enb/uu_unicast_provisioning_info
uu_mbms_provisioning_info	Topic: /vis/enb/uu_mbms_provisioning_info
pc5_provisioning_info	Topic: /vis/enb/pc5_provisioning_info
provide_v2x_msg_distribution_server_info	Topic: /vis/ue/provide_v2x_msg_distribution_server_info
provide_predicted_qos	Topic: /vis/ue/provide_predicted_qos
publish_v2x_message	Topic: /vis/ue/publish_v2x_message
prov_chg_uu_uni	Topic: /vis/ue/chg/uu_uni
prov_chg_uu_mbms	Topic: /vis/ue/chg/uu_mbms
prov_chg_pc5	Topic: /vis/ue/chg/pc5
v2x_msg	Topic: /vis/ue/v2x_msg
pred_qos	Topic: /vis/ue/pred_qos

Annex B (informative): Complementary material for API utilization

To complement the definitions for each method and resource defined in the interface clauses of the present document, ETSI MEC ISG is providing for the VIS API a supplementary description file compliant to the OpenAPI Specification [i.3].

In case of discrepancies between the supplementary description file and the related data structure definitions in the present document, the data structure definitions take precedence.



Annex C (informative): Radio access network scenario options for V2X communication

There are several radio access network options for V2X communications.

Scenario option 1: Only Operator A has eNBs in a specific area. Operator A's eNBs are shared with Operator B for all services including V2X.

Scenario option 2: Only Operator A owns the dedicated V2X spectrum in a specific area. Operator A's eNBs are shared with Operator B only for V2X service.

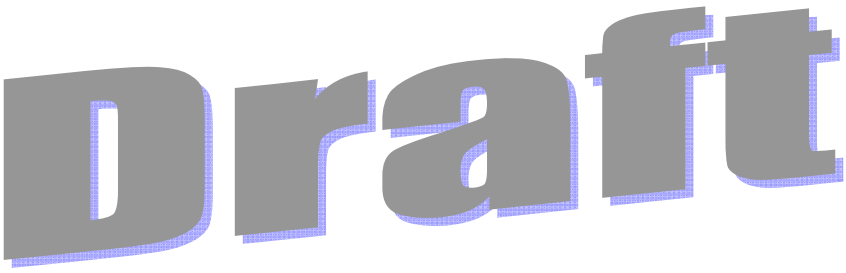
Scenario option 3: Both Operators A and B have eNBs in a specific area. V2X server distributes the V2X message to both operators' network.

All three options are valid, where option 1 requires least coordination among operators and option 3 may require coordination among operators.

Option 1 applies to the cases where one operator has no coverage in a specific area and has to use another operator's network through RAN sharing.

Option 2 is a potential choice when multiple operators are selected to provide V2X communication, where each operator provides V2X service in a certain area. Inter-working may be required at the "V2X service border" of two operators.

Option 3 maintains the separation of two operators. All levels (RAN, CN, Application) coordination may be required based on different use cases. It requires most effort, especially in deployment.



Annex D (informative): State-of-the-art of using a Message Broker as V2X Message Distribution Server for exchanging non-session based V2X messages

One important achievement of the MEC solution is to allow low latency communications and, consequently, to cover use cases that, in the past, were only possible with direct communication among vehicles. For this reason, there is the need that MEC Apps can interact with UEs to exchange V2X messages.

Several V2X services are non-session based and a possible option to exchange the V2X messages related to these services over the Uu interface is to use a V2X Message Distribution Server as introduced in Annex D of [10]. A common implementation solution is to employ a Message Broker.

This method has been borrowed by the IoT world considering the vehicle as an IoT object, or, better, as a container of different objects that can send and consume data. In the last years, several experimentations have been performed and different application layer protocols for the implementation of the Message Brokers have been considered.

An example of this approach has been provided by the Autopilot project [i.18]. All the Pilot sites of this large scale H2020 project collected data on the cloud through the MQTT [i.19] protocol. Similar experimentation with a Message Broker located at the edge for the exchange of V2X messages can be found in 5GCroCo [i.20] (MQTT), 5G-Carmen [i.21] (AMQP 1.0 [i.22]) and 5G-META [i.23] (MQTT or AMQP 1.0) projects.

Although the objectives and the realized use cases of these projects are different, the proposed approach is the same as evidence of its goodness. A Message Broker can make possible to exchange non-session based V2X messages among vehicles, infrastructure (e.g. RSUs or fixed sensors) and other relevant sources.

