CAMP LLC

Vehicle-to-Infrastructure 2 (V2I-2) Consortium

# Vehicle-to-Infrastructure Program

# V2I Safety Applications

# Event Driven Configurable Messaging (EDCM) Phase - I

*Submitted to the United States Department of Transportation*
*Federal Highway Administration (FHWA)*

*August 28, 2020*

*In Response to Cooperative Agreement Number*
*DTFH6114H00002*

# Executive Summary

This document describes the development of Event-Driven Configurable Messaging (EDCM) Project. During the previous Advanced Messaging Concept Development (AMCD) Project, it was determined that, as a means of improving roadway operations and communications with Transportation Management Centers (TMCs), a flexible message structure could effectively provide a mechanism in which infrastructure applications could request data from connected vehicles (CV). It was also concluded that a dynamically reconfigurable messaging scheme could be developed based on the needs of infrastructure applications.

The EDCM Project was conducted by the Crash Avoidance Metrics Partners LLC (CAMP) Vehicle-to-Infrastructure 2 (V2I-2) Consortium, consisting of Ford Motor Company, General Motors LLC, Hyundai Motor Group and Toyota, in cooperation with the Virginia Tech Transportation Institute (VTTI). The Project was sponsored by the Federal Highway Administration (FHWA) through Cooperative Agreement DTFH6114H00002.

The EDCM Project sought to do the following:

1) Develop and verify a real-world vehicle-to-infrastructure (V2I) communication environment supporting infrastructure-focused traffic management applications based on a flexible messaging structure
2) Demonstrate the flexibility and utility of a reconfigurable V2I data exchange
3) Develop, implement, and evaluate TMC applications
4) Employ an information flow for the transmission of dynamic information to connected vehicles
5) Engage with standards organizations toward message structure and data element standardization

The report herein details the technical aspects of the Phase I EDCM Project.

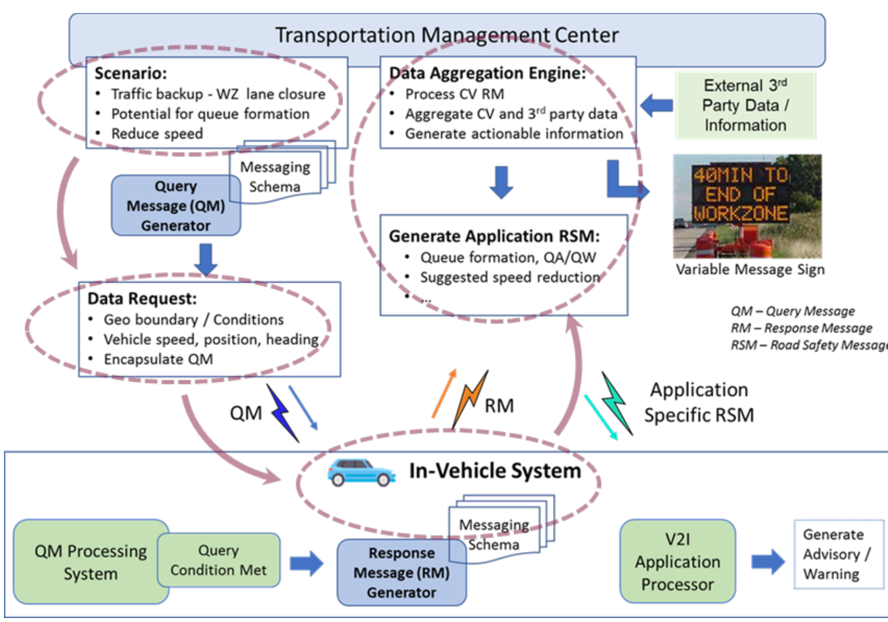
The EDCM system operates within the larger CV environment, which includes supporting communication infrastructure, security protocols and privacy management techniques required for EDCM to function. It enables a TMC to request information from CVs equipped with EDCM capabilities in specified areas regarding current conditions at varying rates and time of day. EDCM-equipped CVs then provide vehicle dynamics and status data in response when, where, and as often as requested by the TMC using a flexible messaging schema.

Through a series of outreach and focus group interactions, stakeholder engagement defined a set of potential use-case scenarios to ensure an EDCM system met the needs of end users. While the EDCM Project focused on applications relating to queues and work zones, these stakeholder interactions documented how EDCM enables numerous use cases without requiring changes to the underlying messaging strategy. The team also confirmed the desire

of Infrastructure Owner Operators (IOOs) to directly influence the content and volume of data acquired from vehicles. These IOOs emphasized the importance of tailoring applications according to localized needs across both the operational and strategic applications. Regarding operational applications, which are used in near real-time within the TMC, IOOs cautioned the team to avoid pitfalls related to overwhelming operators with unnecessary data and instead requested that information provided by EDCM systems be directly actionable.

Building on this knowledge foundation, the team developed a Concept of Operations (ConOps) to describe how the EDCM capabilities may achieve the objectives of a flexible V2I messaging strategy. Readers, to gain familiarity with EDCM are encouraged to review the ConOps, a standalone document, prior to reading the body of this report. The information contained within this EDCM Phase I Final Report focuses on implementation of the system described within the ConOps including a focus on Queue and Work Zone use cases.

To inform implementation, the Technical Team next designed and developed a system architecture for the EDCM. The system provides a flexible messaging approach between connected vehicles and TMC(s) by leveraging connected vehicle infrastructure and communications to identify events and road conditions that potentially impede the safe mobility of our nations traveling public. The EDCM system includes this messaging approach and supporting software for: (1) exchanging information between the TMC and EDCM-enabled vehicles, (2) aggregating information with other data sources, (3) processing these data, (4) summarizing complex data through actionable information which describes the current roadway status and supports decision and response by the IOO (Figure 1).




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium and Virginia Tech Transportation Institute (VTTI)*

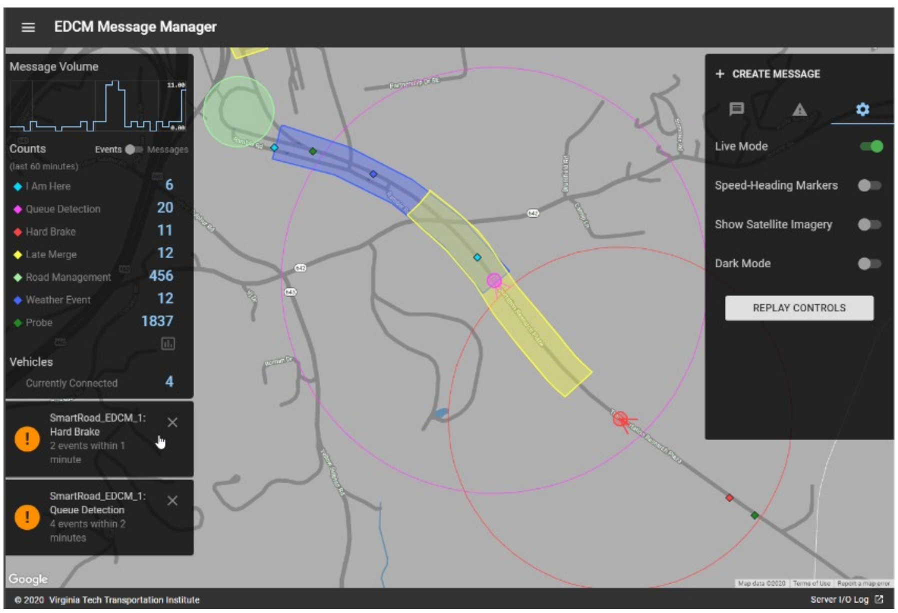
**Figure 1: EDCM System Architecture**

At the core of EDCM is a message strategy that permits the IOO to request data from connected vehicles through the following key processes:

- Wireless communications to establish two-way connectivity between IOOs and CVs. The technology is not explicitly specified within this project; however, a system verification was performed with traditional cellular data communications using a Transport Control Protocol. The architecture was also developed such that it should be readily supported by Dedicated Short Range Communications (DSRC) and Cellular Vehicle-to-Everything (C-V2X) technologies without substantial alternations.
- Query Messages (QM) generated by the TMC to request desired information from connected vehicles. Message formulation includes a detailed list of requested data elements and the specific conditions under which the data should be provided. These trigger conditions are associated with vehicle data, such as the current geolocation, vehicle dynamics, time of day, and other data available within a given vehicle implementation. Thus, data content and volume are affected by the QM content which are constructed according to the needs of the local IOO.
- Response Messages (RM) packaged by the vehicle and transmitted back to the TMC according to the request contained within the QM. A vehicle is not required to respond to all QMs but may respond based-on availability of the requested data, availably processing capacity, privacy settings, and other factors as appropriate to the vehicle implementation. This strategy supports the forward compatibility designed into the EDCM architecture.
- The RMs received by the TMC wherein the data is processed and aggregated with external information for generating actionable information. This information may be leveraged by the TMC to improve operations, such as detecting queue formation, obstructions in roadways, inappropriate work zone configurations, and many other use cases as described in the outreach section of this report. Because of the flexible messaging architecture, these applications need not be fully defined a priori and may evolve with time.
- Future application may cause an automatic generation of appropriate Road Safety Messages (RSMs) including transmission to CVs and other TMC subsystems (e.g., variable message sign system) to affect immediate changes on the roadway network. In addition, information obtained by the IOO via RMs may be used for more strategic applications, such as by traffic engineers to identify opportunities for improving roadway infrastructure (e.g., late merges, near-crash hot spots, icing propensity)

With considerations for connectivity, security, and privacy, the team developed the core message structure for the EDCM system architecture. Possibly the largest contribution of this project, this novel query and response messaging schema enables collaboration through a standardized common language. The eXtensible Markup Language (XML), a human- and machine-readable mechanism for describing hierarchical data, was selected for messaging because of its inherent flexibility and numerous toolsets across various computing platforms. Both the infrastructure and the vehicle use this scheme to encode and decode the message and subsequently act on the logic within. The EDCM scheme, including numerous examples for appropriate message structure for a given use case, are detailed within this document to serve as reference for future developers.

To ensure functionality of this novel EDCM approach, and to ensure readiness for future work, the team conducted an engineering verification at the project conclusion. This implementation developed the software and hardware needed to provide an end-to-end pipeline including a: (1) TMC web application for creating QMs with a geospatial graphical user interface, (2) a backend which automatically generated and packaged the XML message, (3) connectivity to establish communications with the vehicle, and (4) vehicles equipped with an emulated Onboard Equipment (OBU). These vehicles could then read QMs, setup multiple processes for listening to vehicle network variables simulations, generate RMs with the requested variables when the QM(s) conditions were satisfied, and transmit RMs to the aforementioned TMC interface (Figure 2) wherein data was processed displayed as actionable information.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 2: Main EDCM Message Manager Window with Alert Notifications**

Overall, the EDCM Project created and demonstrated the basic capability of a flexible messaging system design and schema to support use cases identified by stakeholders. The project also confirmed the ability to manage the V2I communications across TMCs. If implemented broadly, the EDCM concept should provide effective, actionable information to TMCs, specifically allowing customized vehicle data to be requested with a configuration based on user-defined vehicle conditions and/or states to provide timely and actionable information to the TMC operator.

# Table of Contents


**Executive Summary** ........ i

**1 Introduction** ........ 1

1.1 Organization of the Document ........ 3

**2 Stakeholder Engagement** ........ 4

2.1 Overview ........ 4

2.2 VDOT Workshops ........ 4

2.3 Key Outcomes ........ 6

2.3.1 Current Traffic Sensing Technology ........ 6

2.3.2 Potential Use Cases ........ 7

2.4 Outreach Summary ........ 10

**3 Event-Driven Configurable Messaging (EDCM) System Architecture** ........ 12

3.1 In-vehicle Architecture Example ........ 13

**4 Connectivity, Security, and Privacy** ........ 16

4.1 Connectivity Considerations ........ 16

4.1.1 Connection to TMC via Vehicle-to-OEM-Center ........ 16

4.1.2 Direct Vehicle-to-TMC Connection ........ 16

4.2 Implemented Connectivity Example ........ 16

4.3 Security Considerations ........ 17

4.4 Privacy Considerations ........ 17

**5 Flexible Messaging Schema** ........ 19

**6 Using XML to Formulate a Message** ........ 21

6.1 Query Message (QM) ........ 21

6.1.1 Query Message Frame: <qmFrame> ........ 22

6.1.2 Event Message: <eventMsg> ........ 22

6.1.3 Vehicle Data Request: <dataRequest> ........ 23

6.1.4 Query Message Duration: <qmDur> ........ 24

6.1.5 Query Message Action: <qmAction> ........ 24

6.1.6 Geo Fenced Regions: <gfRegion> ........ 25

6.1.7 Query Message Trigger: <qmTrigger> ........ 28

6.2 Response Message (RM) ........ 28
6.2.1 Response Message Frame: <rmFrame> ........ 29
6.2.2 Event Message: <eventMsg> ........ 29
6.2.3 Vehicle Parameters: <vehVars> ........ 29
6.2.4 Vehicle Position: <vehPos> ........ 30
6.2.5 Vehicle Acceleration Status: <vehAccelStatus> ........ 30
6.2.6 Vehicle Brake Status: <vehBrakeStatus> ........ 31
6.2.7 Vehicle Exterior Light Status: <extLightStatus> ........ 31
6.2.8 Vehicle Geo Fenced Status: <gfRegionEntryExitStatus > ........ 32
6.3 Examples of QM and RM ........ 32
6.3.1 Traffic Congestion – Potential for Queue Formation ........ 32
6.3.2 Inclement Weather– Slippery Road Condition ........ 33
6.3.3 Roadway & Road Surface Management - Deficient Roadway Section ........ 35
6.3.4 EDCM Vehicle Geofence Entry/Exit Status ........ 37
6.3.5 Query Message Interpretation and Response Criteria ........ 38
**7 Engineering Verification Implementation ........ 40**
7.1 Implementation ........ 41
7.1.1 Engineering Verification Architecture ........ 41
7.1.2 TOC Client Application ........ 42
7.2 Engineering Verification ........ 53
7.2.1 Approach ........ 53
7.2.2 Vehicles and Equipment ........ 54
7.2.3 Virginia Smart Road ........ 54
7.2.4 Test Protocol and Locations of Scenarios ........ 54
7.3 Results and Discussion ........ 56
7.4 Conclusions ........ 71
**8 Queue Advisory / Queue Warning (QA/QW) System ........ 72**
8.1 Traffic Congestion Categories ........ 72
8.2 Traffic Congestion Scenarios ........ 72
8.2.1 Congestion caused by a known incident or at a recurring location ........ 72
8.2.2 Congestion caused at a non-recurring location ........ 73
8.3 QA/QW Concept and System Design ........ 73

8.3.1 Use Cases ........ 74
8.3.2 Queue Formation, Queue Detection and Detection Improvement ....... 75
8.4 Queue Advisory / Queue Warning System Behavior ........ 76
8.4.1 Application Fidelity ........ 76
8.5 In-Vehicle QA/QW Application ........ 78
8.5.1 Estimating Inform and Warn Distances from BoQ ........ 79
8.5.2 Estimating Distance to BoQ ........ 80
8.6 In-vehicle QA/QW Application Data ........ 81
8.6.1 Road Safety Message (RSM) Data Elements ........ 81
**9 Summary ........ 85**
**10 References ........ 89**
**Appendix – A: Table of Supported Vehicle Data in QM ........ 91**

# List of Figures


Figure 1: EDCM System Architecture ..... ii
Figure 2: Main EDCM Message Manager Window with Alert Notifications ..... iv
Figure 3: VDOT District Office Regions ..... 6
Figure 4: EDCM System and Communication Framework ..... 12
Figure 5: EDCM System Architecture ..... 13
Figure 6: Functional Diagram of EDCM-related Components ..... 14
Figure 7: Session Security Needs for Individual or Group Vehicle Internet Protocol Addresses ..... 17
Figure 8: Example - CV Registering for Services with TMC ..... 18
Figure 9: A Four Node Polygon using <gfRegion> ..... 26
Figure 10: From Location to Location Using <gfRegion> ..... 27
Figure 11: Circular Region of Interest for Slippery Road Condition in Query Message ..... 34
Figure 12: Engineering Verification Server Architecture ..... 41
Figure 13: Engineering Verification OBE Emulator Architecture ..... 41
Figure 14: EDCM Message Manager Layout ..... 43
Figure 15: Create Message Bar Options ..... 44
Figure 16: QM Edit Window ..... 45
Figure 17: Example of Available QMs List ..... 45
Figure 18: Example of Raw Start XML Message Generated for QM ..... 46
Figure 19: QM Geofence Regions for Engineering Verification ..... 47
Figure 20: Drive Distance Geofence Definition ..... 48
Figure 21: Polygon Geofence Definition ..... 49
Figure 22: Defined Work Zones and Associated QM List ..... 50
Figure 23: Incoming RM Data Display ..... 51
Figure 24: Bar Graph of the RMs Received ..... 51
Figure 25: Main EDCM Message Manager Window with Alert Notifications ..... 52
Figure 26: QM Response Message Content ..... 53
Figure 27: Test Protocol and Locations of Events on the Virginia Smart Road for the Engineering Verification ..... 55
Figure 28: QM Edit Window ..... 57
Figure 29: Hard Brake QM XML ..... 58
Figure 30: Queue Detection Geofence Region QM XML Definition ..... 58
Figure 31: Late Merge QM XML ..... 59
Figure 32: Weather Event QM XML ..... 60
Figure 33: Weather Event QM XML for Distance Interval Reporting ..... 61
Figure 34: Road Management QM XML ..... 61
Figure 35: Probe QM XML ..... 62
Figure 36: Vehicle RM Status from Server Log ..... 63
Figure 37: Road Management Message Response Example ..... 63
Figure 38 a, b: Query Response for Road Management Message ..... 64
Figure 39: Server Log Example Showing Responses from Five Vehicles ..... 65
Figure 40: Time Sequence of Late Merge Maneuver ..... 67
Figure 41: EDCM Message Manager with Live Mode Display On ..... 68

Figure 42: EDCM Message Manager with Live Mode Display Off (Persistent View).... 69
Figure 43: Speed-Heading Marker Display ........................................................ 70
Figure 44: Alert Response with Persistent Data Display ................................................ 71
Figure 45: QA/QW System Data Flow Diagram ............................................................ 74
Figure 46: Various Queue Formations.............................................................. 75
Figure 47: Typical Queue Detection.................................................................. 75
Figure 48: Improvement in Queue Detection Using CV DATA ..................................... 76
Figure 49: Flow Diagram for In-Vehicle QA/QW Application....................................... 79
Figure 50: Estimation for "Inform" and "Warn" Zones .................................................... 80
Figure 51: Estimating Host Vehicle Distance to BoQ ..................................................... 81

# List of Tables

Table 1: VDOT Stakeholder Engagement ........ 4
Table 2: <eventMsg> - Supported Elements / Attributes ........ 22
Table 3: <dataRequest> - Supported Elements / Attributes ........ 23
Table 4: <qmDur> - Supported Elements / Attributes ........ 24
Table 5: <qmAction> - Supported Elements / Attributes ........ 24
Table 6: <gfRegion> - Supported Geofence Regions ........ 25
Table 7: <qmTrigger> - Supported Elements / Attributes for QM Triggering ........ 28
Table 8: <vehVars> - Supported Elements / Attributes ........ 29
Table 9: <vehPos> - Supported Elements / Attributes ........ 30
Table 10: <vehAccelStatus> - Supported Elements / Attributes ........ 30
Table 11: <vehBrakeStatus> - Supported Elements / Attributes ........ 31
Table 12: <extLightStatus> - Supported Elements / Attributes ........ 31
Table 13: <gfRegionEntryExitStatus> - Supported Elements / Attributes ........ 32
Table 14. Engineering Verification Query Message IDs ........ 62
Table 15: Data Elements for In-Vehicle QA/QW Application ........ 83

# List of Acronyms

| Acronym | Definition |
|---|---|
| ABS | Anti-lock Brake System |
| AMCD | Advanced Messaging Concept Development |
| BMM | Basic Mobility Message |
| BoQ | Back of Queue |
| BSM | Basic Safety Message |
| CAMP | Crash Avoidance Metrics Partners LLC |
| CAN | Controller Area Network |
| CARMA | Cooperative Automation Research Mobility Application |
| ConOps | Concept of Operations |
| CV | Connected Vehicle |
| C-V2X | Cellular Vehicle-to-Everything |
| CV PFS | Connected Vehicle Pooled Fund Study |
| CWZ | Connected Work Zone |
| DSRC | Dedicated Short Range Communication |
| DVI | Driver Vehicle Interface |
| EDCM | Event-Driven Configurable Messaging |
| ETSI | European Telecommunications Standards Institute |
| FHWA | Federal Highway Administration |
| FoQ | Front of Queue |
| HCM | Highway Capacity Manual |
| HRD | Hampton Roads District |
| HRBT | Hampton Roads Bridge-Tunnel |
| HV | Host Vehicle |
| IEEE | Institute of Electrical and Electronics Engineers |
| INFLO | Intelligent Network Flow Optimization |
| IOO | Infrastructure Owner Operator |
| IP | Internet Protocol |
| MMMBT | Monitor-Merrimac Memorial Bridge-Tunnel |
| NoVA | Northern Virginia, Operations District in Virginia |
| OBE | On-Board Equipment |
| OBU | On-Board Unit |
| OEM | Original Equipment Manufacturer |
| PDM | Probe Data Message |
| PKI | Public Key Infrastructure |
| PoC | Proof of Concept |
| PoP | Period of Performance |

| Acronym | Definition |
|---|---|
| QA/QW | Queue Advisory / Queue Warning |
| QM | Query Message |
| RIC | Richmond District, Operations District in Virginia |
| RM | Response Message |
| RSM | Road Safety Message |
| RSU | Roadside Unit |
| RSZ | Reduced Speed Zone |
| SAE | SAE International |
| SAL | Salem District, Operations District in Virginia |
| SCMS | Security Credential Management System |
| SPaT | Signal Phase and Timing |
| SSP | Safety Service Patrol |
| TCP | Transport Control Protocol |
| TMC | Transportation Management Center |
| TOC | Transportation Operation Center |
| USB | Universal Serial Bus |
| USDOT | United States Department of Transportation |
| V2I | Vehicle-to-Infrastructure |
| V2I-2 | Vehicle-to-Infrastructure 2 Consortium |
| VCC | Virginia Connected Corridor |
| VDOT | Virginia Department of Transportation |
| VTTI | Virginia Tech Transportation Institute |
| WZM | Work Zone Management |
| XML | eXtensible Markup Language |
| XSD | eXtensible Markup Language Schema Definition |

# 1 Introduction

This document describes the development of an Event-Driven Configurable Messaging (EDCM) Project, a flexible messaging system with the ability to dynamically adjust two-way data exchange between connected vehicles and a Transportation Management Center (TMC). The EDCM Project was conducted by the Crash Avoidance Metrics Partners LLC (CAMP) Vehicle-to-Infrastructure 2 (V2I-2) Consortium, consisting of Ford Motor Company, General Motors LLC, Hyundai Motor Group and Toyota, in cooperation with the Virginia Tech Transportation Institute (VTTI). The Project was sponsored by the Federal Highway Administration (FHWA) through Cooperative Agreement DTFH6114H00002.

Previous USDOT-sponsored research under the Advanced Messaging Concept Development (AMCD) Project [1] evaluated "the ability of connected vehicles to generate, and infrastructure to collect, Basic Safety Message (BSM), Probe Data Message (PDM), and Basic Mobility Message (BMM) alternatives using both cellular and Dedicated Short Range Communication (DSRC), employing basic message control strategies in-real world driving conditions for non-safety-critical applications." Results from this research suggest that (1) a flexible message structure may provide an effective mechanism for infrastructure applications to request data from vehicles with the aim of improving roadway operations and (2) the development of a dynamically reconfigurable messaging scheme based on the needs of infrastructure applications utilizing data received from vehicles would provide additional value.

