Presented by: Miltiadis (Miltos) Filippou
Intel
MEC 030 WI Rapporteur

For: everyone

Episode #8 – V2X Information Service
In this episode ...

- We will learn:
  - Reference scenarios for the VIS service
  - Functionalities of the VIS service
  - The resource structure of the VIS API
  - How the VIS API can be used to provide journey-specific QoS predictions to a VIS consumer (e.g., a vehicular UE)
Reference scenarios of the VIS service

The figure shows all the scenarios applicable to V2X services. In particular:

- Some V2X services can be managed by original Equipment Manufacturers - OEMs (the so called "Vehicle OEMs scenario"), and, thus, it is reasonable to consider both single and multi-operator scenarios for such services.
- Similarly, the same applies when the “Intelligent Transport Systems - ITS Operator scenario” is considered, that may additionally provide services for different vehicle OEMs.
- In all cases, V2X services are expected to be provided by different network operators in the same country and/or in different countries.

<table>
<thead>
<tr>
<th>Scenario 1</th>
<th>Scenario 2</th>
<th>Scenario 3</th>
<th>Scenario 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle OEM scenario, single MNO</td>
<td>ITS operator scenario, single MNO</td>
<td>ITS operator scenario, single OEM, single MNO</td>
<td>ITS operator scenario, multiple OEMs, multiple MNOs</td>
</tr>
<tr>
<td>Vehicle OEM scenario, multiple MNOs</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Multi-operator scenarios and V2X services

The left hand side of the figure 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 the figure, 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.
MEC V2X Information Service & its main functionalities

- 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 (V2X Information Service API) is aimed at helping the ecosystem adopt MEC for automotive use cases.

- The MEC VIS includes the following functionalities:

  - 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).
  - Exposure of this information to MEC apps (also potentially belonging to different MEC systems).
  - Enablement of MEC apps to communicate securely with the V2X-related 3GPP core network logical functions (e.g. V2X control function).
  - Enablement of MEC apps in different MEC systems to communicate securely with each other.
  - Possibly gathering and processing information available in other MEC APIs (e.g. RNI API, Location API, WLAN API etc.) in order to predict radio network congestion and provide suitable notifications to the UE.
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.
# Resource structure of the V2X Information Service API

**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 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. |
Example: producing journey-specific 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.

The VIS may expose relevant (i.e. journey-specific) information about the QoS prediction to authorized UEs.

As part of the request, the VIS consumer (e.g., a vehicular UE) provides information about its potential routes.

Then, by reaching out to a Prediction Function (PF, out of scope of specification), the VIS obtains QoS predictions relevant to the planned routes and sends them to the VIS consumer.

Flow of a V2X application requesting the predicted QoS of a UE with potential routes

Attributes of the PredictedQos data structure

<table>
<thead>
<tr>
<th>Name</th>
<th>Data type</th>
<th>Cardinality</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>timeGranularity</td>
<td>TimeStamp</td>
<td>0..1</td>
<td>Time granularity of visiting a location</td>
</tr>
<tr>
<td>locationGranularity</td>
<td>String</td>
<td>1</td>
<td>Granularity of visited location. Measured in meters.</td>
</tr>
<tr>
<td>routes</td>
<td>Structure (inlined)</td>
<td>1..N</td>
<td>Information relating to the potential routes of a vehicular UE.</td>
</tr>
<tr>
<td>&gt;routeInfo</td>
<td>Structure (inlined)</td>
<td>2..N</td>
<td>Information relating to a specific route.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This first structure shall relate to the route origin and the last to the route destination. Intermediate waypoint locations may also be provided.</td>
</tr>
<tr>
<td>&gt;&gt;location</td>
<td>LocationInfo</td>
<td>1</td>
<td>Vehicular UE location</td>
</tr>
<tr>
<td>&gt;&gt;time</td>
<td>TimeStamp</td>
<td>0..1</td>
<td>Estimated time at the location</td>
</tr>
<tr>
<td>&gt;&gt;rsrp</td>
<td>UInt8</td>
<td>0..1</td>
<td>Reference Signal Received Power as defined in ETSI TS 136 214 [16]. Shall only be included in the response.</td>
</tr>
<tr>
<td>&gt;&gt;rsrq</td>
<td>UInt8</td>
<td>0..1</td>
<td>Reference Signal Received Quality as defined in ETSI TS 136 214 [16]. Shall only be included in the response.</td>
</tr>
</tbody>
</table>

NOTE: The data type of locationGranularity is a string which indicates the granularity of a visited location by means of latitudinal and longitudinal margins.

© ETSI 2022 – All rights reserved
Conclusions and further resources

What we have learnt:

• Reference scenarios for which the VIS service can be proven useful
• Main functionalities of the MEC VIS service and its API
• Resource structure of the VIS API
• How to request for and obtain journey-specific QoS predictions by using the VIS API

• Interested to learn more?

• Visit the MEC Sandbox and play with V2X Information Services: https://try-mec.etsi.org/
• Learn more about V2X Information Service API in MEC030 at https://www.etsi.org/deliver/etsi_gs/MEC/001_099/030/02.02.01_60/gs_mec030v020201p.pdf
• Also look at V2X Information Service API in MEC030 at https://forge.etsi.org/rep/mec/gs030-vis-api
• Follow also the next episodes of the MEC TECH Series 😊