The CROADS Platform [i.24] also performs different on-field test using the AMQP 1.0 protocol and dealing with interoperability among different Message Brokers, that may also belong to different countries. One example is provided by the tests on the Brenner pass between Italy and Austria that are underway by CROADS and by the H2020 ICT4CART project [i.25] that uses a similar communication architecture. The CROADS project has also detailed a set of filter criteria for the reception of only relevant V2X messages. In the specific, they introduced a criterion to identify a geographical area of relevance exploiting a geospatial indexing based on quadkeys [i.26].

History

Document history		
V3.0.1	March 2021	First Early Draft version based on ETSI GS MEC 030 v2.1.1. Change of scope per the new WI description.
V3.0.2	March 2021	Correcting the scope per contribution MEC(21)000135 accepted during MEC#197-Tech call.
V3.0.3	April 2021	Incorporated agreed paper: MEC(21)000112r1 - MEC030 – adding missing referenced structured data type LinkType, agreed at MEC#198-Tech call. Editorial corrections by the Rapporteur.
V3.0.4	July 2021	Incorporated agreed papers: MEC(21)000252 - MEC030 – fixing error per ETSI TTF T012 feedback, agreed at MEC-Tech F2F#7 meeting. MEC(21)000226r2 - MEC030 – Interaction between the VIS and non-session based V2X services, agreed at MEC#212-Tech call. MEC(21)000335r1 - MEC030 – Provision of information from the VIS to MEC Apps about V2X Message Distribution Server, agreed at MEC#214-Tech call. Editorial corrections by the Rapporteur.
V3.0.5	November 2021	Incorporated agreed papers: MEC(21)000270r6 - MEC030 – extending the domain of in-advance predicted QoS notifications, agreed at MEC#225-Tech call. MEC(21)000516r1 - MEC030 – Align with MEC009 & MEC011 security considerations (for ETSI GS MEC 030 v3.1.1), agreed at MEC#225-Tech call. MEC(21)000533r3 - MEC030 – Implementing the WebSocket pattern to VIS subscription data types (towards v3.1.1), agreed at MEC#226-Tech call. MEC(21)000538 - MEC030 – API version update to v2 (towards v3.1.1 of GS), agreed at MEC#226-Tech call. Editorial corrections by the Rapporteur.
V3.0.6	December 2021	Incorporated agreed paper: MEC(21)000601 - MEC030 – Aligning with ETSI GS MEC 001 terms & abbreviations (towards v3.1.1 of GS), agreed at MEC#28 meeting. Editorial corrections by the Rapporteur.
V3.0.7	March 2022	Incorporated agreed papers: MEC(22)000045 - MEC030 – Updating notes on the informative reference to OpenAPI™ Specification (towards v3.1.1), agreed at MEC#232-Tech call. MEC(22)000065r1 - MEC030 – Annex A.2 updates - aligning with API definitions (towards v3.1.1), agreed at MEC#234-Tech call.
V3.0.8	July 2022	Clean-up done by ***editHelp!*** E-mail: mailto:edithelp@etsi.org

V3.0.9	September 2022	Incorporated agreed paper: MEC(22)000289r3 - MEC030 – 5GAA Input to ETSI GS MEC 030 on supporting end-to-end QoS prediction, agreed at MEC#258-Tech call. Editorial corrections by the Rapporteur.
V3.0.10	September 2022	Incorporated agreed papers: MEC(22)000438 - MEC030 – Annex A.2 updates - aligning with API definitions, agreed at MEC#31 meeting. MEC(22)000439 - MEC030 - Correcting introductory description of PredQosSubscription, agreed at MEC#31 meeting. MEC(22)000440 - MEC030 - Editorial fixes on top of GS MEC 030 v3.0.9 based on RC comments for GS MEC 030 v2.2.1, agreed at MEC#31 meeting. Editorial corrections by the Rapporteur.
V3.0.11	October 2022	Clean-up done by ***editHelp!*** E-mail: mailto:edithelp@etsi.org
V3.0.12	October 2022	Incorporated agreed paper: MEC(22)000497 - MEC030 – prepare GS for publication by removing open ENs, agreed at MEC#262-Tech call. Editorial corrections by the Rapporteur.
V3.0.13	December 2022	As announced at MEC#32, MEC030 is ready to move to Final draft. Final draft V3.0.13 is similar to Stable draft V3.0.12, and ready to go for MEC approval process.
V3.0.14	January 2023	Final draft V3.0.14 is similar to Final draft V3.0.13 as there were no comments duing the first RC for review. Thi draft is ready to go to the second RC for approval before ETSI publication.