The high-level goals of the EDCM Project were to:

- Develop and deploy a vehicle-to-infrastructure (V2I) communication environment supporting infrastructure-focused traffic management applications based on a flexible messaging structure
- Demonstrate the flexibility and utility of reconfigurable V2I data exchange by showing:
    - Two-way data exchanges between vehicles and TMC operations under real-world conditions
    - Infrastructure applications that transform V2I data into actionable information
- Develop, implement, and evaluate exemplar TMC applications that:
    - Help address local roadway operational conditions and needs
    - Show the range of applications that may be supported by a flexible V2I messaging structure
- Implement a complete information flow that transmits dynamic information to the vehicle to enable relevant in-vehicle warnings based on vehicle and infrastructure data sources

- Engage with standards organizations for message structure and data element standardization

The technical Period of Performance (PoP) for development of the overall EDCM system was split into two phases. Phase 1 laid the technical foundation for the EDCM concept. Coordination, planning and outreach efforts to stakeholders were conducted with the aim of ensuring that the final product addresses a broad set of use cases for a wide variety of Infrastructure Owner Operators (IOOs) and automotive Original Equipment Manufacturers (OEMs). A system architecture was defined including the message structures, communications protocols and system interfaces required to implement the initial use cases. A bench-test reference system was created to ensure end-to-end functionality of flexible messaging, including communication mechanisms and functionality necessary to demonstrate the required flow and processing of messages to support the anticipated EDCM protocols.

Phase 2 was not pursued due to a reprioritization of federal research funds. In the Phase 2, the project plan was to:

- Continue expanding upon Phase 1 accomplishments by building the additional functionality required to support the general EDCM concept of operation and system requirements.
- At sufficient maturity, apply the resulting EDCM system combined with custom application features to support Queue Advisory / Queue Warning (QA / QW) and Connected Work Zone (CWZ) applications.
- Develop software to support the TMC functionality required for a TMC operator to manage a reference EDCM system with support for the QA/QW and CWZ use case scenarios.
- Build test vehicles that contain the equipment, interfaces, and functionality required to demonstrate the EDCM functionality through QA/QW and CWZ application scenarios.
- Demonstrate the EDCM Proof of Concept (PoC) system by conducting a live demonstration of example QA/QW and CWZ use case scenarios on the Virginia Smart Road.

In addition to this report, two additional items were pulled forward from Phase 2 to document the findings and progress achieved during Phase 1 of the research plan:

1. EDCM Concept of Operations (ConOps) [2] and supported use cases
2. EDCM Queue Advisory & Queue Warning (QA/QW) System and In-Vehicle Application Requirements - This document was developed in support of Connected Vehicle Pooled Fund Study (CV PFS) companion project in cooperation with the technical team lead of CV PFS Project.

## 1.1 Organization of the Document

The objectives of the Phase 1 Project were achieved through collaborative engineering activities documented in this report. Section 2 provides an overview of the conducted stakeholder workshops  and associated workshops with the participating transportation agencies, key outcomes and potential use cases motivating the development of an EDCM strategy. Using the information from prior projects and the stakeholder outreach, the EDCM system architecture and information flows, including an in-vehicle architecture, are provided in Section 3. Section 4 discusses connectivity between the TMC and CVs, including security and privacy considerations for the EDCM system. A flexible messaging schema, core of the EDCM strategy for implementing two-way information exchanges between the TMC and CVs is defined in Section 5, and Section 6 illustrates formulating query and response messages with detailed examples using the schema. In Section 7, the bench test reference implementation is described including example use cases to conduct an engineering verification of the end-to-end EDCM system and information flows. Section 8 provides a QA/QW system design for queue detection supporting in-vehicle alerts in a connected work zone. Section 9 summarizes the outcomes of EDCM Phase 1 research.

# 2 Stakeholder Engagement

## 2.1 Overview

The purpose of the stakeholder engagement was to gather information from end users to aid in the development of the EDCM system. In particular, there were five areas in which the team sought feedback to ensure that EDCM would support the needs of the user:

- The EDCM architecture and Concept of Operations which enables a TMC to request vehicle data and receive responses from vehicles
- Support for the QA/QW and CWZ applications using the EDCM framework
- Additional use cases of interest to a TMC
- An in-vehicle system architecture to support needs for selected applications

## 2.2 VDOT Workshops

The Virginia Department of Transportation (VDOT) is our primary partnering agency serving as a representative for the IOO community. VDOT agreed to collaborate with the EDCM Team throughout the duration of the project. Coordinated through VTTI, VDOT provided input into the system design elements and was planned to work directly with VTTI to develop and deploy the necessary infrastructure components to support EDCM in Phase 2.

Table 1 lists a series of workshops that were held with several VDOT districts to gather insights regarding the high-level EDCM concept architecture, the basic needs of the IOO users of EDCM, and to facilitate idealization and review of the additional generated use cases. The workshop planning and outcome materials including the goals, activities, schedules, materials, and raw notes were compiled for development of EDCM system.

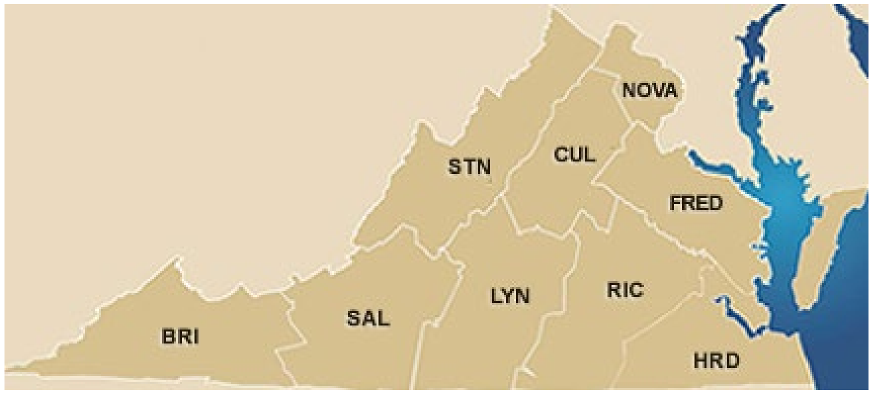
**Table 1: VDOT Stakeholder Engagement**

| District Name | Date | Duration |
|---|---|---|
| Salem | August 15, 2019 | 3 Hours |
| Northern Virginia (NoVA) | August 26, 2019 | 3 Hours |
| Richmond | August 29, 2019 | 4 Hours |
| Hampton | October 1, 2019 | 3 Hours |

The EDCM Team coordinated with VDOT to schedule a series of IOO workshops with four of the nine Operations Districts in Virginia (Figure 3). The regions selected are described below and have unique roadway management considerations due to population density, traffic density, road use distribution, and geographic features.

- Salem (SAL)[1]: This is largely a rural district, and it contains 86.6 miles of the 325 miles of I-81, second only to the Staunton District. As a whole, I-81 carries 42% of the statewide truck traffic volume. In some sections of I-81, trucks account for 20-30% of the traffic volume.
- Northern Virginia (NoVA)[2]: This district maintains more than 14,000 lane miles of roads and contains some of the heaviest traffic in the US. This was reflected in a list of the 25 worst traffic hotspots that was compiled by INRIX Roadway Analytics in 2017 in which the section of I-95 South that runs through this district from the 495 beltway to US Route 17 was ranked as the worst traffic hotspot in the nation with 1,394 traffic jams.
- Richmond (RIC): Encompassing the state capital of Virginia, this district covers one of the larger geographic areas and is comprised of 14 counties, eight cities and manages 18,000 miles of roads.
- Hampton Roads (HRD)[3]: In 2016, the TomTom Traffic Index ranked Hampton Road as the 31st worst traffic area in the country. This is largely due to the waterways which are central to the geography of the region. A network of bridges and tunnels exist including:
    - Monitor-Merrimac Memorial Bridge-Tunnel (MMMBT)[4]: This is a four-lane dual tunnel system.
    - Hampton Roads Bridge-Tunnel (HRBT)[4] carries nearly 3 million vehicles each month and over 100,000 vehicles a day during tourist season.
    - Downtown Tunnel[4]: This tunnel carries over three million vehicle each month.
    - Chesapeake Bay Bridge-Tunnel[4]: This 17.6 mile long bridge-tunnel complex is the largest in the world.

[1] Source: Virginia Department of Transportation 2017
[2] Source: https://www.nj.com/traffic/2017/10/these_are_the_25_worst_traffic_hotspots_in_america_four_are_here.html
[3] Source: https://wtkr.com/2016/04/08/tomtom-ranking-hampton-roads-has-31st-worst-traffic-congestion-in-the-country/
[4] Source: https://www.virginiadot.org/travel/hro-tunnel-default.asp




*Source: Virginia Department of Transportation (VDOT)*

**Figure 3: VDOT District Office Regions**

Each workshop was intended to (1) establish a dialogue with each operations district of VDOT, (2) socialize the EDCM concept and obtain VDOT's concept review, and (3) brainstorm use cases and capture operational priorities for the application of EDCM from the perspective of the IOO. At each workshop, outputs for the development team were gathered and the associated feedback integrated into the architecture and application development activities. The technical discussions provided valuable information regarding EDCM requirements and use case input from the perspective of the infrastructure.

## 2.3 Key Outcomes

The following summarizes the findings from the VDOT workshops. It first looks at the current tools that are in use by the Transportation Operation Center (TOC) to monitor and manage the roadways as this provides context as to the type and content of information used relative to the information that could be obtained from CVs. Next, a summary of the use case discussions is presented which concentrates first on the QA/QW and Work Zone Management (WZM) cases, the focus of this project, and then additional use cases identified during the workshops.

### 2.3.1 Current Traffic Sensing Technology

For each district, cameras are a primary tool used to gather information. As expected, different districts have varying quantities of cameras at their disposal. For the more rural district of Salem, cameras are primary limited to interstates and are positioned in key locations such as interchanges (e.g., I-81 and I-77) and mountain passes (e.g., I-77, Fancy Gap north of the Virginia and North Carolina border). The urban areas in Northern Virginia, Richmond and Hampton Roads districts utilizes cameras extensively on the interstate. Hampton Roads also has local operators at key waterway crossings with a primary task of monitoring traffic. In addition, external data sources are utilized such as on-line mapping and traffic services, road user reports, and observations from Safety Service Patrol (SSP) vehicle operators. Richmond commented that their sensor data is pulled at a lower frequency, so they often have information from other sources by the time the sensor data arrives at the TOC.

The current data landscape provides an opportunity for V2I communication to provide improvements in the type, quantity, quality, control, and timeliness of the data leveraged by the TOCs. Some of the potential benefits that the attendees noted during the workshops were the potential for:

- Faster incident identification
- Additional modes of communication
- Decreased communication time
- More and better information on what vehicles are doing on roadways

The primary benefits the attendees noted of this information was the TOC operators and engineering analysis. It was noted that the information must not increase the cognitive workload of the operators, but instead, provide precise and actionable information that helps them focus on what is most important. The operators should not be required to analyze the incoming raw data as this distracts from their primary job of maximizing safety and mobility of the network in real-time. For engineering analysis, the attendees noted aggregated data that would enable advanced analysis as a means to improve safety and efficiency of travel.

### 2.3.2 Potential Use Cases

During the workshops, the team led discussions on how the data made available through the EDCM could be used for QA/QW and WZM use cases and what additional use cases could be envisioned.

#### *2.3.2.1 Queue Advisory / Queue Warning*

One of the first items of interest is how the regions defined a queue. In general, it was broadly defined as a deviation from what is expected or normal. In addition, they identify queues as two types of manifestation, that is, reoccurring congestion or non-reoccurring congestion. Within the context of the general definition, typical slow traffic during the morning and evening commutes, is not defined as a queue in that it is not considered an event by the TOC. Consequently, identification of non-reoccurring queues are of particular interest as they pose a greater safety risk. However, how a queue event is quantified varied between regions and within the districts themselves. Some assessed a queue as a comparison between volume of vehicles versus roadway capacity while others based it on travel time between given points, yet others expressed it as a simple speed differential from typical. Given the unique conditions that exist in different regions which can cause a queue, final determination of when an incident should be created is a manual process within the TOCs and is not explicitly defined by VDOT.

In discussing how these data could be utilized, the attendees focused on the potential to more quickly identify the onset of nonrecurring queues based on quantitative data from the roadway. This would allow them to facilitate a more appropriate response by providing information to drivers of the location of unexpected delays, the impact on travel time and potentially providing alternate routing or dynamic speed harmonization if supporting interfaces existed (e.g., the dynamic roadway signs along I-66). In all the workshops, there was interest in being able to communicate this information directly to the driver in their vehicle rather than relying on the availability of rather costly dynamic roadway signage.

#### *2.3.2.2 Work Zones*

It was noted in one workshop that being able to monitor work zones is more important than simple QA/QW because it impacts the safety of more people (drivers and workers) and they have more tools available to affect traffic.

For WZM, three sub use cases were identified on how the data could be utilized. The first was to use the data to confirm and identify the location and configuration of work zones. One district noted that they are not notified of 25% of the work zones in their district. They see this data as another way to potentially identify unreported work zones. For work zones that they have been notified about, they see this data as a means to potentially identify and verify changes in the location and configuration of the work zones as work progresses throughout the duration of a project.

Related to this is the monitoring and management of the impact a work zone has on traffic. Specifically, participants noted that this could be particularly useful in the opening and closing of a work zone based on data coming back from vehicles in and around a work zone. For mobile work zones (e.g., striping) which are particularly difficult to monitor, they saw this as a means to identify traffic queues forming and pull or temporarily stop work to alleviate the congestion.

The final category that was identified for WZM was to evaluate the efficacy of the work zone management including hours of operation and alternate routings. Having historic data from vehicles traveling through and around work zones provides information that can inform existing projects as well as having the potential to help define the configuration and operation of future projects.

#### *2.3.2.3 Other*

In addition, after review of specific QA/QW and WZM use cases, participants were asked to brainstorm additional use cases and applications they could enable with information provided by an EDCM strategy. From these discussions, the team identified five categories of use cases each containing a selection of applications. This should not be viewed as all potential uses of EDCM, but rather an initial set of additional priority applications which may be worth investigating further.

##### 2.3.2.3.1 Traffic Monitoring

One benefit the participants saw to data provided through EDCM is as a tool to monitor the general state of the roadways network. With the ability to configure the information provided by the vehicles, participants saw a benefit to being able to target specific regions or roadway types and specific particular information to provide performance indicators useful for their specific regions and unique challenges.

In addition, they noted the benefit of being able to use EDCM for event detection. While QA/QW falls in this category, it also includes additional anomalous vehicle behavior that could indicate a traffic or roadway condition that should be further investigated.

Another potential EDCM use case focuses on a vehicle monitoring and driver advisory tool. Different regions focused on different aspects of this use case, but all saw the potential to improve safety if there was the ability to identify and communicate to a specific driver. Specific applications for this include identification of wrong way driving, curve speed warning, excessive speed and heavy vehicle over height or over length warning. While this use case may be enabled by carefully located dynamic signs, the vehicle warnings were preferred by the participants. While the current EDCM definition was not designed to target or message a specific vehicle on the roadway, these applications identified by the workshop participants are compelling and should be considered if privacy and other concerns associated with targeting a specific vehicle/driver for are sufficiently addressed.

#### 2.3.2.3.2 Crash and Near Crash Detection

The ability to have event driven messages based on kinematic and vehicle state data, such as longitudinal acceleration and airbag deployment, provides the opportunity for the TOC to receive timely information regarding safety critical events on the roadways. For the purposes of real-time management, participants stated this information allows the TOC to more quickly and efficiently dispatch appropriate responders and potentially provide information to other drivers. One district provided an example of where having this information could have allowed them to respond to a vehicle rollover that was unobserved and therefore unreported.

In addition to incident response, participants noted that these data could also be used for engineering analysis. Incidents such as hard braking, lane deviation, ABS or stability control activation, could be used to identify hot spots on the roadways that show systematic problems. One example that was given was a curve whose super elevation was inappropriately modified during road work resulting in vehicles departing the roadway. Having messages from vehicle triggered based on traction or stability control or high lateral acceleration or yaw rates, could have provided insight into the issue proactively. Similarly, participants noted that hard braking could be used to proactively identify problematic intersections that may need to be retimed or geometrically adjusted.

#### 2.3.2.3.3 Weather

Adverse road conditions due to weather is another use case of interest to VDOT. Two specific conditions of interest identified were icy roads and excessive water on the roadways. Given the size and varied terrain of some of the districts, weather can vary significantly across the regions. The ability to monitor temperature and vehicle state (e.g., wipers, traction and stability control) could provide localized information on areas that needed treatment to address ice or snow on the roadways. In the case of heavy rains, drivers could be notified of potentially hazardous conditions ahead through signage or potentially through paring within vehicle applications through the Road Safety Message (RSM).

#### 2.3.2.3.4 Road Hazard

Similar to the weather use case, using vehicle kinematic triggers could allow a TOC to identify potential road hazards and dispatch service vehicles. In particular, participants discussed the use of EDCM to identify repeated swerves or lane deviations (e.g., yaw, lateral acceleration, turn signal activation) at the same location as an indication of an object in the road. The inclusion of vertical acceleration was also noted as a means to identify potholes and bridge offsets, all of which could be readily evaluated by dispatching the appropriate personnel to the location associated with the crowd-sourced data.

#### 2.3.2.3.5 Active Traffic Management

Another use case discussed in the workshops was the ability to use the EDCM strategy to allow more targeted traffic management. Some regions operated by the districts in Virginia currently have the infrastructure to support variable speed limits. These participants observed that having the ability to query speed directly from the vehicles throughout their region would provide more opportunities for speed harmonization and other active traffic management strategies during heavy or disrupted traffic scenarios.

Workshop participants also commented on the potential to actively adjust signal phase and timing (SPaT) at intersections to improve safety and mobility. As with previous use cases, this could be used in real-time SPaT modification or in engineering analysis to adjust traffic signal configurations. An example was given for an intersection near a school where the turn lane would back up into the adjacent through lane and remain that way for multiple light cycles at the beginning and end of school. Setting up a query message centered on a specific intersection could provide data showing that the turn lane needed to hold longer during unusually high usage to clear adjacent lanes.

Some of the regions were also interested in the potential to provide alternate route information. Two specific examples that were provided were during long-term road construction projects to provide drivers with accurate information for alternate routes that were longer but achieved shorter travel times. An example of how this could be utilized was given during the meeting with the Hampton Roads district which manages the road used to access Virginia Beach and the Outer Banks. The most direct route to these destinations is not always the fastest route due to the congestion that can occur through the Hampton Roads Bridge-Tunnel. Having real-time data from vehicles on both routes would allow for more accurate and earlier notification to drivers as to the faster route, enabling active load balancing and increasing the efficiency of the roadway network.

## 2.4 Outreach Summary

The following summarizes some of the key outcomes from the workshops.

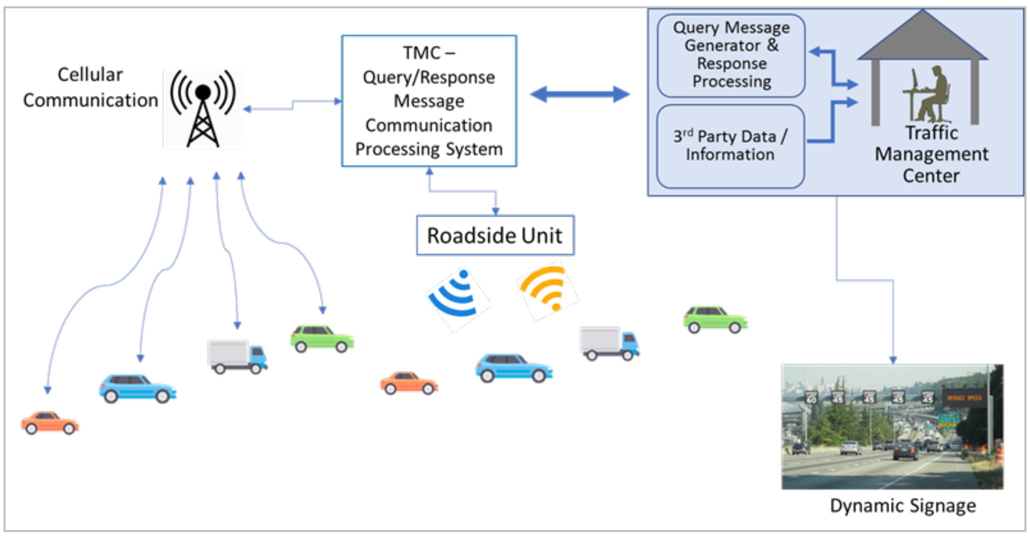
- There is a common set of EDCM use cases that all the districts identified. However, the priority and application of these different use cases varies depending on the unique roadways, geography, traffic distribution and population density patterns that exist within a district.

- These differences highlight the need for flexibility in the message structure to allow the individual TOCs to tailor the messages to extract the data that is most relevant and useful for their area.
- A theme from every workshop was the need to ensure the information coming back to the TOC operators would be actionable and reduce workload. The participants noted that increase workload could result if the operator was required to analyze and interpret the data. The expectation expressed was that the query messages and data aggregation, analysis and presentation could be predefined by TOC managers and engineers in order to provide only relevant information to the operators on the floor.

Participants saw significant value in the potential of having access to more data than is currently available, in a timelier manner and over a larger coverage area. They also expressed a desire to have more accurate and robust data from vehicle sensors, rather than the data they have today which is often captured through mobile devices or roadside sensors and does not readily allow control over data selection or volume.

# 3 Event-Driven Configurable Messaging (EDCM) System Architecture

EDCM system provides a *flexible messaging approach* between connected vehicles and TMC(s) that leverages existing connected vehicle infrastructure and communications to identify events and road conditions that potentially impede the safety and mobility of the traveling public. The EDCM system includes this messaging approach and supporting software for exchanging information between the TMC and EDCM-enabled vehicles, aggregating information with other data, analyzing that data, and determining the situation. As shown in Figure 4, the EDCM system operates within the larger EDCM environment which includes supporting connected vehicle infrastructure, connected vehicles, two-way communication and other features like security protocols and privacy in the TMC and region that are required for EDCM to function, but not detailed in this document.

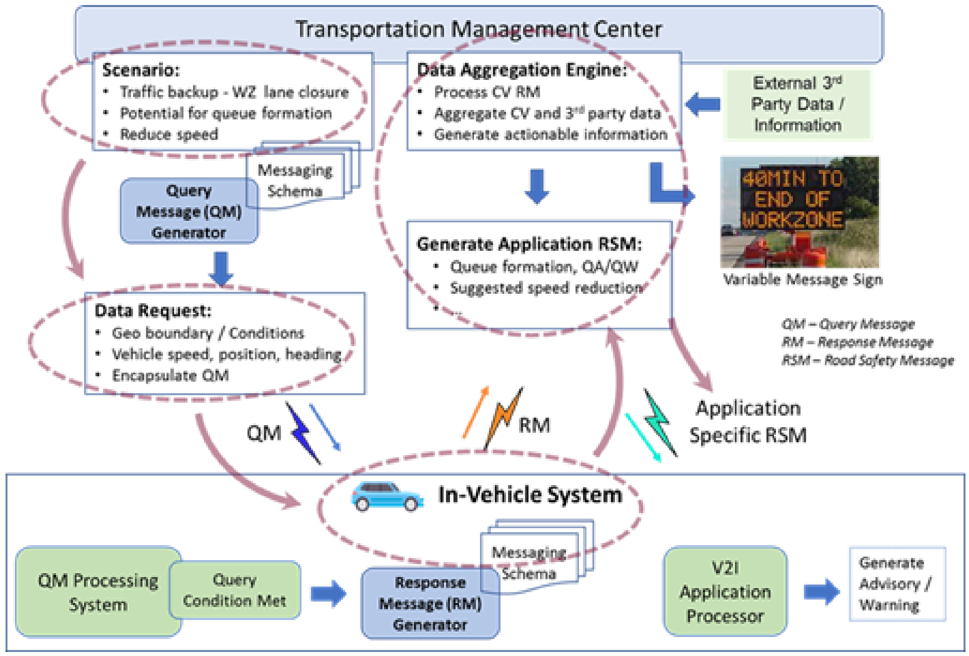



*Source: Vehicle-to-Infrastructure (V2I-2) Consortium and Virginia Tech Transportation Institute (VTTI)*

**Figure 4: EDCM System and Communication Framework**

EDCM system allows for forward and backward compatibility that reflects unforeseen changes in communication and vehicle technologies and can easily be adapted for new use cases through a flexible messaging strategy. As shown in Figure 5, the EDCM system architecture consist of the following:

- Query Message (QM) generation enables TMC personnel to request vehicle status information from CVs. Message formulation using a defined schema is detailed in Chapter 5.
- Wireless communication to establish connectivity with CVs, communicate QMs and receive Response Messages (RMs).
- RM processing, analysis and aggregation with external third-party information generating actionable information, including application specific RSMs for CVs and information for other TMC subsystems (e.g., variable message sign system) for non-connected vehicles.



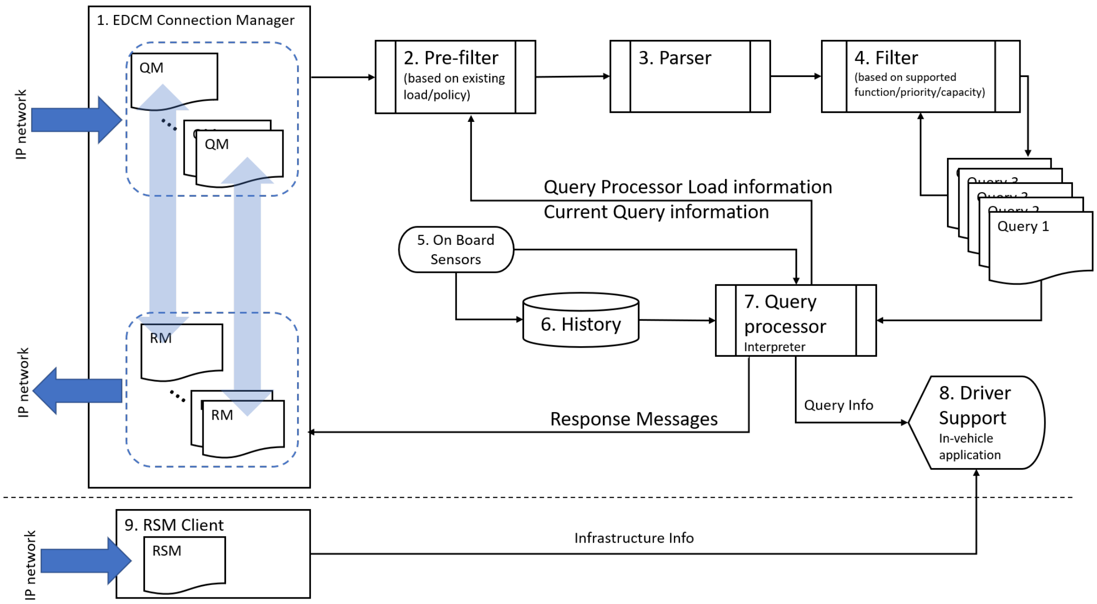

*Source: Vehicle-to-Infrastructure (V2I-2) Consortium and Virginia Tech Transportation Institute (VTTI)*

**Figure 5: EDCM System Architecture**

Communication between the in-vehicle On-Board Unit (OBU) and the TMC will be over a Transport Control Protocol (TCP) connection. The OBU will request a TCP connection with the TMC. That TCP connection will remain open to allow quick communication between the OBU and TMC. If the TCP connection is terminated, such as due to poor cellular reception, the OBU will re-establish the TCP connection.

## 3.1 In-vehicle Architecture Example

The OBU performs various EDCM functions in order to receive and respond to TMC queries. The process of consuming TMC information, such as RSMs, can be independent of its EDCM functions. While there may be multiple methods for achieving EDCM support within a vehicle, an example implemented during this project is shown in Figure 6, as explained below.




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 6: Functional Diagram of EDCM-related Components**

An example EDCM implementation may contain the following components:

1. **Connection Manager**: This handles connections to the TMC network, receiving and queuing QMs for processing. It also handles queuing and sending RMs to TMCs.

2. **Pre-filter**: Based on current processing load, the Query processor may not be able to handle new QM requests. A pre-filter either buffers or discards new queries without looking at their content.

3. **Parser**: This module parses the incoming queries into their basic elements such as trigger, region, duration, and requested data. This allows the filter module to decide whether to process an incoming QM or not based on relevance and data availability.

4. **Filter**: Decides whether to further process a query based on its internal policy, priority, current load, overall processing capacity.

5. **On-board Sensors**: Integrated within the vehicle are on-board sensors that provide the measurements needed to both satisfy the QM's trigger criteria as well as fulfill the data request(s).

6. **History**: Some QMs refer to or request past data. For this purpose, a history of sensor measurements involved in such triggers is kept in memory.

7. **Query Processor**: The Query Processor is essentially an interpreter. It periodically checks all active queries for satisfaction of trigger conditions and pre-conditions. If these are satisfied, the corresponding RMs are generated and sent to the Connection Manager.

8. **Driver Support:** In-vehicle applications combine information from the query processor and incoming road safety messages to provide support to the driver in terms of vehicle control or haptic / audiovisual interaction such as QA/QW.

9. **RSM Client**: Independent of EDCM function, a vehicle may include a client module that only consumes infrastructure information such as RSMs (as opposed to providing information through EDCM). The input from such a module can be used for in-vehicle driver support applications.

The software components described above may reside inside an OBU or an external device (such as a cellphone) connected to the vehicle by wire or wirelessly. The only components that need to be production-vehicle-integrated are sensors that are hard to install after-market.

# 4 Connectivity, Security and Privacy

This section describes establishing connectivity between the TMC and CVs and considerations for security and privacy. The EDCM Project did not perform a full design and verification of connectivity, security and privacy alternatives. This section provides an overview of some associated considerations that need to be further investigated and implemented within future works.

## 4.1 Connectivity Considerations

In developing this project, two methods of EDCM connectivity were identified. The first method was using DSRC from a Roadside Unit (RSU) and the second method using cellular based long-range communication. The second method below was employed within the bench test reference system. However, either method or a hybrid of the two may be feasible.

### 4.1.1 Connection to TMC via Vehicle-to-OEM-Center

Although not exercised within this bench test reference system developed within this project, vehicles may not need to establish direct connection with the TMC. Instead they connect to an OEM center, which in turn either (1) aggregates the data and communicates summery information with the TMC or (2) anonymizes vehicles and then connects them individually to the TMC. In this case, security and anonymity measures used in DSRC based Security Credential Management System (SCMS) security need not exist on the vehicle itself. Trust and anonymity are provided through the OEM center at the cost of increased network complexity.

### 4.1.2 Direct Vehicle-to-TMC Connection

When the vehicle directly connects to the TMC, two types of security must be considered in a direct vehicle-to-TMC-based (cellular modem-based) production system: (1) trust-related security and (2) privacy-related security.

Both server (TMC server) and clients (vehicle OBUs) need to be authenticated and trusted. This can be established through a commonly trusted certificate authority. At the same time, ideally, vehicles should remain anonymous and untraceable between sessions. SCMS/DSRC-type security scheme could be used for this purpose.

For Internet Protocol (IP) communications, a secure session makes eavesdropping difficult and at the same time is less demanding than SCMS/DSRC-type security in terms of processing power. Authentication/trust can be established at the beginning of the session.

## 4.2 Implemented Connectivity Example

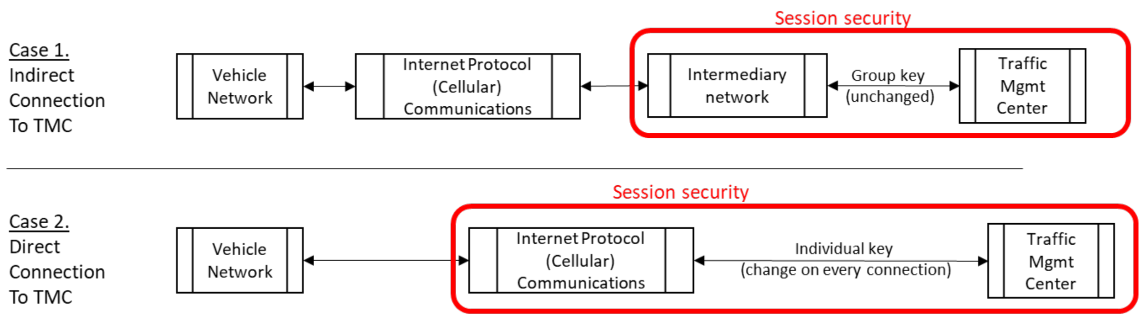
1. From the OBU

- The OBU will use a predetermined port, such as 4450, to open a TCP connection with the TMC
- The OBU will send the following three things, within 10 seconds
    - An EDCM header. This is an 8-byte constant message: {0x45, 0x44, 0x43, 0x4d, 0x52, 0x51, 0x53, 0x54} /* EDCMRQST in ASCII */
    - The size of the following <iamHere> message in bytes. This is a 2-byte short.
    - The <iamHere> message

2. From the TMC
    - If the TMC successfully receives the above information from an OBU, it will respond with the current Query Messages
    - Otherwise, the TMC will terminate the TCP connection

## 4.3 Security Considerations

Security ensures trust in and by the provider and consumer of information. For short range connectivity (e.g., DRSC), security is governed by existing SAE and IEEE standards. For long range connectivity, (e.g., Internet-Protocol-type or IP) similar security features can be implemented using Public Key Infrastructure (PKI). For TCP/IP connections, session security as shown in Figure 7 can be used as long as the vehicle-side keys change on every session to preserve privacy (see Section 4.4). In the case where anonymization is performed by an intermediary station/address, (e.g., OEM or group data center) individual/changing keys are not needed. The Figure 7 shows the two cases.




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 7: Session Security Needs for Individual or Group Vehicle Internet Protocol Addresses**

## 4.4 Privacy Considerations

It may be that privacy for EDCM is at least partially addressed by an opt-in model in which vehicle owners allow some level of tracking. However, to maximize acceptance, privacy by design may be a prudent method of implementing EDCM. In this context, privacy comes in the form of anonymity and untraceability.

As in IEEE 1609.2 standard, the use of pseudonym certificates can preserve privacy. Where as in DSRC privacy, the certificate is preserved by switching pseudonyms often, in an IP-based (assumed through cellular) communications, intractability is attained by limiting the geographical span, time duration and type of information provided in each RM since certain internet connections (of TCP type, for example) have to be tracked in order to maintain connectivity.

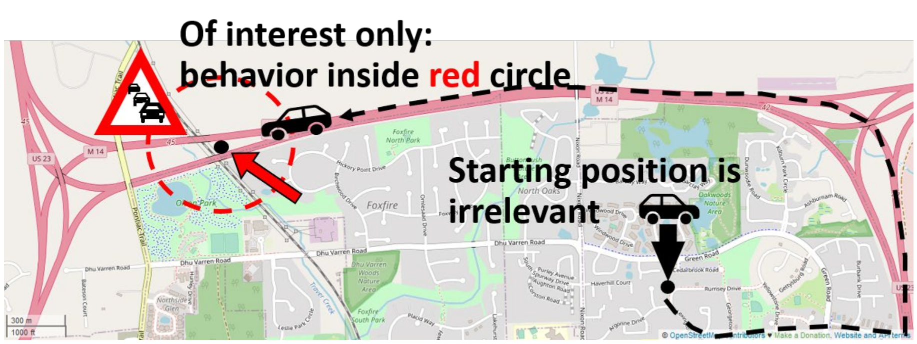
Even though RMs in EDCM syntactically allow tracking of vehicles indefinitely, an onboard-implemented privacy policy may prevent a vehicle from revealing its position more than a certain number of times, or outside a certain span of distance, to prevent tracking:

(a) For example, when an OBU that has revealed its position receives a QM for a distant location it is headed to, the OBU might chose to not respond, close the connection with the TMC, and reopen the connection under a different pseudonym to respond to the same QM, but only when the vehicle is closer to the region of interest does it respond. Similarly, this would be a privacy-preserving approach for a DSRC-based system where the OBU will respond to requests only when received close to the region of interest.

(b) Another way to avoid vehicle tracking is for the CV to not provide the information until it is absolutely needed. For example, once a vehicle establishes a connection with a TMC, there is no need to reveal its status and position until it provides the data required in relevant QMs. A vehicle starting a trip from a driver's home, for example, does not need to reveal its position until it is proximate to a region included in the query's domain. This would not be a privacy-preserving approach for a DSRC-based system unless the QM was broadcast simultaneously at many locations. For QMs that are broadcast only at a few locations, all responders can be traced back to one of those locations.

Developing and exercising the above policies are outside the scope of this project but essential to a production deployment of EDCM. Elements of responses deemed obligatory in current EDCM schema may be omitted in production implementation if they compromise privacy.

Figure 8 shows a connected vehicle registering for service at the starting position (black arrow) and does not need to reveal its position until it approaches the query area (red dashed circle indicating potential highway traffic jam).




*Source: Imagery Map data © 2020.* Overlaid *data by Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 8: Example - CV Registering for Services with TMC**

# 5 Flexible Messaging Schema

The query and response messaging schema enables the TMC and vehicle to collaborate through a standardized common language. The messaging schema, based on a well-defined data dictionary, provides a syntax through which infrastructure operators may generate a flexible QM. The vehicle query processor will use the same schema to decode the QM syntax, respond to the logic within and transmit the RM as appropriate.

Proper messaging syntax consists of language grammar rules, semantics, statements, variables and logical operators. Existing computer languages (e.g., Python, Java, C/C++) were reviewed to determine their suitability to decode QM and encode RM in an embedded system in a vehicle. Traditional programming languages require interpreters or compilers and supporting libraries. The EDCM Technical Team selected eXtensible Markup Language (XML), a universally accepted general-purpose mechanism for describing hierarchical data for messaging. It is important to note that the message processing will operate within a limited operating environment to reduce security concerns.

In XML, while data items are contained within elements and attributes and elements can contain textual data, attributes and other elements, attributes can just contain textual data. An XML schema, commonly known as an XML Schema Definition (XSD) [3], formally describes what an XML document can contain in the same way that a database schema describes the data it can contain in a database (i.e., table structure, data types, constraints). An XSD is developed that defines constraints on the structure and content of the messages above and beyond basic syntactical constraints imposed by XML. Although XML is designed for documents, in EDCM it is used for defining query and response messages. The XML defines a set of rules for encoding these messages in a format that is both human and machine readable. The World Wide Web Consortium's XML 1.0 Specification [4] of 1998 and several other related specifications [5,6] are all free open standards. As a result, there are also many tools available for programmers to work with XML across various hardware and software platforms.

## 5.1 Data Request in a Query

The QM is based on a well-defined data dictionary, known to both the connected vehicle and the infrastructure system. In this initial design, 23 different vehicle data elements including status information can be requested in a query.

The following vehicle data can be requested in a QM.

- **Basic information:**
    - Vehicle type, pseudo Vehicle ID
- **Position and dynamics:**
    - Position, heading, speed, acceleration, yaw, steering wheel angle
- **Status (safety):**
    - Activation of brake, traction control, stability control
- **Status (weather):**
    - Exterior lights, wiper position, external air temperature

A QM can request how and under what conditions to provide RM data to a TMC, for example:

- **Instantaneous value**:
    - vehicle speed, external air temperature or wiper position
- **Composite or averaged data:**
    - average of vehicle speed within given time period or distance traveled
- **Conditional:**
    - Based on certain vehicle status
    - Based on periodic update
    - Based on demand
    - Based on pre-samples
- **Within a region of interest (geofenced):**
    - Within a polygon
    - Within a circle
    - Within travel distance from specific location and direction
    - From → To location

# 6 Using XML to Formulate a Message

This section provides information on how a message is formulated. The query and response messages (QM and RM) are formulated by proper user input of values corresponding to XML tags. Two root elements are developed: (1) <qmFrame> for a query message initiated by a TMC and (2) <rmFrame> for a response message from an EDCM-enabled vehicle. As defined, an XML document has exactly one single root element. It encloses all the other elements and is therefore the sole parent element to all the other elements. The complete EDCM XML schema defined for messaging is provided in a companion document entitled "Event Driven Configurable Messaging (EDCM) XML Schema, Version 1.5" [7]. Following shows an example of XML syntax.

**Example - XML Syntax:**

The first line of an XML document declares that this is an XML document, including the version and encoding used.

```
<?xml version="1.0" encoding="UTF-8"?>
```

The following code shows an example of XML elements with attributes.

```
<?xml version="1.0" encoding="UTF-8"?>
<Case> <!-- Patient Care -->
      <patient name="John Doe" date="March-16-2020"
       condition=" Heart arrhythmia"></patient>
</Case>
```

The components of this XML element are:

| XML Components | |
|---|---|
| root element | Case |
| element name | patient |
| element attributes | name, date, condition |
| attribute values<br>name<br>date<br>condition | <br>John Doe<br>March-16-2020<br>Heart arrhythmia |
| Comment | <!-- Comment --> |

The names of XML elements and XML attributes are case-sensitive.

## 6.1 Query Message (QM)

In the query message frame, the root element <qmFrame> encloses following elements. An element consists of an opening tag, its attributes, associated content and a closing tag.

A query message contains, at a minimum, <eventMsg> and <dataRequest> elements. Other elements are optional.

### 6.1.1 Query Message Frame: <qmFrame>

This is a root element for a query message. Following elements can be enclosed in the root element to formulate a QM. Each element is described with use and example.

- <eventMsg> ... </eventMsg>
- <dataRequest> … </dataRequest>
- <qmDur> … </qmDur>
- <qmAction> … </qmAction>
- <gfRegion> … </gfRegion>
- <qmTrigger> … </qmTrigger>

### 6.1.2 Event Message: <eventMsg>

This is a required element defined in XML schema that describes the event for which this query is generated. This element has several associated attributes to provide event specific information in the query message. Required and optional attributes listed in Table 2. It should be noted that the schema version is a required attribute in the <eventMsg>. This is required in order for the vehicle to use the same version to parse and process the query and generate a response message.

**Table 2: <eventMsg> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| eventID | A unique identification number | R |
| msgDateTime | Message timestamp | R |
| vehType | Vehicle type | O |
| eventInfo | Event information / description | O |
| rmCommType | Communication method for query response | R |
| msgType | Query message type - iamHere, query, response or inform | R |
| msgPriority | Priority for message processing | O |
| vehResponsePct | Requested % of vehicles respond to this query | O |
| vehID | Unique vehicle ID | O |
| cCode | Cause Code (see European Telecommunications Standards Institute (ETSI) cause codes) | O |
| scCode | Sub cause code (see ETSI sub cause codes) | O |
| msgCount | Message Count is incremented to indicate a new message for the same event ID | O |
| schemaVer | XML schema version number | R |

* - R = Required, O = Optional

Example:

```
<eventMsg   eventID="23" msgDateTime="2020-03-20T06:15:00"
            vehType="5" eventInfo="Test Query Message"
            rmCommType="cellAndDSRC" msgType="query" cCode="30"
            scCode="2" msgCount="15" vehResponsePct="30"
            msgPriority="0" schemaVer="1.5">
</eventMsg>
```

### 6.1.3 Vehicle Data Request: <dataRequest>

This is a required element defined in XML schema that enables the request of various data from participating EDCM vehicles (vehicle speed, position, heading, etc.). In all, data requests for 23 different vehicle parameters supported is listed in Appendix - A. The number of pre- and post-triggered samples and sampling interval can also be specified in the data request. Table 3 lists the required and optional attributes supported in this element.

**Table 3: <dataRequest> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| dataName | Vehicle data parameters name | R |
| dataAvgName | Vehicle data parameters name for data averaging | O |
| preTrigSamples | Number of pre trigger samples | O |
| postTrigSamples | Number of post trigger samples | O |
| intervalTime | Sampling time interval | O |
| intervalDistMet | Sampling interval by travel distance (meters) | O |
| timeDur | Total sampling time duration | O |

* - R = Required, O = Optional

The following four examples illustrate various ways to request vehicle data in a QM.

1. Report vehicle speed every 1/10th mile.

```
<dataRequest>
      <provide dataName="speedMps" intervalDistMet="160"/>
</dataRequest>
```

2. Report brake status, acceleration and position when the triggering condition is met. Possible trigger conditions are described later.

```
<dataRequest>
      <provide dataName="vehBrakeStatus"/>
      <provide dataName="vehAccelStatus"/>
      <provide dataName="vehPos"/>
</dataRequest>
```

3. Report wiper position every 10 min for 1.5 hr.

```
<dataRequest>
```

```
        <provide dataName="wiperPos" intervalTime="00:10:00"
        timeDur="PT1H30M"/>
</dataRequest>
```

4. Report average ten pre-triggered samples taken at 1Hz.

```
</dataRequest>
        <provideAvg dataAvgName="speedChangeMps" preTrigSamples="10"
        intervalTime="00:00:01"/>
</dataRequest>
```

### 6.1.4 Query Message Duration: <qmDur>

This is an optional element defined in XML schema that defines duration for which the QM is valid using start and end date and time values as listed in Table 4.

**Table 4: <qmDur> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| startDate | Start date | O |
| endDate | End date | O |
| startTime | Start time | O |
| endTime | End time | O |

* - R = Required, O = Optional

Example:

```
<qmDur   startDate="2020-03-23" startTime="06:00:00"
         endTime="10:00:00">

</qmDur>
```

### 6.1.5 Query Message Action: <qmAction>

This is an optional element defined in XML schema that provides a mechanism to start or stop an active query message at a specified time for processing by the participating vehicles is listed in Table 5.

**Table 5: <qmAction> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| Action | Start \| Stop | O |
| Time | Time represented in date and time format as shown in example | O |

* - R = Required, O = Optional

Example:

```
<qmAction action="stop" time="2019-10-23T10:00:00"></qmAction>
```

### 6.1.6 Geo Fenced Regions: <gfRegion>

This is an optional element defined in XML schema that allows a TMC to define a geographic region of interest (geofence) for which the vehicle response is sought. There are four ways a region can be specified in a query as listed in Table 6.

- Polygon
- Circle with specified radius
- Drive distance from a desired location in specific direction
- From location and To location

For any of the four approaches, elevation can also be specified to further narrow the region of interest.

**Table 6: <gfRegion> - Supported Geofence Regions**

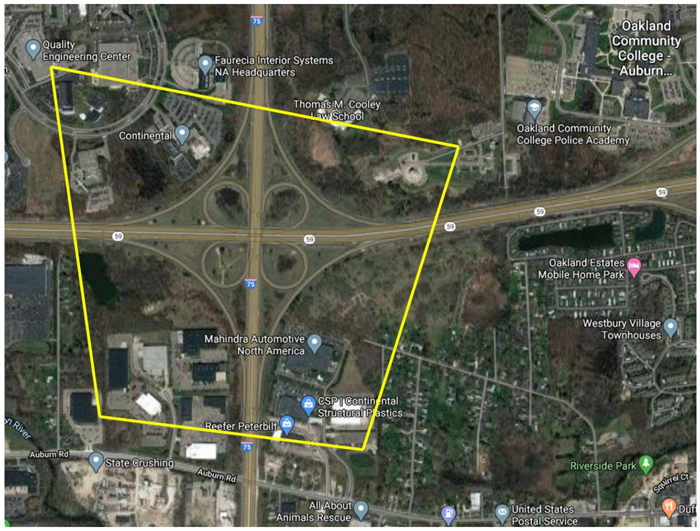
| Element/Attribute Name | Description | R / O* |
|---|---|---|
| gfRegionElev | Optionally, to define elevation for any of the following regions of interest. It allows in specifying road segments at certain elevation | O |
| poly | To define a region of interest as a polygon using a list of nodes represented in degrees of latitude and longitude | O |
| circle | To define a circular region of interest by specifying center of a circle and radius | O |
| from2toLocation | To define From → To location of travel | O |
| driveDistKm | To define a region of interest by driving distance from a specified location in specific direction | O |

* - R = Required, O = Optional

The following examples illustrate the use of each of the four methods to define a geofence for the region of interest.

Examples:

1. Poly – As shown in Figure 9, a polygon consisting of four nodes can be defined to represent a region of interest with elevation. Nodes are represented by latitude, longitude and elevation as min and max values in meters.




*Source: Imagery © 2020 CNES / Airbus. Maxar Technologies, U.S. Geological Survey, USDA Farm Services Agency. Map data © 2020. Overlaid data by Vehicle-to-Infrastructure 2 (V2I-2) Consortium*

**Figure 9: A Four Node Polygon using <gfRegion>**

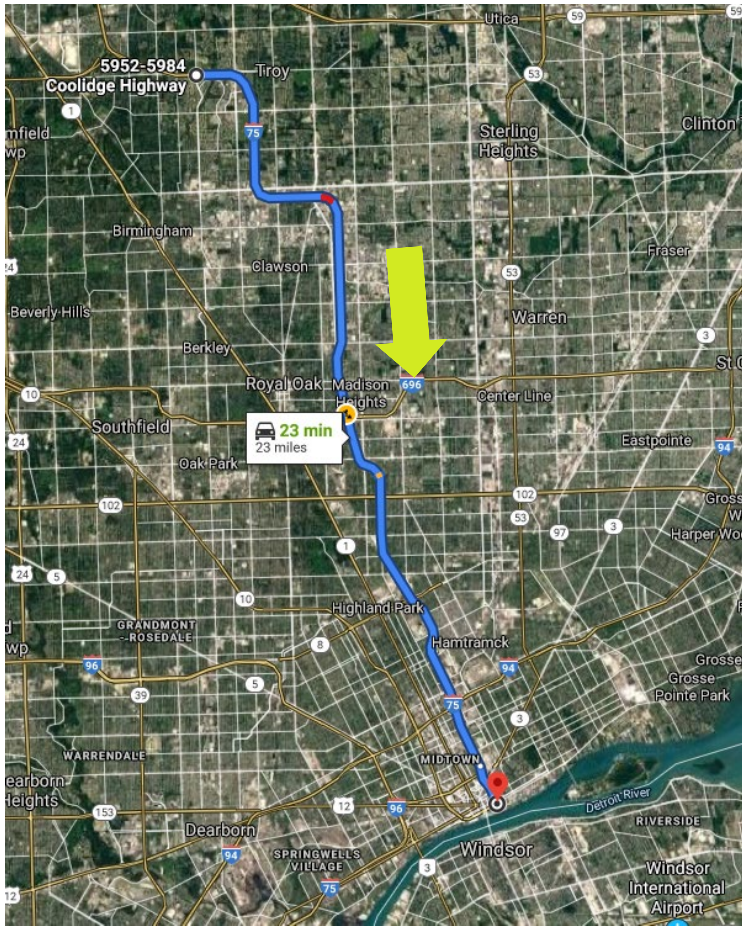
```
<gfRegion>
      <gfRegionElev>
         <elev elevMinMet="23" elevMaxMet="123"/>
      <gfRegionElev>

      <poly>
            <node latDeg=”42.651241" longDeg="-83.246959"/>
            <node latDeg="42.651648" longDeg="-83.231424"/>
            <node latDeg="42.634817" longDeg="-83.233567"/>
            <node latDeg="42.637898" longDeg="-83.252788"/>
      </poly>
</gfRegion>
```

2. Circle – A circular region of interest can be defined using a center and radius in meters.

```
<gfRegion>
      <circle>
            <center latDeg=”42.651241" longDeg="-83.246959”
            radiusMet="12000"/>
      </circle>
</gfRegion>
```

3. From2toLocation – The From and To locations consist of latitude, longitude, vehicle heading angle and heading angle tolerance to provide for a wider angle within certain radius of the from and to points. As shown in Figure 10, TMC can

request CVs traveling on southbound interstate I-75 near Troy, Michigan to Detroit, Michigan. It enables TMC to indicate From and To locations for general direction of travel and not restrict the specific travel path that CV may take.




*Source: Imagery © 2020 CNES / Airbus. Maxar Technologies, U.S. Geological Survey, USDA Farm Services Agency. Map data © 2020. Overlaid data by Vehicle-to-Infrastructure 2 (V2I-2) Consortium*

**Figure 10: From Location to Location Using <gfRegion>**

```
<gfRegion>
      <from2toLocatoin>
         <fromLocation latDeg=”42.611855" longDeg="-83.227565”
            headingDeg="90" toleranceDeg="45" radiusMet=”15”/>
         <toLocation latDeg="42.333673" longDeg="-83.233567"
            headingDeg="180" toleranceDeg="45" radiusMet=”15”/>
      </from2toLocation>
</gfRegion>
```

4. driveDistKm – An alternative to “from2toLocation” for region of interest can be defined as a driving distance from a specified location in a specific direction. For example, starting from southbound I-75 for 32km. The from location consist of latitude, longitude, vehicle heading angle, and heading angle tolerance to provide for a wider angle within a certain radius from the point of origin.

```
<gfRegion>
      <driveDistKm>
         <from latDeg="42.611855" longDeg="-83.227565" distKm="32"
              headingDeg="90" toleranceDeg="45" radiusMet="15"/>
      </driveDistKm>
</gfRegion>
```

### 6.1.7 Query Message Trigger: <qmTrigger>

This is an optional element defined in XML schema that allows a TMC to set various triggering conditions using logical and Boolean operators. Multiple triggering conditions can be set in the same QM. For example, vehicle status information such as speed, position, traction control system status may be requested only from certain vehicle types when external air temperature is near freezing (37° F or 2.7° C) as shown in the example below Table 7. Table 7 lists supported trigger elements. Appendix A lists supported vehicle parameters along with logical and temporal conditions for reporting status information.

**Table 7: <qmTrigger> - Supported Elements / Attributes for QM Triggering**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| Vehicle Parameters | See list in Appendix A | O |
| dataCond | Logical and Boolean conditions | O |
| timeDur | Time duration for trigger condition | O |

* - R = Required, O = Optional

Example:

When (and only when) all of the following conditions are met (Logical AND) and the requested data is reported. In this example, when vehicle speed is GE 31.3 m/s (70 mph), speed change is >= 60% in less than 20s and exterior temperature is LE 2.7°C (37°F).

```
<qmTrigger>
  <when vehType="3"/>
  <when speedMps="31.3" dataCond="GE"/>
  <when speedChangePct="60.0" dataCond="GE"/>
  <when timeDur="PT20s" dataCond="LE"/>
  <when extAirTempC="2.7" dataCond="LE"/>
</qmTrigger>
```

## 6.2 Response Message (RM)

In the response message frame, the root element <rmFrame> encloses following elements. An element consists of an opening tag, its attributes, associated content and a closing tag. A response message contains, at a minimum, <eventMag> and <vehVars> tags. Other elements are optional.

### 6.2.1 Response Message Frame: <rmFrame>

This is a root element for generating a response message. Following elements are enclosed in the root element to formulate a RM.

- <eventMsg> … </eventMsg>
- <vehVars> … </vehVars>
- <vehPos> … </vehPos>
- <vehAccelStatus> … </vehAccelStatus>
- <vehBrakeStatus> … </vehBrakeStatus>
- <extLightStatus> … </extLightStatus>
- <gfRegionEntryExitStatus> … </gfRegionEntryExitStatus>

### 6.2.2 Event Message: <eventMsg>

This is a required element defined in XML schema used inside the <rmFrame> when responding to a query. See subsection 6.1.2 for more detail. The following example illustrates eventMsg.

Example:

```
<eventMsg   eventID="23" msgDateTime="2020-03-20T06:15:05"
            vehType="5" eventInfo="Test Query Message"
            rmCommType="cellAndDSRC" msgType="response"
            cCode="30" scCode="2" msgCount="15"
            vehResponsePct="30" msgPriority="0" schemaVer="1.5”>
</eventMsg>
```

### 6.2.3 Vehicle Parameters: <vehVars>

This is a required element defined in the XML schema that contains the list of attributes for providing vehicle parameters in the response message. Table 8 provides a list of supported attributes.

**Table 8: <vehVars> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| vehType | Vehicle type to identify different category of vehicles | O |
| speedMps | Vehicle speed in m/s | O |
| yawRate | Measured yaw rate of the vehicle | O |
| steeringWheelAngle | Measured steering wheel angle | O |
| wiperPos | Wiper position (normal, intermittent, high) | O |
| extAirTempC | External air temperature in Celsius | O |

* - R = Required, O = Optional
Example:

```
<vehVars>
      <vehData vehType="3" speedMps="23.5" wiperPos="3"
      extAirTempC="2.75"/>
</vehVars>
```

### 6.2.4 Vehicle Position: <vehPos>

This is a required element defined in XML schema that provides vehicle positioning information. It contains five positional attributes. The list of supported attributes is provided in Table 9.

**Table 9: <vehPos> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| latDeg | Vehicle latitude in degrees | R |
| longDeg | Vehicle longitude in degrees | R |
| elevMet | Vehicle elevation in meters | O |
| headingDeg | Vehicle heading in degrees | O |
| toleranceDeg | Vehicle heading tolerance in degrees | O |

* - R = Required, O = Optional

Example:

```
<vehPos latDeg="42.3456" longDeg="-83.3476348 elevMet="23"
headingDeg="180"/>
```

### 6.2.5 Vehicle Acceleration Status: <vehAccelStatus>

This is an optional element defined in XML schema that contains a list attributes for vehicle acceleration parameters as shown in Table 10.

**Table 10: <vehAccelStatus> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| longAccel | Vehicle longitudinal acceleration | O |
| latAccel | Vehicle lateral acceleration | O |
| vertAccel | Vehicle vertical acceleration | O |

* - R = Required, O = Optional

Example:

```
<vehAccelStatus longAccel="-1.4325" latAccel="0.345"
vertAccel=0.223/>
```

### 6.2.6 Vehicle Brake Status: <vehBrakeStatus>

This is an optional defined in XML schema element containing the list of attributes for vehicle brake status. The list of supported elements / attributes is provided in Table 11. The status of these parameters is represented as "yes," "no" or "unavailable."

**Table 11: <vehBrakeStatus> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| brakeApplied | Brake applied | O |
| traction | Traction system | O |
| abs | Anti-lock braking system | O |
| scs | Stability control system | O |
| brakeBoost | Brake boost | O |
| panicBrake | Panic brake | O |

* - R = Required, O = Optional

Example:

```
<vehBrakeStatus brakeApplied="yes" abs="yes" panicBrake="no"
brakeBoost="unavailable"/>
```

### 6.2.7 Vehicle Exterior Light Status: <extLightStatus>

This optional element defined in XML schema that contains a list of attributes for providing the status of a vehicle's exterior lights as shown in Table 12. The status of these parameters is represented as Boolean.

**Table 12: <extLightStatus> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| normalBeam | Brake applied | O |
| highBeam | Traction system | O |
| fogLight | Anti-lock braking system | O |
| hazardLight | Stability control system | O |

* - R = Required, O = Optional

Example:

```
<extLightStatus normalBeam="false" highBeam="true"
fogLight="true"/>
```

### 6.2.8 Vehicle Geo Fenced Status: <gfRegionEntryExitStatus >

This is an optional element defined in XML schema that indicates when a CV enters/exits any geofenced area(s) defined in the active queries. This allows a TMC to know how many CVs are in the region(s) of interest that might be expected to respond to various query messages. Table 13 shows the associated attributes for this element.

**Table 13: <gfRegionEntryExitStatus> - Supported Elements / Attributes**

| Element/Attribute Name | Description | R / O* |
|---|---|---|
| eventID | Specified event ID in the active QM | O |
| gfStatus | Current CV status for the <gfRegion> in the active QM | O |

* - R = Required, O = Optional

Section 6.3.4 provides an example of a TMC querying the status of vehicles entering / exiting a geofenced region(s).

## 6.3 Examples of QM and RM

In this section, four examples of use case scenarios are described to illustrate formulating the QM and RM.

### 6.3.1 Traffic Congestion – Potential for Queue Formation

Traffic flow can be disrupted and delayed at a work zone since the traffic carrying capacity is reduced and vehicle speeds are lower. During peak hours, work zones cause traffic congestion and high potential for forming a queue. In this example, the TMC would like to know during a certain period of the day, the average speed of vehicles at various locations within a geographic region (area of bottleneck), the location of any significant speed differential for use in determining shockwave speed and other relevant parameters. The QM specifies that when the vehicle is within the region of interest indicated by <gfRegion> and when the triggering conditions are satisfied (indicated in <qmTrigger>, the requested vehicle data is returned in a RM.

QM from TMC:

```
<?xml version="1.0" encoding="UTF-8"?>
<qmFrame>   <!--Root element, start of QM-->

   <eventMsg eventID="23" msgDateTime="2020-01-23T06:00:00" vehType="5"
      eventInfo="Work Zone #23" rmCommType="cellAndDSRC"
      msgType="query" msgCount="1" msgPriority="0" schemaVer="1.5">
   </eventMsg>

<!--Data request from vehicles-->
   <dataRequest>
      <provide dataName="speedMps" intervalDistMet="160"
       timeDur="PT1H30M"/> <!--Report every 1/10th mile for 1.5 hr-->
```

```
      <provide dataName="vehPos"/> <!--Report vehicle lat, lon and
                                    elevation-->
   </dataRequest>

<!--Message Duration from 6am to 10am-->
      <qmDur startTime="06:00:00" endTime="10:00:00"></qmDur>

<!--Geo region in drive distance from GW Parkway to Hwy 28-->
   <gfRegion>
      <driveDistKm>
            <from latDeg="38.830333" longDeg="-77.043948" distKm="50"
             headingDeg="0" toleranceDeg="45"/>
      </driveDistKm>
   </gfRegion>

<!--Vehicles to start processing received query when the speed change
     is GE 60% in LE 10sec-->
   <qmTrigger>
      <when speedChangePct="60.0" dataCond="GE"/>
      <when timeDur="PT10S" dataCond="LE"/>
   </qmTrigger>

</qmFrame>  <!--End of QM-->
```

RM from EDCM vehicle: RM repeated every 160m travel with new information

```
<?xml version="1.0" encoding="UTF-8"?>
<rmFrame>   <!--Root element, start of RM-->

   <eventMsg eventID="23" msgDateTime="2020-01-23T06:45:10" vehType="5"
      eventInfo="Work Zone #23" rmCommType="cellAndDSRC"
      msgType="resonse" msgCount="1" msgPriority="0" schemaVer="1.5”>
   </eventMsg>

   <vehVars> <!--speed-->
      <vehData speedMps="23.5"/>
   </vehVars>

   <vehPos latDeg="42.35" longDeg="-83.2432" elevMet="321"
    headingDeg="170" toleranceDeg="45"/>

</rmFrame>  <!--End of RM-->
```

### 6.3.2 Inclement Weather– Slippery Road Condition

During inclement weather, the TMC would like to identify slippery road conditions in a circular region with high potential for ice formation, as shown in Figure 11. The EDCM-enabled vehicles are asked to respond to the query between 6:00 am to 12:00 pm when the following is true:

1. The vehicle is within the specified region of interest indicated by a 500 m radius circular region

2. Vehicle braking system’s Anti-lock Brake System (ABS) and traction control system are activated


*Source: Imagery © 2020 CNES / Airbus. Maxar Technologies, U.S. Geological Survey, USDA Farm Services Agency. Map data © 2020. Overlaid data by Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 11: Circular Region of Interest for Slippery Road Condition in Query Message**

When the conditions are met, the vehicle RM reports the average of the last ten speed samples collected at 1Hz prior to ABS and traction activation, along with external air temperature and vehicle position information.

QM from TMC:

```
<?xml version="1.0" encoding="UTF-8"?>
<qmFrame>   <!--Root element, start of QM-->

   <eventMsg eventID="13" msgDateTime="2020-01-31T06:00:00"
      eventInfo="Bad weather, slippery road" rmCommType="cellAndDSRC"
      msgType="query" msgCount="1" msgPriority="3" schemaVer=“1.5”>
   </eventMsg>

<!--Data request from vehicles-->
   <dataRequest>
      <provideAvg dataAvgName="speedMps" preTrigSamp=”10”
      intervalTime="00:00:01"/> <!--Report avg speed in 10s
                           before the trigger condition is met-->

      <provide dataName="extAirTempC"/> <!—External Air Temp-->
      <provide dataName="vehPos"/> <!--Report vehicle lat, lon and
                              elevation-->
   </dataRequest>

<!--Message Duration from 6am to 10am-->
```

```
      <qmDur startTime="06:00:00" endTime=12:00:00"></qmDur>

<!--Circular region including multi-deck bridge-->
   <gfRegion>
      <gfRegionElev>
            <elev elevMinMet="100" elevMaxMet="275"/>
      </gfRegionElev>

      <circle>
            <center latDeg="42.335378" longDeg="-82.064420"
             radiusMet="500"/>
      </circle>
   </gfRegion>

<!--Trigger conditions when both abs and traction are on-->
   <qmTrigger>
      <when abs="yes"/>
      <when traction="yes"/>
   </qmTrigger>

</qmFrame>  <!--End of QM-->
```


RM from EDCM Vehicle:


```
<?xml version="1.0" encoding="UTF-8"?>
<rmFrame>   <!--Root element, start of RM-->

<!--event message-->
   <eventMsg eventID="13" msgDateTime="2020-01-31T06:35:47"
      eventInfo="Bad weather, slippery Road" rmCommType="cellAndDSRC"
      msgType="response" msgCount="1" msgPriority="3" schemaVer="1.5”>
   </eventMsg>

   <vehVars> <!--Vehicle speed-->
      <vehData speedMps="12.5" extAirTempC="-2.35"/>
   </vehVars>

<!--Vehicle position-->
   <vehPos latDeg="42.353375" longDeg="-83.243232" elevMet="263"
           headingDeg="170" toleranceDeg="10"/>

</rmFrame>  <!--End of RM-->
```


### 6.3.3 Roadway & Road Surface Management - Deficient Roadway Section

The TMC is interested in gathering information on sudden speed changes due to roadway layout along a roadway section causing drivers to apply the brakes or make an abrupt change in direction. The region of interest is defined as vehicles traveling north bound within 3 km from a specific location. Vehicles are requested to provide, speed, position, yaw and steering wheel angle when anyone of the following trigger criteria is met.

1. Lateral acceleration is greater than or equal to 3.0 m/s/s or less than or equal to -3.0 m/s/s
2. Longitudinal acceleration is greater than or equal to 3.5 m/s/s/ or less than or equal to -3.5 m/s/s
3. Vehicle's panic brake assist system is activated

QM from TMC:


```
<?xml version="1.0" encoding="UTF-8"?>
<qmFrame>
      <eventMsg eventID="43" msgDateTime="2020-02-23 T08:35:47"
            eventInfo="Infra Design Assessment"
            rmCommType="cellAndDSRC" msgType="query"
            msgPriority="0" schemaVer="1.5">
      </eventMsg>

      <dataRequest>
            <provide dataName="speedMps"/>
            <provide dataName="yawRate"/>
            <provide dataName="steringWheelAngle"/>
            <provide dataName="vehPos"/>
      </dataRequest>

      <gfRegion>
            <driveDistKm>
                  <from latDeg="38.830333" longDeg="-77.043948"
                   distKm="3.0" headingDeg="0" toleranceDeg="45"/>
            </driveDistKm>
      </gfRegion>

<!--When any one of the following conditions is true-->
      <qmTrigger>
            <when latAccel="3.0" dataCond="GE"/></qmTrigger>
            <!-->= 3.0 m/s/s-->
      <qmTrigger>
            <when latAccel="-3.0" dataCond="LE"/></qmTrigger>
            <!--LE -3.0 m/s/s-->
      <qmTrigger>
            <when longAccel="3.5" dataCond="GE"/></qmTrigger>
            <!--GE -3.5 m/s/s-->
      <qmTrigger>
            <when panicBrake="yes"/></qmTrigger>
            <!--Panic brake activated-->
</qmFrame>
```


RM from EDCM Vehicle:


```
<?xml version="1.0" encoding="UTF-8"?>
<rmFrame>   <!--Root element, start of RM-->

      <eventMsg eventID="43" msgDateTime="2020-02-23T10:23:47"
```

```
            eventInfo="Infra Design Assessment"
            rmCommType="cellAndDSRC" msgType="response"
            msgPriority="0" schemaVer="1.5”>
      </eventMsg>

      <vehVars> <!--Vehicle speed, yaw and steering wheel angle-->
            <vehData speedMps="12.5" yawRate=”18”
             steeringWheelAngle=”45”/>
      </vehVars>

      <!--Vehicle position-->
      <vehPos latDeg="42.353375" longDeg="-83.243232" elevMet="263"
           headingDeg="170" toleranceDeg="10"/>

</rmFrame>  <!--End of RM-->
```

### 6.3.4 EDCM Vehicle Geofence Entry/Exit Status

The TMC wishes to gather entry / exit status of EDCM vehicles for specific region of interest. This helps the TMC assess the density of EDCM vehicles in different areas of interest when multiple queries are active in the different regions. In this query, a predefined eventID of 999 is used. In response, EDCM vehicles provide entry/exit status for each active query for the vehicle with eventID and gfStatus. The gfStatus may contain any of the following four values:

1. 0 = Outside/not applicable for geofenced region
2. 1 = Entered the geofenced region
3. 2 = Inside the geofenced region
4. 3 = Exited the geofenced region

Example:

QM from TMC:

```
<?xml version="1.0" encoding="UTF-8"?>
<qmFrame>   <!--Start of QM -->
      <!-- event -->
      <eventMsg eventID="999" msgDateTime="2019-11-23T06:00:00"
            vehType="0" eventInfo="CVs in GF Region"
            rmCommType="cellAndDSRC" msgType="query"
            msgCount="1"msgPriority="0"  schemaVer="1.5”>
      </eventMsg>

      <!-- data request -->
      <dataRequest>
            <provide dataName="gfRegionEntryExitStatus"/>
            <provide dataName="speedMps"/>
            <provide dataName="vehPos"/>
      </dataRequest>

</qmFrame>  <!--End of QM-->
```

RM from EDCM Vehicle:


```
<?xml version="1.0" encoding="UTF-8"?>
<rmFrame>

      <eventMsg eventID="999" msgDateTime="2019-11-23T07:35:00"
            vehType="0" eventInfo="CVs in GF Region"
            rmCommType="cellAndDSRC" msgType="response"
            msgCount="1"msgPriority="0" schemaVer="1.5">
      </eventMsg>

      <gfRegionEntryExitStatus eventID="23" gfStatus="0"/>
      <gfRegionEntryExitStatus eventID="25" gfStatus="1"/>
      <gfRegionEntryExitStatus eventID="30" gfStatus="2"/>
      <gfRegionEntryExitStatus eventID="33" gfStatus="3"/>

      <vehVars>
            <vehData speedMps="23.5"/>
      </vehVars>

      <vehPos latDeg="42.35" longDeg="-83.2432" elevMet="321"
            headingDeg="170" toleranceDeg="45">
      </vehPos>

</rmFrame>  <!--End of RM-->
```


### 6.3.5 Query Message Interpretation and Response Criteria

The following guidelines should be utilized when interpreting a QM and generating the corresponding RM:

- Within a <qmTrigger>, all <when> elements must be true for the trigger to be true.
- When there are multiple <qmTrigger> elements, each one is considered separately. If any of the <qmTrigger> element is true, a RM is generated.
- In a <when> clause, a dataCond attribute compares the vehicle's value to the value provided. For example, a dataCond attribute "LT" checks if the vehicle's value is less than the value provided.
- The timeDur attribute should be interpreted as a maximum duration.
- If a vehicle leaves a geofence boundary while a <qmTrigger> is otherwise true and a timeDur has not yet been reached, the vehicle stops generating and sending RMs and the timeDur is reset.
- If a <qmTrigger> switches from true to false before a timeDur has been reached, the vehicle stops sending RMs and the timeDur is reset.
- After a vehicle has sent a RM, it should wait for the reporting interval, if specified, before sending the next RM. Even if the trigger switches from true to false back to true in less time than the specified interval, the next RM should be delayed until the interval has expired.

- Any requested data that is not present in a RM is assumed to be unavailable. In particular, for brake values with possible values of “yes,” “no,” or “unavailable,” not providing that type of data is equivalent to sending “unavailable.” This produces a smaller message.
- speedChangeMps requires a timeDur attribute to specify the amount of time in which the speed change must occur.
- A timeDur used with a dataCond indicates how long that condition must be true before the clause is considered true.
- If a QM is received that contains the same eventID as a current QM, all processing on the current QM is stopped. All timers and status indicators pertaining to that QM are reset and the new QM starts fresh.
- The timeDur within <dataRequest> is optional.
- To make a <dataReqest> that reports forever, the timeDur attribute is omitted.
- When data are requested as different rates, the reporting rates of each type of data need to be maintained separately. Depending on the rates chosen, it is possible that sometimes only a single requested data element will be included in a RM and sometimes multiple requested data elements will be included in a RM.

# 7 Engineering Verification Implementation

Due to adjustments in the project, the first phase of the project was modified to conclude with an engineering verification of the messaging schema for QM and RM described in Section 6. The goal was to demonstrate that EDCM and the associated XML structure can be successfully implemented. The portions of the schema employed were based on a set of use cases identified during the stakeholder outreach which also exercise various portions of the messaging structure. These use cases were divided into two categories, event detection and non-event monitoring, based on the condition required for the vehicle to generate response messages.

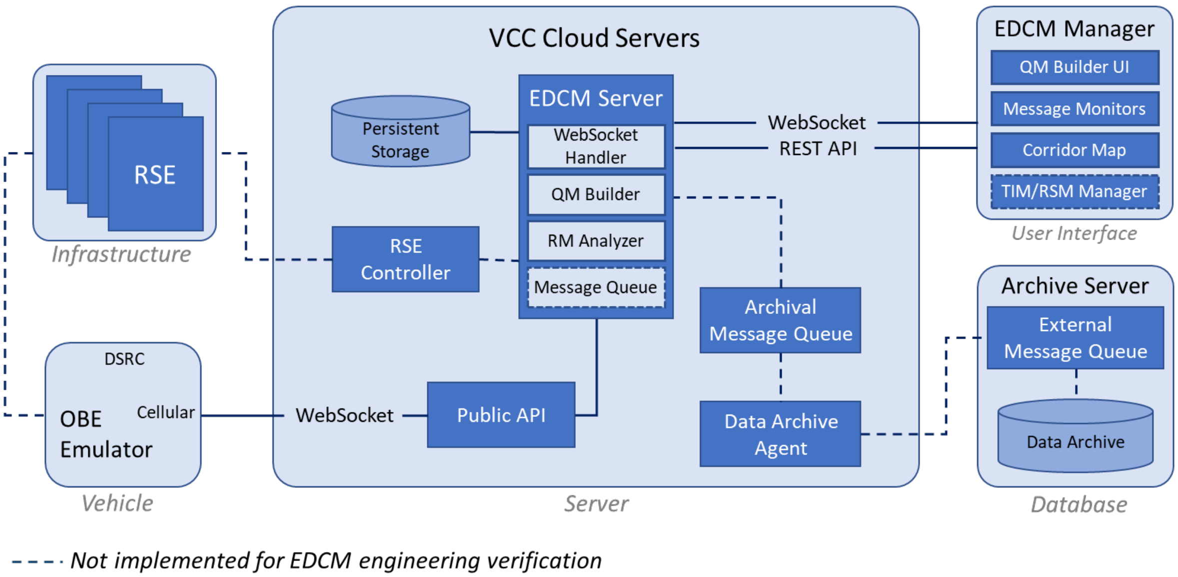
- Event Detection
  - Work Zone Monitoring: The following three use cases are relevant for work zones but not necessarily unique to WZM. The QMs associated with these three are also applicable outside of a work zone.
    - Queue Detection: As implemented, this is based on a simple speed-based threshold to indicate that the speeds in or around a work zone have dropped below a predefined value.
    - Hard Brake: This is based on brake status and longitudinal acceleration and is designed to capture events in which a driver may have to brake suddenly due to a condition that exits in or around a work zone i.e., merging vehicle, work zone ingress/egress, unexpected braking from other vehicles.
    - Late Merge: This QM is designed to capture aggressive driving behavior associated with merging traffic due to a lane closure that is part of a work zone. The trigger conditions as defined are based on a combination of yaw, lateral acceleration and steering wheel angle.
  - Weather Event: This triggered event is designed to capture environmental conditions that could be an indication of slippery road conditions. Vehicle variables available for triggering are external temperature, wiper status, stability control state and traction control state.
- Non-event Based Monitoring
  - Probe Message: The intent of this is to provide a low frequency, periodic snapshot of a minimal set of data from vehicles on the roadway to gauge the current conditions on the roadway.
  - Road Management/Engineering Evaluation: This is set up as probe message for vehicle traveling through a defined area to collect data for subsequent evaluation of a specific road segment or feature.

Since vehicles are not required to respond to a QM, in addition to the RMs associated with the use cases listed above, a simple “I am here” message is sent from the vehicle upon connection to the server to provide an indication of the potential number of vehicles that could respond.

## 7.1 Implementation

### 7.1.1 Engineering Verification Architecture

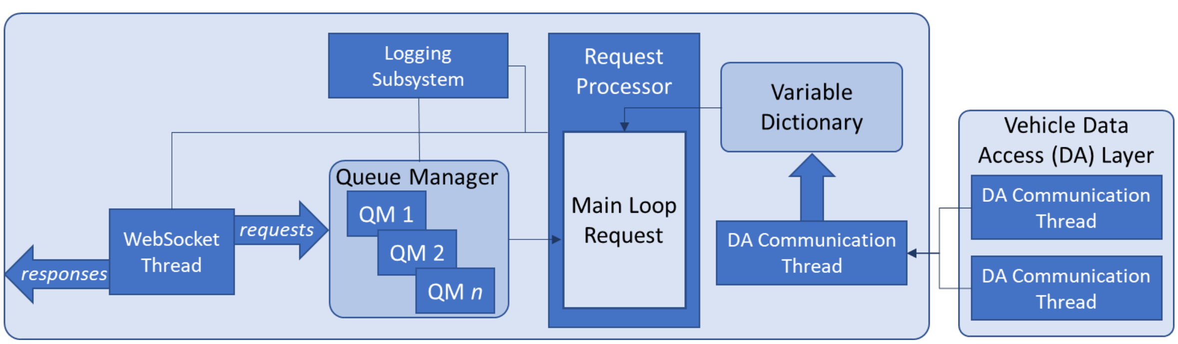
To support the engineering verification, the server and TOC application were set up in accordance with the following architecture (Figure 12). The existing Virginia Connected Corridor (VCC) was enhanced to support the EDCM strategy.




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium and Virginia Tech Transportation Institute (VTTI)*

**Figure 12: Engineering Verification Server Architecture**

On the vehicle side, on-board equipment (OBE) that would be utilized in a production system was emulated using a PC-based solution running Linux operating system. As indicated in Figure 13, while the architecture supports DSRC, communication between the vehicle and the server was limited to cellular via a Universal Serial Bus (USB) modem for this verification. Figure 13 shows the OBE emulator architecture.




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium and Virginia Tech Transportation Institute (VTTI)*

**Figure 13: Engineering Verification OBE Emulator Architecture**

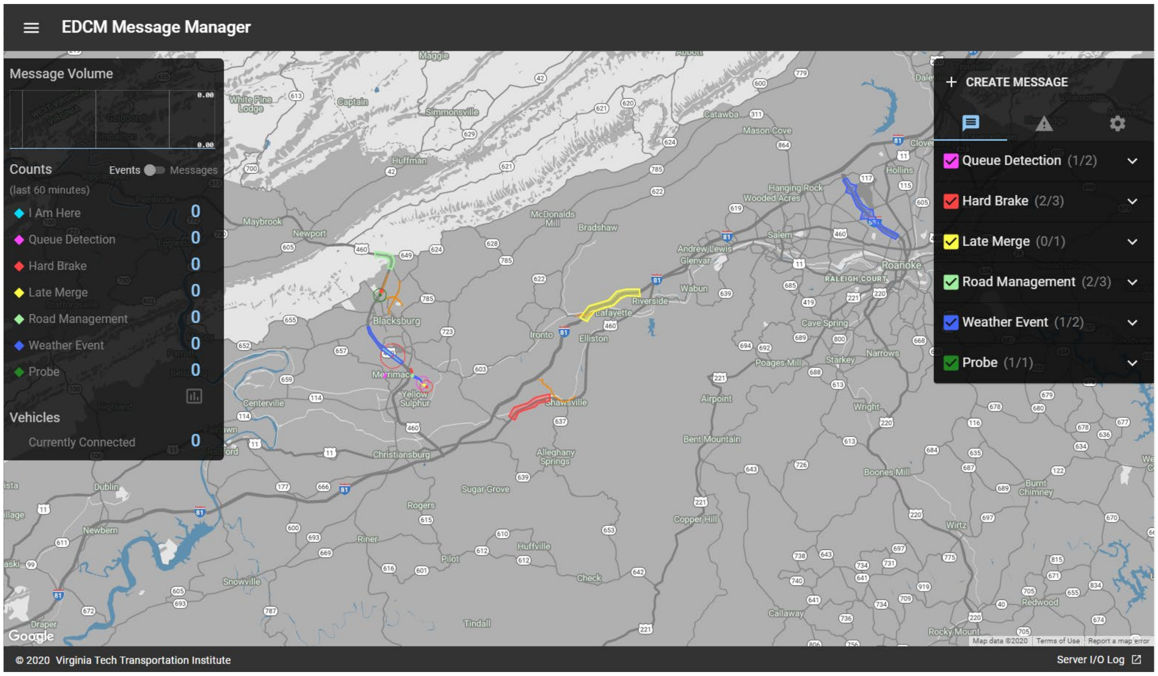
Since the primary purpose of the engineering verification was to confirm the message schema based on a representative set of use cases, some of the aspects of the system were not fully implemented such as support for DSRC communication (as indicated in Figure 12). In addition, rigorous security and privacy protocols were not implemented as a means to do that for (Vehicle-to-Everything) V2X applications has been specified (SAE J2945/1 for V2V safety applications) and demonstrated. However, as discussed previously, this is an important aspect of the system design.

### 7.1.2 TOC Client Application

A TOC client application was built to support the engineering verification use cases identified and demonstrate the basic functionalities afforded by the EDCM schema. This includes the creation and transmission of QMs, the receipt and interpretation of RMs, and the display of the data from the vehicles including basic compilation of data to provide examples of how the information could be compiled for TOC operators. Note that the application was designed and built to support the engineering verification for the work performed as part of Phase 1 of the EDCM Project. It is not designed to be a final TOC product but rather to demonstrate the basic elements that are needed to request and receive data and then show examples of how that data could be used in a TOC. The following section provides and overview of the application and then a brief description of the operation of the application including the creation of QMs and the viewing of the data from the RMs.

#### *7.1.2.1 Application Layout*

Figure 14 shows the layout of the starting screen. The panel on the left provides data regarding the real-time message volume of all incoming messages, counts of messages or events for all the active QMs for each category/use case received in the past hour, and the number of vehicles currently connected. In addition, a button allows for the graphical display that corresponds to the event and message counts over the last 24 hours. The panel on the right provides the controls for creating QMs and the associated alerts, generating and linking QMs to work zones defined in the VTTI Work Zone Builder application, and configuration of the data display. There are also controls for loading and playing vehicle data from previously recorded trips as well as the ability to view the xml for the QMs and the real-time log for the RMs being received by the server.

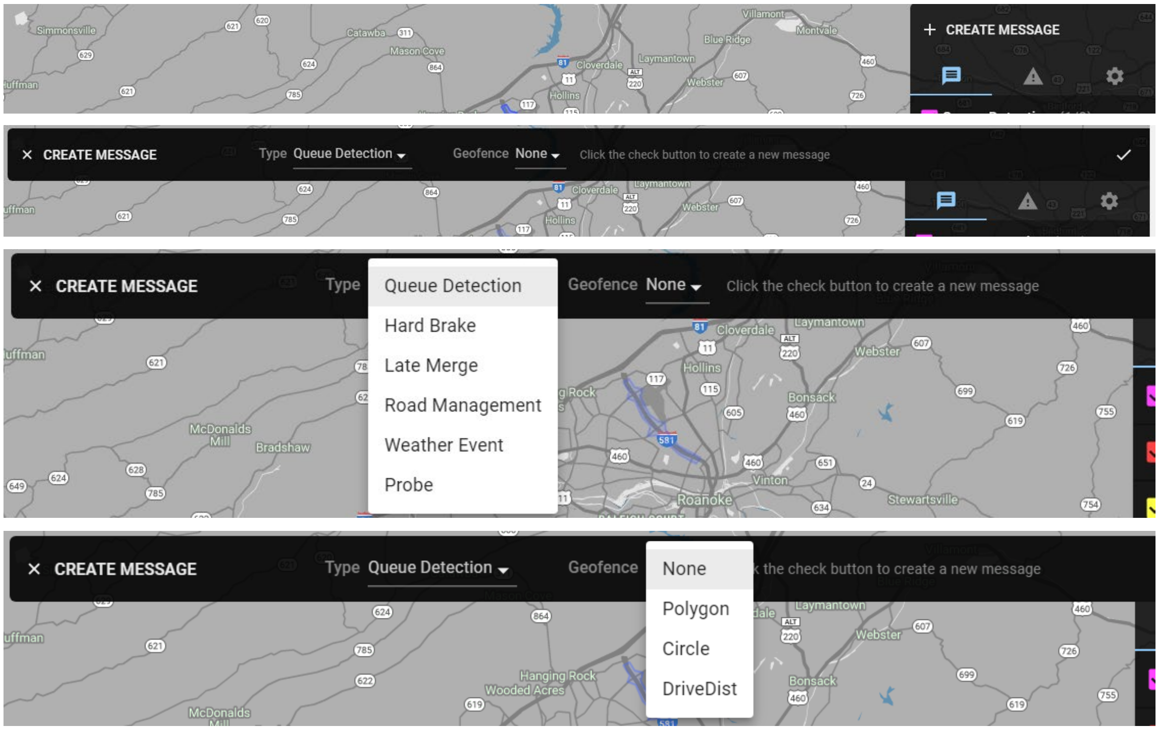



*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 14: EDCM Message Manager Layout**

### *7.1.2.2 QM Creation*

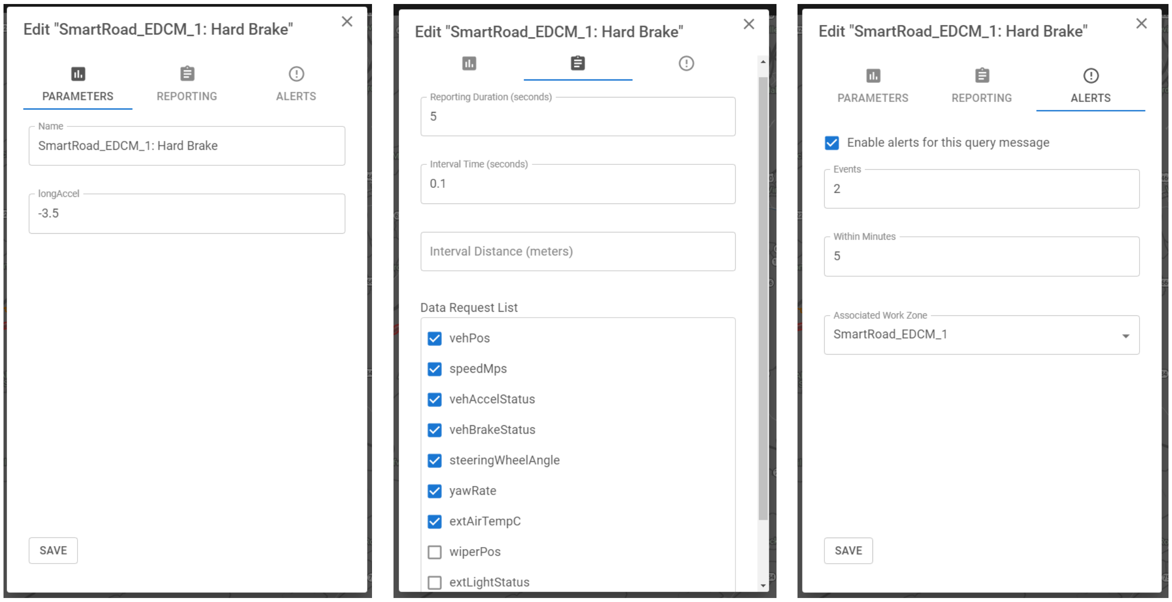
For this project, the application integrated the portions of the schema that support the selected use cases. The type of QM is constrained to same set of use cases. To create a new QM, the user clicks on the "+ Create Message" button which expands the bar to show menus for the QM type (which correspond to the six use cases) and the geofence method defined in the xml schema (none, circle, polygon and point/distance) as shown in Figure 15.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 15: Create Message Bar Options**

Clicking the check mark on the right side of the bar opens a window that allows the user to name the QM and triggering parameters for the event driven messages (Figure 16a) and set the reporting parameters including duration, frequency (in time or distance) and vehicle data content (Figure 16b). In addition, the user can create alerts associated with the QM that generate notification pop-ups based user defined parameters for the number of triggered events within a given period of time e.g., two hard brake events within 5 minutes (Figure 16c).

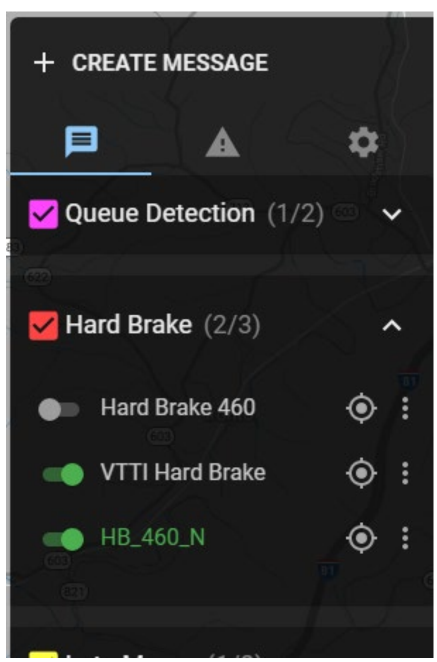



a) Parameters b) Reporting c) Alerts

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 16: QM Edit Window**

Each QM is displayed with the associated QM type. The list of QMs can be expanded/collapsed by clicking on the down/up arrow to the right of the QM type name (Figure 17). Once displayed, an individual QM can be activated and deactivated using the slider. Clicking the name of QM brings up the Edit window to let the user reconfigure the parameters and conditions associated with the QM.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 17: Example of Available QMs List**

Clicking on the reticle centers the map on the geofence region of the QM. The three horizontal dots brings up a message that allows the user to edit the geofence region, edit the QM parameters (also accessible by clicking on the name in the list view), deleting the message and viewing the automatically-generated xml associated with the message (Figure 18).


*Source: Virginia Tech Transportation Institute (VTTI)*

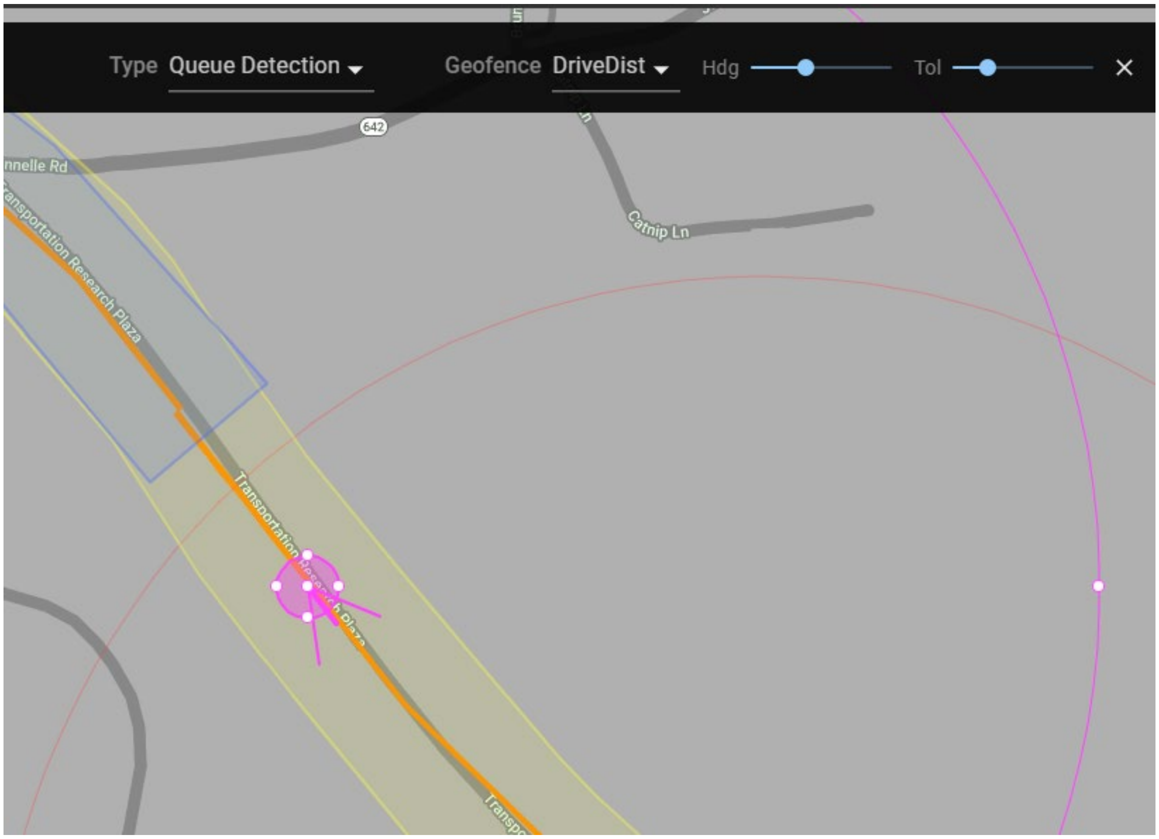
**Figure 18: Example of Raw Start XML Message Generated for QM**

Figure 19 shows the geographic regions associated with the QMs defined for the engineering verification and provide examples of the three geofence types used to define these regions. In the upper left corner, the light green circle is a geofence type associated with the Road Management QM. Any vehicle entering that region, defined by a center and diameter, that has received the associate QM, will start transmitting an RM as defined by the QM while in the region. The blue and yellow blocks, which correspond to the Weather Event QM and Late Merge QM, are defined as a polygon geofence type and follow the basic contour of the roadway. These are defined by the user with a set of control points which can be added to create the desired shape. The larger diameter circles (pink and red) correspond to Queue Detection and Hard Brake QMs, respectively, and specify a drive distance geofence defined by a starting region (depicted by the inner circle) that the vehicle must pass through within a given heading range (depicted by the open triangle). The vehicle will then generate associated RMs while it is driving within the area defined by the radius of the larger circle. From this graphic, the multi-query processing nature of the EDCM structure is shown where, in addition to the global probe message, a vehicle can potentially generate RMs associated with three of the additional QMs at given points along the test area where the geofence regions overlap i.e., weather, late merge and queue detection or late merge, queue detection and hard brake.


*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 19: QM Geofence Regions for Engineering Verification**

Geofence regions are defined within the interface by selecting points which can then be moved on the screen with the control handles. Figure 20 shows a magnified view of the queue detection geofence region and how it is defined. The two circles have control points that can be moved to position the center and the diameters. The direction the vehicle must be traveling when it enters the smaller circle and is defined by the line and V shape emanating from the center of the smaller circle. While the line denotes the required heading for the vehicle, the V shape defines the heading tolerance. These two values are manipulated with the sliders on the right side of the edit bar.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 20: Drive Distance Geofence Definition**

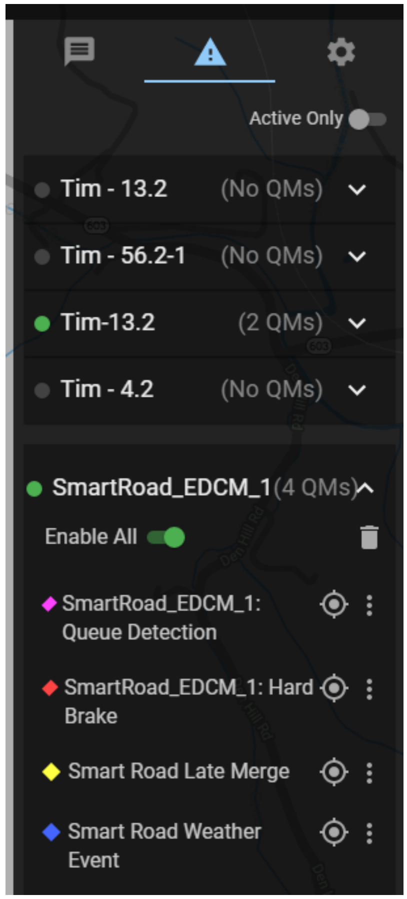
The circle geofence region is defined in the same way as the circles for the drive distance region.

A polygon region is defined by clicking on the map display which defines node points. Figure 21 shows an example of this with the satellite imagery overlay. Each of the control points on the polygon can be moved to define a specific shape. Additional node points are added by clicking and dragging segments.


*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 21: Polygon Geofence Definition**

The orange line that follows the basic contour of a section of the road indicates a simulated work zone which was generated within the VCC Work Zone Builder App. The Work Zone Builder App, which was not developed within this project, allows work zones to be rapidly setup and digitized by IOO and their contractors based on work zone layout standards. The integration of these systems shows how the EDCM strategy can automatically generate QMs based on data created by advanced IOO applications (and possibly 3$^{rd}$-party applications as well). The work zone is defined based on standard elements that are used to configure a work zone and can automatically generate QMs based on these standard elements including local probe, late merge, queue detection and hard brake. In addition, user generated QMs can be associated with an existing work zone. Figure 22 shows the list of work zones currently defined along with the individual QMs associated with work zone seen in Figure 19.

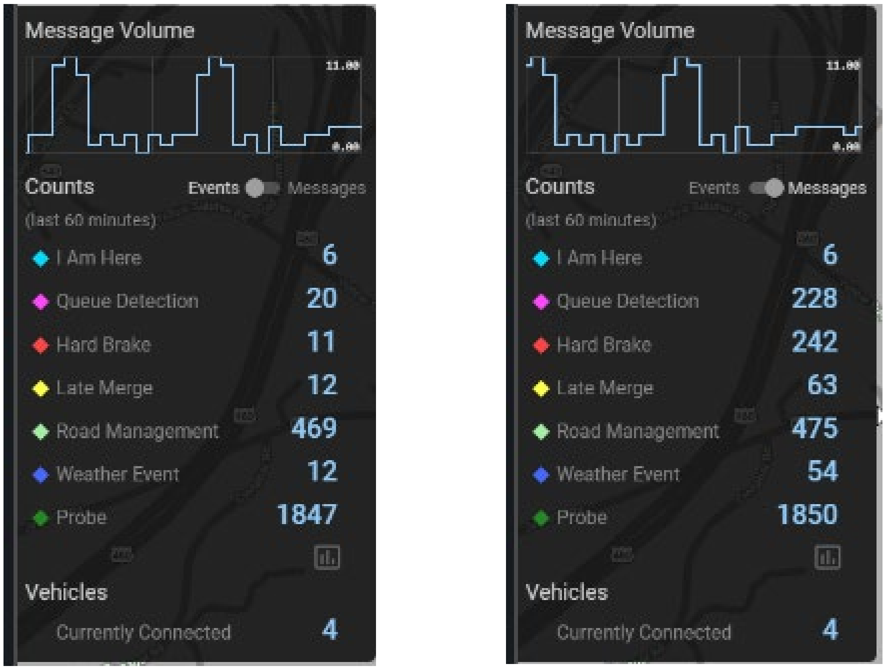



*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 22: Defined Work Zones and Associated QM List**

#### *7.1.2.3 RM Data Display*

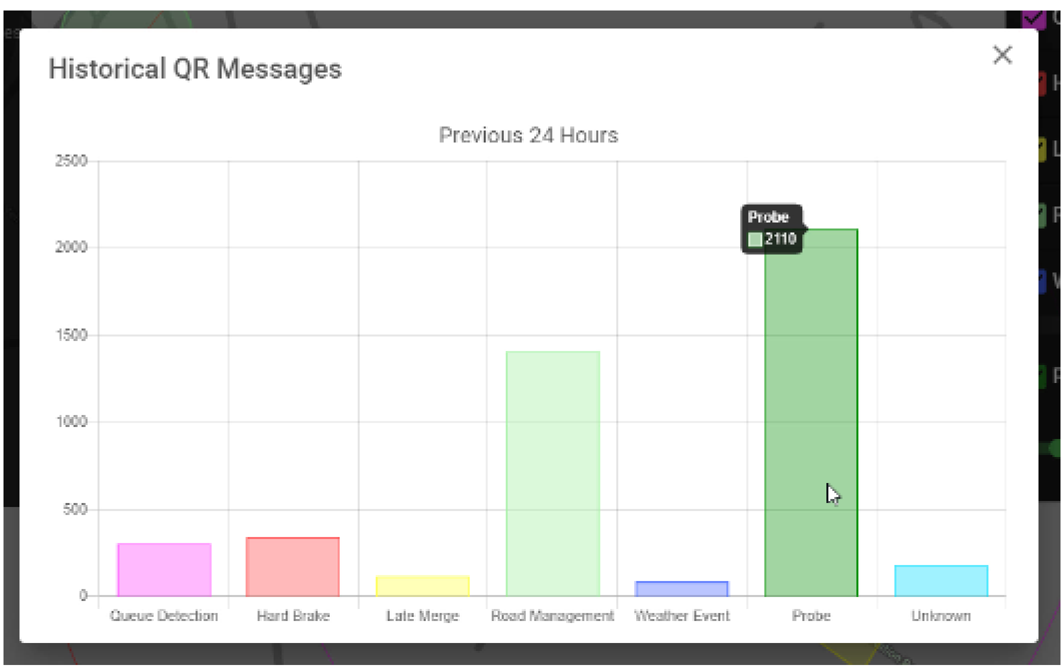
As discussed earlier, the frame on the left side of the screen provides quantitative data on the received messages from the connected vehicles. In Figure 23, the top strip chart provides a real-time display of the message volume received for the RMs from all connected vehicles. Immediately below the chart, a table shows the total of events (shown in the left frame) or message (shown in the right frame) received in the previous 60 minutes. Note that the periodic message types ("I am here," probe and road management) are not event driven messages. Therefore, the number of events is equal to the number of messages for the message types.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 23: Incoming RM Data Display**

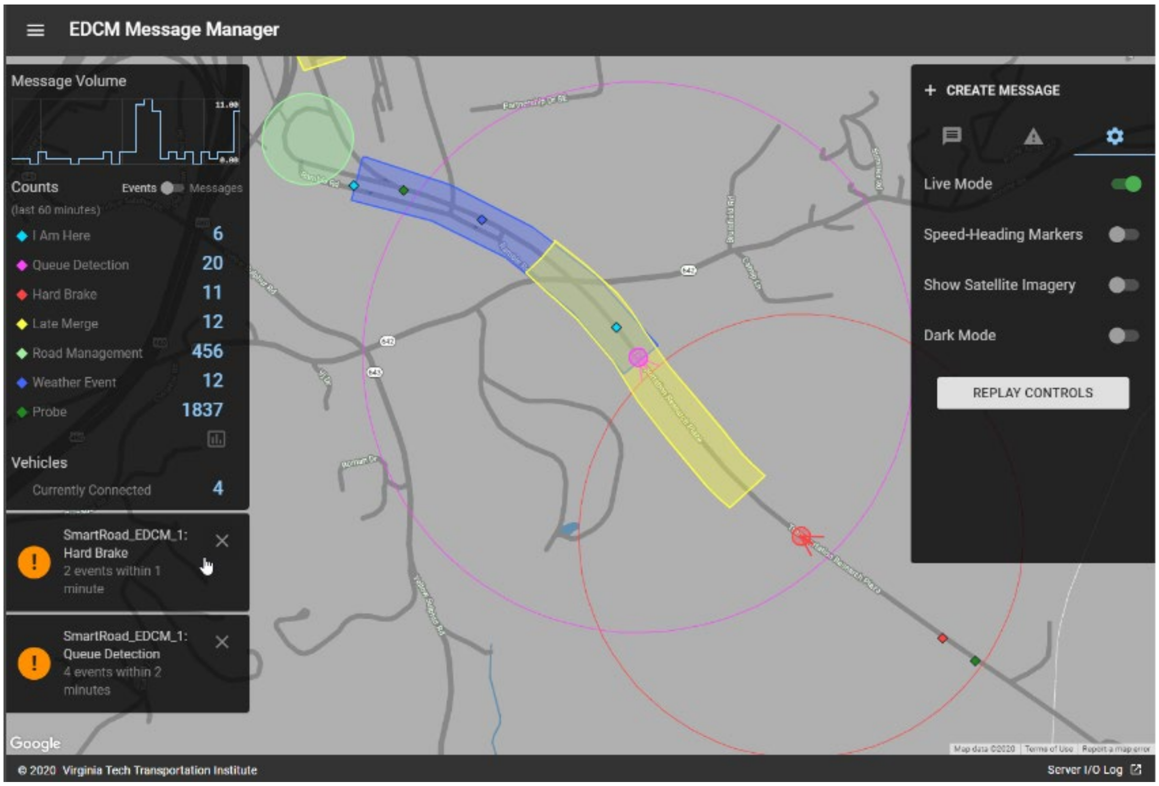
The current number of connected vehicles is shown at the bottom of the frame. The final quantitative data available to the user is a bar graph showing the number of events or messages received during the previous 24-hour period (Figure 24). A window showing this data is opened with the button immediately below the column of counts.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 24: Bar Graph of the RMs Received**

When an alert condition is met, the server displays a pop-up message below the RM data display frame that includes the QM name and conditions associated with the alert. Figure 25 shows two alerts that have been triggered, one for hard-braking event and the other for queue detection.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 25: Main EDCM Message Manager Window with Alert Notifications**

Figure 25 also shows a snapshot of the primary qualitative connected vehicle status represented by the colored squares. These vehicles are updated based on the last RM received by the server from a given vehicle. Following the roadway from the upper right to the lower left, four unique QM responses are represented by the dots: "I am here" (light blue), probe (green), weather event (blue), and hard brake (red). In addition, the tool allows the user to click on specific vehicle icons on the screen to view the associated RMs. An example of this is show in Figure 26.

Query Responses for: Global Probe

| ID | Event ID | Received Timestamp | Message Timestamp | Vehicle ID | Vehicle Index | Vehicle Type | Speed (mps) | Latitude (deg) | Longitude (deg) | Elevation (m) | Headi (de |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 191423 | 14 | 2020-06-08T15:49:23.213Z | 1591631420000 | 626 | 1714 | 1 | 13.72396931 | 37.18226956 | -80.38877532 | 602.936 | 1 |
| 191426 | 14 | 2020-06-08T15:49:24.592Z | 1591631499000 | 627 | 1716 | 1 | 0 | 37.18756892 | -80.39782829 | 640.314 | 1 |
| 191427 | 14 | 2020-06-08T15:49:25.211Z | 1591631422000 | 626_12 | 1713 | 1 | 12.25261397 | 37.18322034 | -80.38966163 | 608.818 | 1 |
| 191421 | 14 | 2020-06-08T15:49:21.592Z | 1591631496000 | 627_12 | 1715 | 1 | 0 | 37.18756881 | -80.39783255 | 640.479 | 3 |

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 26: QM Response Message Content**

Here we see the elements that are included in the probe message that are being returned by the four vehicles currently on the roadway. From the data, we observe that the first vehicle (Vehicle ID 626) has started to move and has its shadow vehicle while the second vehicle (Vehicle ID 627) has not. This is likely not a tool that would be readily available in a final TOC application, but it does provide functionality that is useful in the development and verification of operation.

## 7.2 Engineering Verification

The scope of engineering verification includes the demonstration of (1) basic functionality of EDCM schema, (2) functionality to support selected operational use cases, and (3) basic concepts for supporting TOC function. The final demonstration of the engineering verification was conducted on the Virginia Smart Road by two vehicles and took about an hour. It may be helpful for the reader to understand that verification took place during the COVID-19 epidemic in June of 2020 wherein social distancing requirements were in effect. Thus, demonstration was limited to remote viewing in real-time and offline review of video and data capture.

### 7.2.1 Approach

As previously discussed, the purpose of this activity is to verify the functionality of the basic design of the message schema and the associated tools used to realize the message concept. As such, the data collection focused on evaluation of key data elements to confirm the transmission and receipt of information, confirmation of operation (primarily inspection of the display of the information), and subsequently an assessment of the efficacy of the schema itself. Since this was an interim development step, the effort was scoped to include the specific use cases presented earlier. Consequently, the entirety of the schema was not implemented or tested, however, a sufficient portion was included to confirm what was identified as the primary use cases. As intended, this level of effort demonstrates that EDCM strategy is sufficiently mature to warrant the additional research originally planned for Phase 2 of the project.

### 7.2.2 Vehicles and Equipment

Two 2011 Buick La Crosse vehicles were used during this project. Each vehicle had a Controller Area Network (CAN) interface connected to a VTTI FlexDAS which was configured as an OBU emulator. The Flex DAS established connections, received QMs from the server, established agents/listeners which monitored the CAN data for conditions specified in the QM, assembled RM with the requested vehicle data, and transmitted messages back to the server. Test vehicles also were equipped with web cameras for remotely monitoring and capturing the demonstration using video.

To permit additional verification of the EDM strategy beyond the two instrumented vehicles, the bench test system had the functionality to instantiate virtual vehicles in two ways. The first was by playing back log files generated during previous trips. These files can be selected and played back from the EDCM Message Manager with the added ability to scale the speed to allow for the simulation of slower and faster traffic. The team implemented this feature to allow for the creation of higher traffic density during the verification activity. In addition, the live vehicles have the ability to create one or more shadow vehicles. These vehicles follow the same trajectory of the live vehicle, albeit at a user defined time lag. In this way, multiple vehicles execute the maneuvers associated with each scenario in sequential order, providing additional data streams with a known input allowing verification of EDCM support for repeated triggered RMs across multiple vehicles (e.g., emulating crowed sourcing of IOO-relevant data).

### 7.2.3 Virginia Smart Road

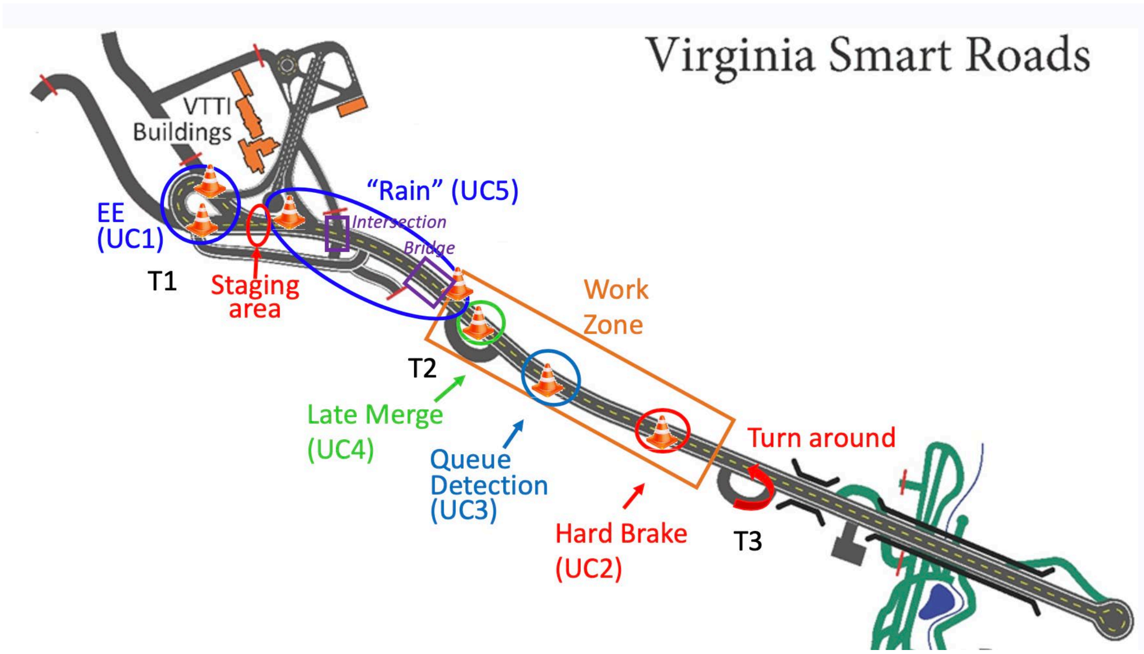
The demonstration was conducted on the Virginia Smart Road (Figure 27). The Virginia Smart Road is a 3.54-km (2.2-mile) closed-course test track that was designed to facilitate research on V2X communication, human factors, and transportation safety, as well as road surface properties. The road is built to Virginia Department of Transportation and Federal Highway Administration standards. To ensure participant safety when conducting experiments, the Smart Road restricts public access and is monitored through video surveillance by control room staff.

### 7.2.4 Test Protocol and Locations of Scenarios

Two vehicles, offset about 180 degrees of phase, conducted the demonstration on the closed course without any interactions with each other and other objects. Trained drivers performed scripted maneuvers at or above the predetermined threshold (but within normal driving range) to trigger response messages for the server. Prior to the demonstration, the research team placed cones and signs to indicate and mark areas for maneuvers or tasks based on the use cases identified in IOO and stakeholder interviews which include:

- UC1: Road management/engineering evaluation
- UC2: Work Zone, Queue detection based on the speed of the vehicle
- UC3: Work Zone, Hard brake based on the brake status and acceleration of the vehicle

- UC4: Work Zone, Late merge based on steering wheel angle, yaw, and acceleration of the vehicle
- UC5: Weather event based on the wiper status




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 27: Test Protocol and Locations of Events on the Virginia Smart Road for the Engineering Verification**

Each test vehicle started driving laps from the staging area near the first turn around (T1), started the OBU emulator and live video and accelerated to 45 mph. Near the second turn around (T2), the test vehicle performed the late merge scenario (UC4) in a work zone by a hard braking (-3m/s$^2$) followed by an abrupt swerving (4 deg/s) maneuver. At the next cone within the work zone, the test vehicle performed the queue detection scenario (UC3) based on a simple speed-based threshold to indicate that the speeds in or around a work zone have dropped below a predefined value of 30 mph. Near the last cone in the work zone, the hard-brake scenario (UC2) was implemented by an abrupt braking maneuver (-4m/s$^2$). The test vehicle turned around at the T3 and accelerated up to 45 mph. Near the second turn around (T2), the test vehicle performed the rain scenario (UC5) by manipulating the wiper until it reached the cone near the merge. While the test vehicle turned around at T1 to approach the staging area, the research team conducted the engineering evaluation scenario (UC1). This was set up as a probe message for the vehicle traveling through a defined area to collect data for subsequent evaluation of a specific road segment or feature. Each test vehicle repeated the scenario for five times for the live demonstration and data capture.

## 7.3 Results and Discussion

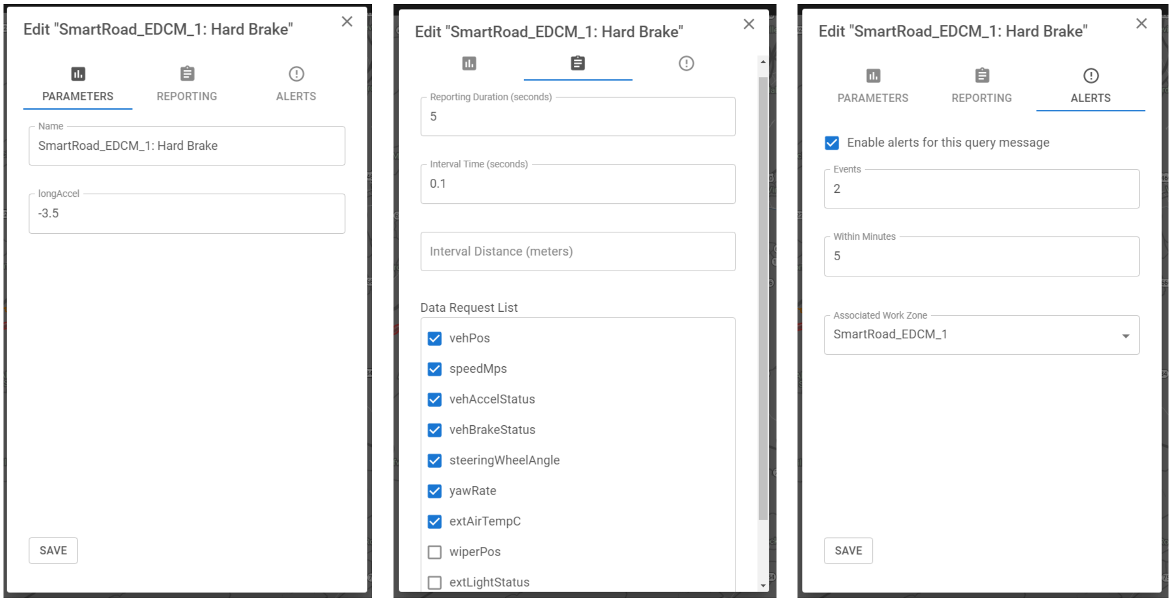
The engineering verification activity demonstrated the basic capability of the message design and schema to support the representative use cases identified during the workshops held with VDOT. In addition, the EDCM Message Manager and supporting server backend applications confirm the ability to manage the V2I communication to allow the retrieval of information of interest to a given TMC and subsequent presentation of the information in a timely and meaningful way to the TMC operators.

A principle benefit of the EDCM concept is to provide useful, targeted information in an easily digestible manner for a TOC operator. The use cases identified provide the opportunity to verify that the messages afford this capability. As demonstrated, the system allows:

1. Specific vehicle data to be requested at a specified rate and period as part of a QM
2. Data to be requested from a targeted area
3. Data to be sent based on user defined vehicle conditions and/or states
4. Timely and actionable information be provided to the operator

The following section traces the information flow during the engineering verification test from the server, to the vehicle, back to the server and finally to the user.

As discussed previously, the user configures the QMs to request targeted information from the vehicles. This provides specific information in a timely manner for given scenarios that may occur on the roadways. The QMs include the conditions for the initiation of the data transmission, the content of the messages from the vehicles, any geographic constraints when the messages are sent, and the rate and duration of the messages. The QM reporting configuration tool in the client application was shown previously (Figure 16) for the hard-brake event and is repeated here in Figure 28. This interface allows a TMC to configure the QMs that support their specific needs.




a) Parameters b) Reporting c) Alerts

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 28: QM Edit Window**

Once the user configures the message, the EDCM Message Manager generates the xml message based on the schema developed as part of the project. In Figure 29, we see the key elements to the message. The first element of the message provides general attributes for each QM including items such as the unique ID, date/time stamp, and user defined name. The dataRequest sub-element provides a list of the requested vehicle variables and the associated interval time and duration. The 0.1 second sample time can be confirmed, and the 5 second duration ("PT5S") is set for each variable requested e.g., vehicle position, speed, acceleration, brake status, steering wheel angle, yaw rate and air temperature. Immediately after this, the gfRegion sub-element specifies the geofence regions type and its associated parameters, and, in this case, the latitude and longitude node points used to define the polygon region. The gmTrigger sub-element defines the trigger conditions (e.g., longitudinal acceleration less than -3.0 m/s^2) that initiate the generation and transmission of the message by the vehicle.

```xml
<qmFrame>
  <eventMsg eventID="61" msgDateTime="2020-06-08T15:40:31.210Z" eventInfo
      ="SmartRoad_EDCM_1: Hard Brake" rmCommType="cell" msgType="query"
      msgCount="19" schemaVer="1.5" />
  <dataRequest>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName="vehPos"/>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName="speedMps"/>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName
        ="vehAccelStatus"/>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName
        ="vehBrakeStatus"/>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName
        ="steeringWheelAngle"/>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName="yawRate"/>
    <provide intervalTime="00:00:00.1" timeDur="PT5S" dataName="extAirTempC"
        />
  </dataRequest>
  <gfRegion>
    <driveDistKm>
      <from latDeg="37.180083093390834" longDeg="-80.38592462256207"
          radiusMet="21" distKm="0.5183844" headingDeg="122" toleranceDeg
          ="30"/>
    </driveDistKm>
  </gfRegion>
  <qmTrigger>
    <when longAccel="-3.5" dataCond="LT" />
  </qmTrigger>
</qmFrame>
```


*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 29: Hard Brake QM XML**

The discussion of the additional use cases will highlight differences in the QMs and associated XML which provide examples of the flexibility of the message structure. The Queue Detection event uses the drive distance geofence definition to allow for isolation of vehicles traveling along a road segment in a specific direction where a queue may form such as a work zone. Figure 30 shows the portion of xml which defines the geofence region used for the queue detection associate with the test protocol.


```xml
<gfRegion>
  <driveDistKm>
    <from latDeg="37.1828578444291" longDeg="-80.38938824684894" radiusMet
        ="21" distKm="0.53428197" headingDeg="142" toleranceDeg="30"/>
  </driveDistKm>
</gfRegion>
```


*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 30: Queue Detection Geofence Region QM XML Definition**

Here we see the coordinates for the center and radius of the circle which defines the gate location the vehicle must pass through. For this example, the heading and tolerance (142 ± 30 degrees) limits the response to vehicles traveling southwest on the road segment.

Once the vehicle passes through the gate in the correct direction, it will continue to respond until it travels 0.54 km.

The late merge provides an example of a more complex trigger consisting of Boolean operators relating lateral acceleration, yaw rate and deceleration. For test purposes, the trigger occurs by either a combination of lateral acceleration (> 1.0 m/s^2) and deceleration (< -1.0 m/s^2) or a combination of yaw rate (>2.0 deg/s) and deceleration (< -2.0 m/s^2). The following (Figure 31) shows the associated segment of XML for these trigger conditions. Note that the deceleration can be defined independent in each trigger condition.

```
<qmTrigger>
  <when latAccel="1.0" dataCond="GT" />
  <when longAccel="-1.0" dataCond="LT" />
</qmTrigger>
<qmTrigger>
  <when yawRate="2.0" dataCond="GT" />
  <when longAccel="-2.0" dataCond="LT" />
</qmTrigger>
```

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 31: Late Merge QM XML**

The weather event use case provides an example of a triggered event message whose duration is based on a geographical region (Figure 32). Once the trigger conditions are met, the vehicle continues to transmit associated RMs until the vehicle leaves the geofence region. Thus, we see the dataRequest sub-element and interval time but no time duration. Consequently, the vehicle will respond if the trigger conditions are met while it is in the region defined by a polygon defined by the node point attributes in the gfRegion sub-element. These points correspond to the region shown earlier in Figure 19.

```xml
<qmFrame>
  <eventMsg eventID="74" msgDateTime="2020-06-08T04:02:59.593Z" eventInfo
      ="Smart Road Weather Event" rmCommType="cell" msgType="query"
      msgCount="13" schemaVer="1.5" />
  <dataRequest>
    <provide intervalTime="00:00:02.0" dataName="vehPos"/>
    <provide intervalTime="00:00:02.0" dataName="speedMps"/>
    <provide intervalTime="00:00:02.0" dataName="vehAccelStatus"/>
    <provide intervalTime="00:00:02.0" dataName="vehBrakeStatus"/>
    <provide intervalTime="00:00:02.0" dataName="steeringWheelAngle"/>
    <provide intervalTime="00:00:02.0" dataName="yawRate"/>
    <provide intervalTime="00:00:02.0" dataName="extAirTempC"/>
    <provide intervalTime="00:00:02.0" dataName="wiperPos"/>
  </dataRequest>
  <gfRegion>
    <poly>
      <node latDeg="37.18803153732615" longDeg="-80.39738717132626"/>
      <node latDeg="37.18744391547241" longDeg="-80.39515825563488"/>
      <node latDeg="37.186565679378496" longDeg="-80.39299371295986"/>
      <node latDeg="37.185533578652404" longDeg="-80.39131733232556"/>
      <node latDeg="37.18409117823187" longDeg="-80.38969459587155"/>
      <node latDeg="37.18349284108516" longDeg="-80.39058508926449"/>
      <node latDeg="37.184781396981265" longDeg="-80.39193424039898"/>
      <node latDeg="37.18593317211816" longDeg="-80.39369108730374"/>
      <node latDeg="37.18670030017238" longDeg="-80.39566251092968"/>
      <node latDeg="37.18713407665388" longDeg="-80.39776268058834"/>
      <node latDeg="37.18803153732615" longDeg="-80.39745154434262"/>
    </poly>
  </gfRegion>
  <qmTrigger>
    <when wiperPos="0" dataCond="GT" timeDur="PT15S"/>   <!-- wipers must
        be on for a period of time -->
  </qmTrigger>
  <qmTrigger>
    <when traction="yes" dataCond="EQ" />
  </qmTrigger>
  <qmTrigger>
    <when scs="yes" dataCond="EQ" />
  </qmTrigger>
</qmFrame>
```


*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 32: Weather Event QM XML**

Similar to the late merge event trigger, the weather event message can be triggered by multiple conditions. However, in this case, it is based on the state of the wiper, traction control system or stability control system. The wiper state shows another feature of the trigger definition. Here the wipers must be on for at least 15 seconds before the vehicle generates the RM. This allows for simple filtering of transient events that may not be of interest e.g., cleaning the windshield. If either of the traction or stability control systems are active, a weather event message is generated. Note that these would likely have a higher sampling frequency and time-based duration in a real-world application.

The weather event also provides an example where an interval distance may be useful for sending a response message. The following (Figure 33) shows how the dataRequest sub-element changes to send the message every 2 seconds or 30 meters.

```
<dataRequest>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="vehPos"/>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="vehAccelStatus"/>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="vehBrakeStatus"/>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="steeringWheelAngle"/>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="yawRate"/>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="extAirTempC"/>
  <provide intervalTime="00:00:02.0" intervalDistMet="30" dataName
      ="wiperPos"/>
</dataRequest>
```

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 33: Weather Event QM XML for Distance Interval Reporting**

The final two use cases provide examples of a periodic, or non-event based, message type. The road management message (Figure 34) has a geographic constraint defined in the sub-element gfRegion, with a circle geofence defined by its center coordinates and radius.

```
<qmFrame>
  <eventMsg eventID="137" msgDateTime="2020-06-05T23:12:25.803Z" eventInfo
     ="EDCM_Eng_Verification" rmCommType="cell" msgType="query" msgCount="8"
     schemaVer="1.5" />
  <dataRequest>
    <provide intervalTime="00:00:00.5" dataName="vehPos"/>
    <provide intervalTime="00:00:00.5" dataName="speedMps"/>
    <provide intervalTime="00:00:00.5" dataName="vehAccelStatus"/>
    <provide intervalTime="00:00:00.5" dataName="vehBrakeStatus"/>
    <provide intervalTime="00:00:00.5" dataName="steeringWheelAngle"/>
    <provide intervalTime="00:00:00.5" dataName="yawRate"/>
  </dataRequest>
  <gfRegion>
    <circle>
      <center latDeg="37.188405766145095" longDeg="-80.39890851679172"
          radiusMet="106"/>
    </circle>
  </gfRegion>
</qmFrame>
```

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 34: Road Management QM XML**

Conversely, the probe message does not have a geographic constraint (Figure 35) and only specifies a minimal data set from the vehicle (position and speed) every 5 seconds to support basic traffic flow monitoring.

```xml
<qmFrame>
  <eventMsg eventID="14" msgDateTime="2020-06-08T15:21:17.312Z" eventInfo
      ="Global Probe" rmCommType="cell" msgType="query" msgCount="20"
      schemaVer="1.5" />
  <dataRequest>
    <provide intervalTime="00:00:05.0" dataName="vehPos"/>
    <provide intervalTime="00:00:05.0" dataName="speedMps"/>
  </dataRequest>
</qmFrame>
```

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 35: Probe QM XML**

Table 14 shows a summary of the Event ID associated with the QMs defined above.

**Table 14. Engineering Verification Query Message IDs**

| Message Type | Event ID |
|---|---|
| Queue Detection | 60 |
| Hard Brake | 61 |
| Late Merge | 69 |
| Road Management | 137 |
| Weather Event | 74 |
| Probe | 14 |

*Source: Virginia Tech Transportation Institute (VTTI)*

The above verifies the creation of the xml defined by the EDCM schema to support the use cases. The next step is to confirm the vehicle receipt of these QMs and the subsequent transmission of the data.

To confirm the receipt of the QMs by the vehicle, the team implemented vttiStatusMessage that provides a means to request the list of QMs on the vehicle. Figure 36, shows an excerpt from the server log recorded at 10 minutes and 44 seconds during the final video of the engineering verification test. It shows the xml attributes extracted from vttiStatusMessage for vehicle 1715 (vehID = 627_12) including a list of the QMs on the vehicle, if they are active (running = "yes") and a date/time stamp. As a result, it can be seen that the Event IDs listed in Table 14 are present and active. The additional Event IDs listed correspond to other QMs defined for different regions and message types.

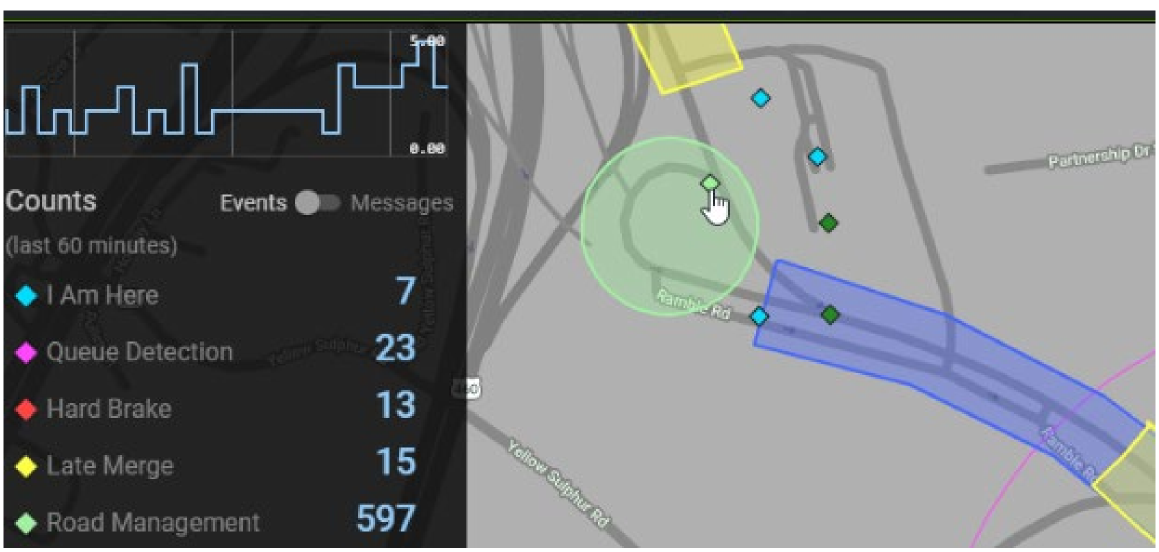
```
data:***********Extracted from rmFrame:  vehicle 1715 (vehID = 627_12) is at lat 37.18757086, lng -80.39782804
)
data:***********RECEIVED MESSAGE FROM VEHICLE 1715:
data:&lt;vttiStatusMessage&gt;
data:    &lt;qmStatus eventID="69" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="142" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="72" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="73" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="74" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="14" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="143" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="18" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="21" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="22" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="23" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="137" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="60" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;qmStatus eventID="61" running="yes" status="2020-06-08T15:52:16"/&gt;
data:    &lt;vehInfo latDeg="37.18757086" longDeg="-80.39782804" elevMet="640.394" headingDeg="172.0612" speedMps="0.0"/&gt;
data:    &lt;error/&gt;
data:&lt;/vttiStatusMessage&gt;
data:
```

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 36: Vehicle RM Status from Server Log**

Transmission of the RMs from the vehicles were confirmed quantitatively by reviewing the data received by the server and qualitatively through inspection of the vehicle video feed and the EDCM Message Manager.

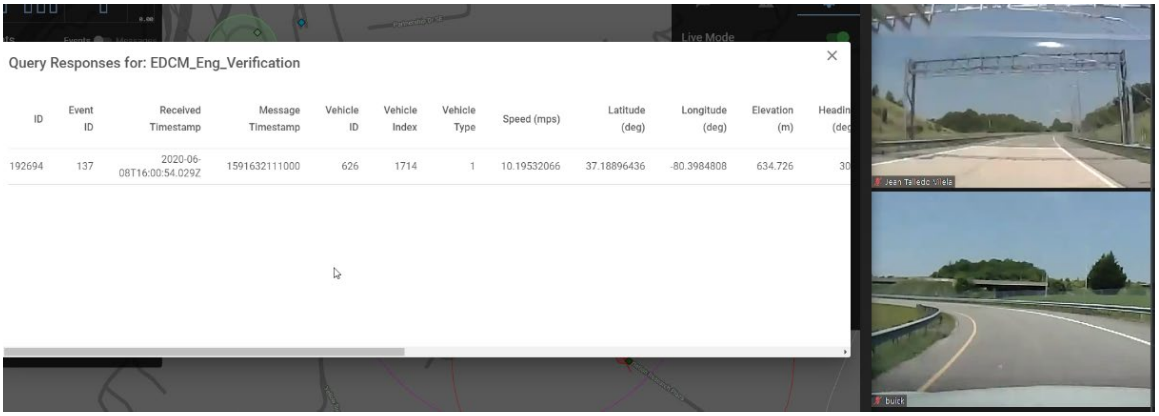
Since the client application allows for live review of data associated with moving map markers, clicking on a marker brings up the associated data for that marker. Figure 37 shows a screen capture during the testing. The light green marker indicates the RM is associated with the road management event.



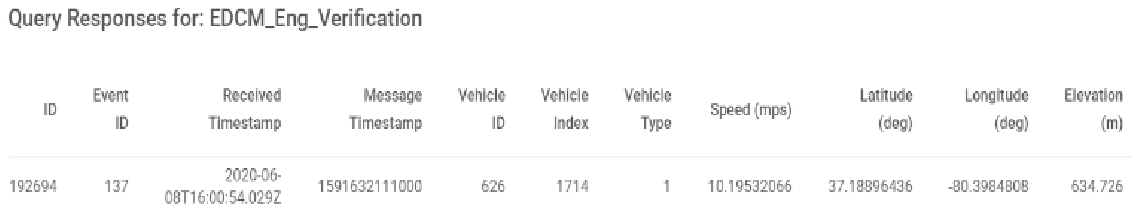

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 37: Road Management Message Response Example**

The next image (Figure 38a) is a screen capture showing the message associated with the marker and the live video from the vehicle that generated the RM (lower right). Figure 38b zooms in on the query response data.




a.



Query Responses for: EDCM_Eng_Verification

| ID | Event ID | Received Timestamp | Message Timestamp | Vehicle ID | Vehicle Index | Vehicle Type | Speed (mps) | Latitude (deg) | Longitude (deg) | Elevation (m) |
|---|---|---|---|---|---|---|---|---|---|---|
| 192694 | 137 | 2020-06-08T16:00:54.029Z | 1591632111000 | 626 | 1714 | 1 | 10.19532066 | 37.18896436 | -80.3984808 | 634.726 |

b

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 38 a, b: Query Response for Road Management Message**

The data table in the above figure shows the associated Event ID which corresponds to the road management QM defined above (Figure 34). The video frame in the lower right shows the vehicle in a sweeping left hand turn which is the start of the turnaround at the top of the VTTI Smart Road and corresponds to the position of the marker on the map in Figure 39.

Another example of the verification of the data RM from the vehicles to the server is shown in real time display of the server log available within the EDCM Message Manager. Figure 39 is a screen capture from the recording made during the engineering verification. Here the data is extracted from the server from five different vehicles: 1718, 1713, 1715, 1717, and 1714 as highlighted by the doted frames. (The blue highlighted item, vehID="626_12" was part of the live recording). Each vehicle is responding to event ID 14 which is the global probe message. This confirms the receipt of messages from multiple vehicles connected to the server.

From this, the functionality implemented for the bench test system to generate additional data from shadow vehicles and prerecorded test runs can be seen. This also demonstrates the playback ability. The first and fourth vehicles are prerecorded vehicles (as indicated by the r-12 and r-11), and the second and third vehicles are the shadow vehicles, 626_12 and 627_12, which are set up to repeat the behavior of the two vehicles being driven on the test track with a 12-second delay (e.g., vehicle 626_12 follows vehicle 626 by 12 seconds). The data received from vehicle 1714 (vehID = 626) is the only response from a live vehicle shown in this screen capture.

```
data:***********Extracted from rmFrame: vehicle 1718 (vehID = r-12) is at lat 0.0, lng 0.0

data:***********RECEIVED MESSAGE FROM VEHICLE 1713:
data:<rmFrame>
data:    <eventMsg eventID="14" msgDateTime="2020-06-08T16:00:02" vehType="1" msgType="response" vehID="626_12"
data:    <vehVars>
data:        <vehData speedMps="12.40886409"/>
data:    </vehVars>
data:    <vehPos latDeg="37.17747278" longDeg="-80.38123192" elevMet="553.183" headingDeg="304"/>
data:</rmFrame>
data:

data:***********Extracted from rmFrame: vehicle 1713 (vehID = 626_12) is at lat 37.17747278, lng -80.38123192

data:***********RECEIVED MESSAGE FROM VEHICLE 1715:
data:<rmFrame>
data:    <eventMsg eventID="14" msgDateTime="2020-06-08T16:01:21" vehType="1" msgType="response" vehID="627_12"
data:    <vehVars>
data:        <vehData speedMps="20.32987738"/>
data:    </vehVars>
data:    <vehPos latDeg="37.1846625" longDeg="-80.39101478" elevMet="620.623" headingDeg="139"/>
data:</rmFrame>
data:

data:***********Extracted from rmFrame: vehicle 1715 (vehID = 627_12) is at lat 37.1846625, lng -80.39101478

data:***********RECEIVED MESSAGE FROM VEHICLE 1717:
data:<rmFrame>
data:    <eventMsg eventID="14" msgDateTime="2020-06-08T15:59:07Z" vehType="1" msgType="response" vehID="r-11" s
data:    <vehVars>
data:        <vehData speedMps="0.0"/>
data:    </vehVars>
data:    <vehPos latDeg="0.0" longDeg="0.0" elevMet="0.0" headingDeg="0"/>
data:</rmFrame>
data:

data:***********Extracted from rmFrame: vehicle 1717 (vehID = r-11) is at lat 0.0, lng 0.0

data:***********RECEIVED MESSAGE FROM VEHICLE 1714:
data:<rmFrame>
data:    <eventMsg eventID="14" msgDateTime="2020-06-08T16:00:05Z" vehType="1" msgType="response" vehID="626" sc
data:    <vehVars>
data:        <vehData speedMps="19.75696025"/>
data:    </vehVars>
data:    <vehPos latDeg="37.17884523" longDeg="-80.38369103" elevMet="567.797" headingDeg="304"/>
data:</rmFrame>
data:

data:***********Extracted from rmFrame: vehicle 1714 (vehID = 626) is at lat 37.17884523, lng -80.38369103
```

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 39: Server Log Example Showing Responses from Five Vehicles**

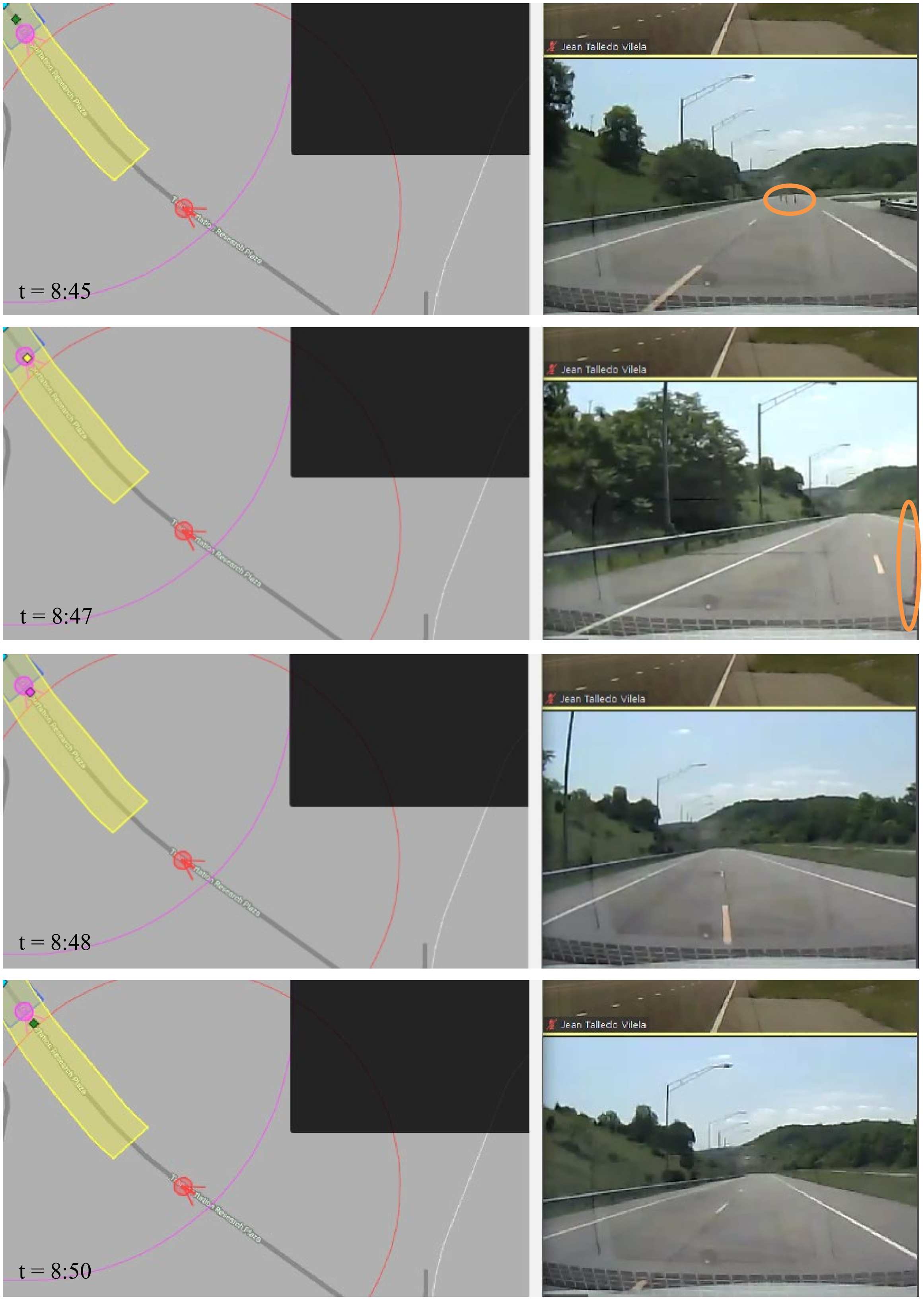
The final area to confirm is that the test bench client application demonstrates a means to display incoming information in a meaningful way. It should be noted that the application was designed to support the development and testing of the EDCM as well as show how information could be entered and viewed by the end user. A follow on engagement with TMC operators and managers was part of the next phase of the project to further develop and refine the application to meet the needs of active TMC personnel.

Three aspects of the display will be reviewed to verify they meet the design goals for the timely and accurate display of information in a manner that enhances the data available for the TMS operator.

- The map view that shows moving markers for each vehicle RM received.

- The persistent view of markers which provides a recent history of markers and vehicle activity.
- The alert notifications that indicate potential areas of interest based on an accumulation of data from multiple vehicles over a given period of time.

The primary display provides a map view with colored markers that correspond to the message types being returned from vehicles. Correlating the display and color of markers with maneuvers during the test protocol provides a means to verify the information flow from the vehicle to the end user. The following sequence of images (Figure 40) shows the generation of markers in response to three different QMs: probe, late merge and queue detection. The first frame occurs at 8:45 of the test sequence and shows the vehicle approaching the three late merge cones (circled). It also shows the first instance of the display of the green marker (upper left corner) in the frame. This indicates the receipt of a probe RM. Since the probe QM is configured to send a periodic message every five seconds, the next update of the probe marker should occur at 8:50. The next frame is at 8:47 when the yellow marker in the upper right corner is displayed. It corresponds to the first receipt of the RM triggered by the swerve maneuver designed to mimic a late merge into traffic in a work zone lane closure. Since this maneuver includes a deceleration, the vehicle speed triggers the queue detection message as the vehicle exits the geofence gating area a second later at 8:48. The last frame shows the next receipt of a probe RM which occurs at 8:50 as expected. This confirms the receipt and display of information in the TMC client application associated with three message types which correspond to the action of the vehicle on the roadway.



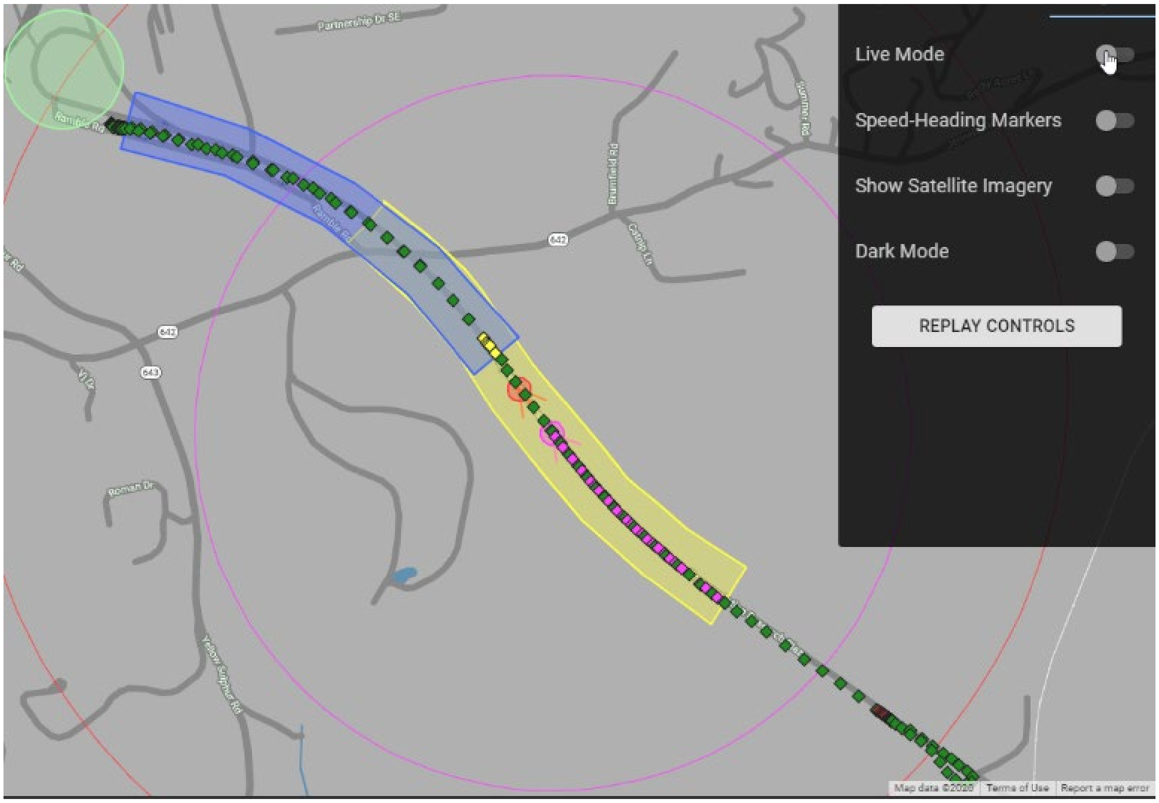

*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 40: Time Sequence of Late Merge Maneuver**

The persistent view of markers demonstrates a means to show general roadway conditions based on speed and message responses. Figure 41 shows the default view in which each vehicle only has one marker associated with it on the map at a time. At this point in the testing, there are only two connected vehicles as represented by the two green markers on the road. Since the markers are not persistent, this allows for observation of the movement of individual markers. Figure 42 shows the same vehicle configuration with live mode off showing up to the last 500 markers from all vehicles (two) on the roadway. This mode allows for a quick view of where different events are being triggered on the roadway. Here we see the yellow block of markers corresponding to the position of the late merge event, the start of the pink markers at the queue detection geofence gate and the red markers (lower right road segment) corresponding to the position of the hard brake event.


*Source: Virginia Tech Transportation Institute (VTTI)*

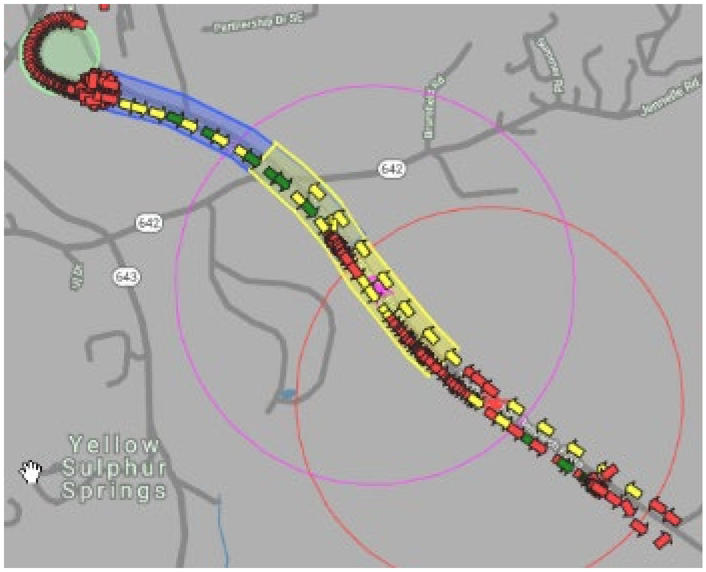
**Figure 41: EDCM Message Manager with Live Mode Display On**




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 42: EDCM Message Manager with Live Mode Display Off (Persistent View)**

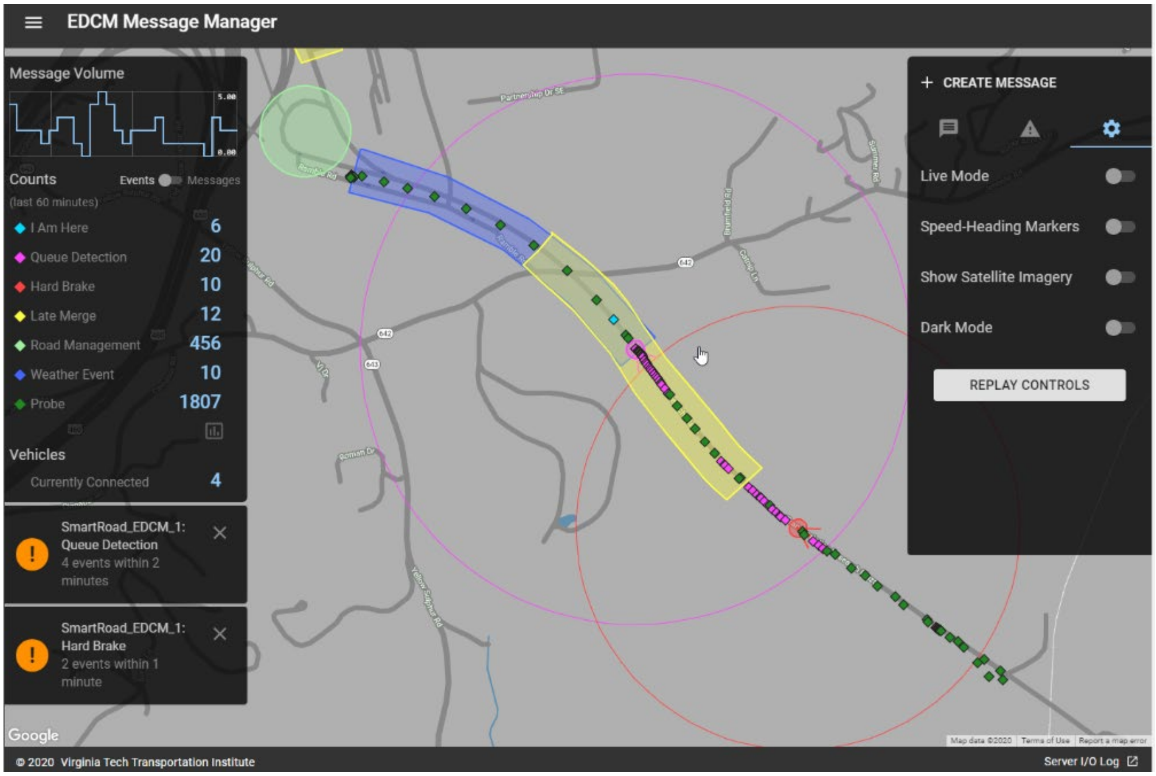
In addition to the event-based markers, speed based arrows provide a representation similar to what is provided on many mobile map applications to indicate traffic speeds. Figure 43 shows an example of this type of display. Green indicates speeds greater than 40 mph, yellow is 35 – 40 mph and red is less than 35 mph. While these thresholds may be too tight for normal TMC operation, they were selected to highlight the functionality for the given test protocol. The persistent view for event markers and speed markers provide another potential way for operators to quickly and easily gain insight into the health and activity on the roadways based on a cumulative view of the all the vehicles providing data to the server.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 43: Speed-Heading Marker Display**

The final and most targeted way that the system could provide useful information associated with specific QMs is through the use of alerts as shown in Figure 43. The following figure (Figure 44) shows two alerts that have been triggered based on the conditions assigned during the QM creation (7.1.2.2). With the persistent display, the density of the pink (queue) and the dark area (from the overlay of red and green) in the lower right (circled) provide additional indication of where the alerts are being generated. Adding icons on the roadways that correspond to these alerts, similar to what some mapping applications do for crowd sourced accident notifications, would be a way to provide further information that is easily processed by an operator.




*Source: Virginia Tech Transportation Institute (VTTI)*

**Figure 44: Alert Response with Persistent Data Display**

These three aspects of the display verify the potential to provide timely, targeted and actionable information based on the EDCM structure.

## 7.4 Conclusions

The engineering verification activity was designed to confirm the primary design features of the EDCM Project through the execution of a test protocol designed around the six use cases. The review of the data from the test execution demonstrates the potential of the EDCM concept to meet the design goals for the message structure, verifies the ability of the schema to support the transmission of the relevant data, and confirms the ability to display the data in a meaningful and useful manner to provide value to a TMC.

# 8 Queue Advisory / Queue Warning (QA/QW) System

This section describes the concept developed for queue determination and in-vehicle advisory/warning as a CV application operating under the EDCM system. As suggested in the Concept of Operations document for the Intelligent Network Flow Optimization (INFLO) Project [8,9,10], the QA/QW system informs vehicle operators of queue ahead in time for the operator to take appropriate action(s).

## 8.1 Traffic Congestion Categories

The traffic congestion is viewed differently by traffic engineers of different regions[1] depending on what they consider 'normal' operation. For the purposes of this project, four categories of traffic flow are defined as follows:

1. *Free Flow* - The flow of traffic on a given roadway segment is classified as free flow when the average speed of the vehicles in that segment is greater than 70% of the posted speed limit.
2. *Moderate Congestion* - The flow of traffic on a given road segment is classified as moderately congested when the average speed of the vehicles in that segment is between 50% and 70% of the posted speed limit.
3. *Heavy Congestion* - The flow of traffic on a given road segment is classified as heavily congested when the average speed of the vehicles in that segment is between 25% and 50% of the posted speed limit.
4. *Crawling Traffic* - The flow of traffic on a given road segment is classified as crawling when the average speed of the vehicles in that segment is below 25% of the posted speed limit.

These thresholds can be adjusted to fit local norms during implementation as appropriate.

## 8.2 Traffic Congestion Scenarios

The location where congestion occurs can be categorized by the following two general scenarios described in the following sections.

### 8.2.1 Congestion Caused by a Known Incident or at a Recurring Location

The cause and location of this type of congestion is generally known to the traffic operators. Some examples are:

- Congestion caused by a work zone / construction
- Congestion at or near an exit ramp
- Congestion at or near an entrance / on ramp
- Congestion at or near freeway interchanges

[1] Based on notes from stakeholder workshops conducted at four different districts of VDOT

### 8.2.2 Congestion Caused at a Non-recurring Location

The cause and location of this type of congestion is generally not known. Some examples are:

- Congestion caused by weather related incidents
- Congestion caused by an accident or due to a broken-down vehicle
- Congestion caused by gawkers

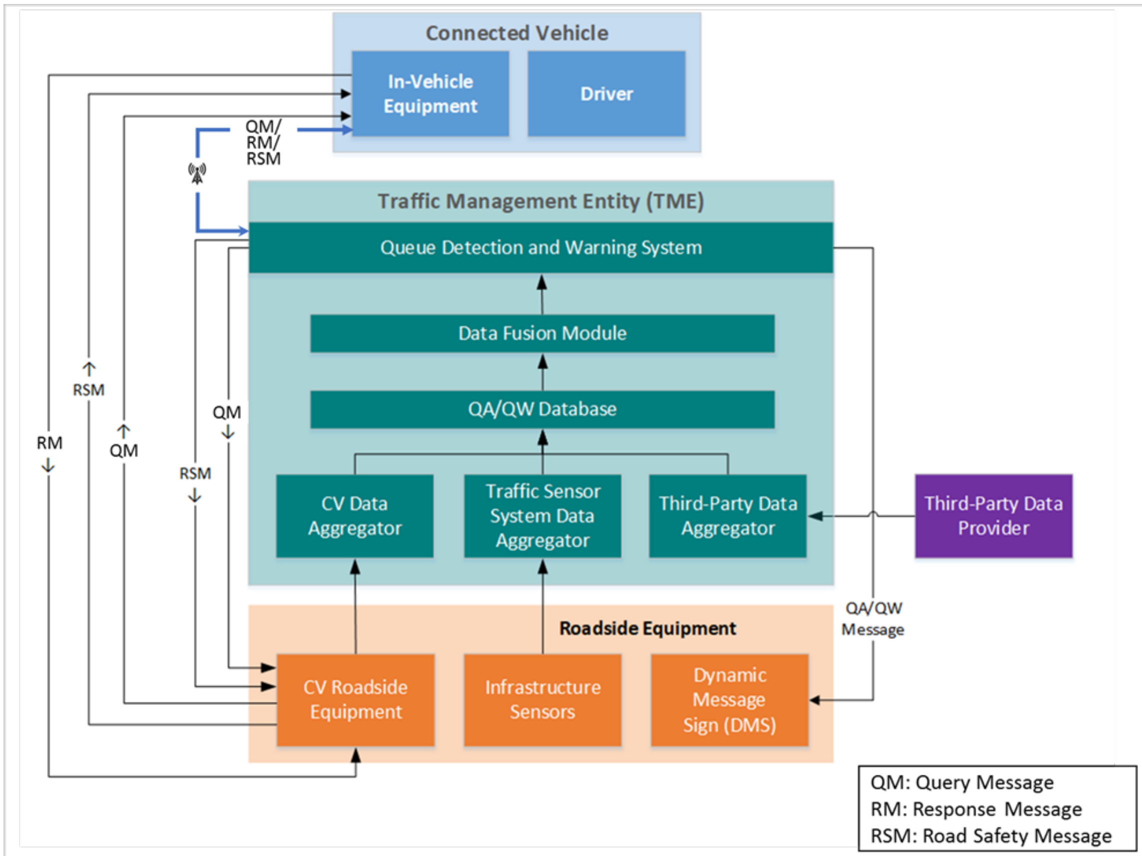
## 8.3 QA/QW Concept and System Design

The concept and design of the proposed QA/QW system was developed in cooperation with the CVPFS V2I Queue Advisory/Warning Project [11]. The analysis describes the use of CV data, Traffic Sensor data and Third-Party data in determining potential queue formation and formulating actionable information for dissemination to the CV in-vehicle application as well as to operators of unequipped vehicles.

The objective is to develop a high-level design for CV QA/QW applications for future development and testing efforts using best possible combinations of:

- Data from connected vehicles in near real-time
- Data from infrastructure (e.g., roadside sensor data)
- Data from third-party and other external sources

A conceptual diagram of data flow from various entities is shown in Figure 45. The TMC queue detection system queries CVs for relevant data, aggregates the CV data with infrastructure and third-party data to generate actionable information QA/QW specific RSMs for use by the in-vehicle application and for dynamic message signs.




*Source: Texas Transportation Institute*

**Figure 45: QA/QW System Data Flow Diagram**

### 8.3.1 Use Cases

Various use cases for QA/QW described by the CV PFS V2I Queue Advisory/Warning Project [12] are useful for developing system and in-vehicle application requirements. Various data sources used for queue detection significantly differ in:

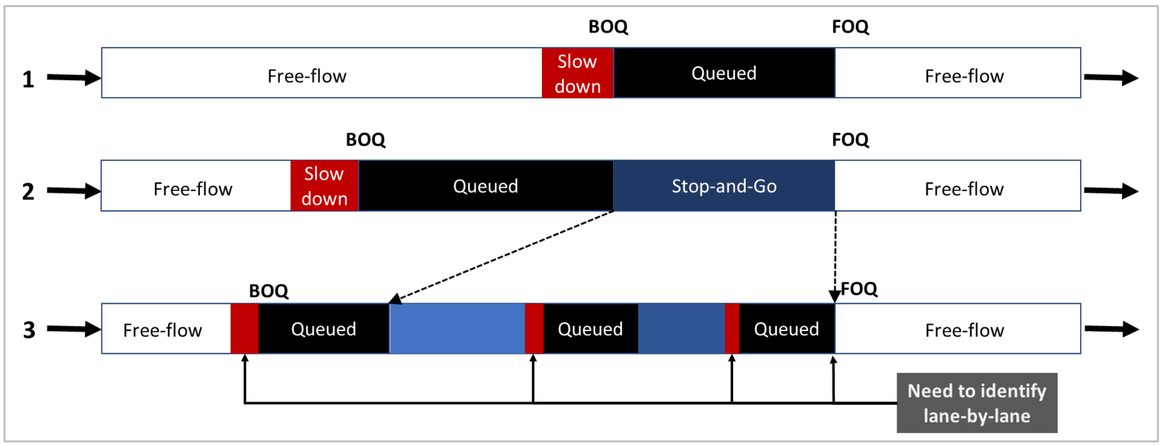
- Spatial and temporal resolution
- Latency
- Location referencing
- Queue detection accuracy
- Queue prediction capability

The following operational scenarios are considered using data from various sources based on:

- Infrastructure data only
- Third-Party data only
- Infrastructure and Third-Party data
- Infrastructure and CV data
- Infrastructure, CV and Third-Party data

### 8.3.2 Queue Formation, Queue Detection and Detection Improvement

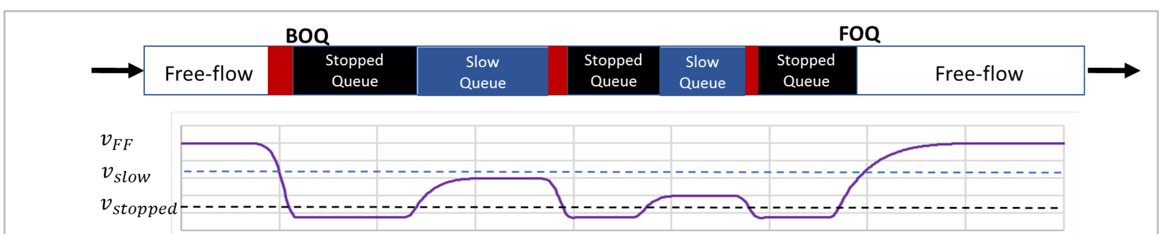
Variations in queue structure, methods for detecting queues and potential means to improve detection using CV data are described in this section to identify QA/QW system needs and develop requirements. Examples of various queue formations are shown in Figure 46. In the first illustration, free flowing traffic slows down as it approaches the back of stopped queue (BoQ). In the second illustration, the queued traffic experiences stop-and-go conditions forming a slow-moving queue before reaching the front of queue (FoQ), thus extending the BoQ further towards the upstream traffic. In the third illustration, this condition repeats several times further extending the BoQ location upstream.




*Source: Texas Transportation Institute*

**Figure 46: Various Queue Formations**

Figure 47 shows a method for queue detection using vehicle speed, speed thresholds and vehicle density. The computations shown in the figure are made by the traffic management center for their region and local needs.




Speed (v)

| Condition | Traffic state |
|---|---|
| $p_1 v_{FF} \le v$ | No Queue |
| $p_2 v_{FF} \le v < p_1 v_{FF}$ | Slow Queue |
| $v < p_2 v_{FF}$ | Stopped Queue |

**E.g., Freeway:**
$v_{FF} = 75\ mph$
$p_1 = 0.6$
$p_2 = 0.2$

| Condition | Traffic state |
|---|---|
| $45\ mph \le v$ | No Queue |
| $15\ mph \le v < 45\ mph$ | Slow Queue |
| $v < 15\ mph$ | Stopped Queue |

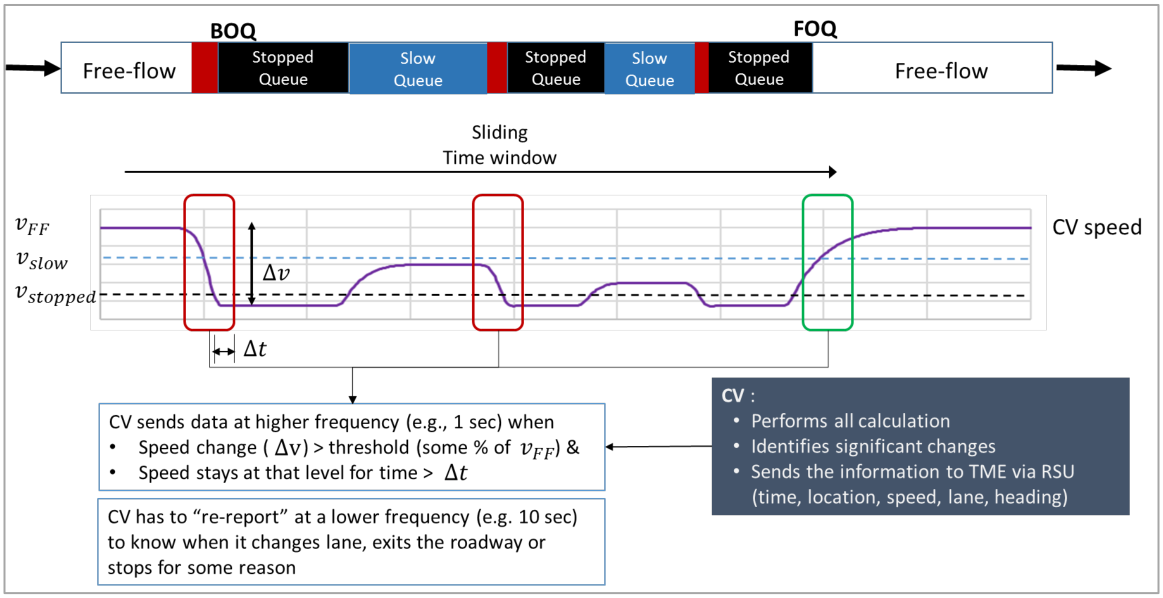
BOQ : Back of queue
FOQ : Front of Queue
$v$ : vehicle speed (mph)
$v_{FF}$ : Free-flow speed (mph)

p1 : percentage of free-flow to define slow queue threshold
p2 : of free-flow to define slow queue threshold
k : vehicle density (veh/mile)
k* : density threshold (veh/mile) used for queue detection

*Source: Texas Transportation Institute*

**Figure 47: Typical Queue Detection**

The Figure 48 shows how CV data from specific location(s) can improve queue detection by the TMC.




*Source: Texas Transportation Institute*

**Figure 48: Improvement in Queue Detection Using CV DATA**

This improvement comes when CVs provide speed, position and heading under various conditions at different frequency to the TMC. Speed change (△v) can also be reported when a TMC specified trigger threshold is crossed. This information enables the TMC to improve detection of queue formation and dissipation at a lane level. The CV can provide the information at a different rate for the TMC to determine whether the CV is approaching the BoQ or is inside the queue. It should be noted, however, that TMC requested response frequency from the CV is also dependent on the type of communication link between the TMC and the CV and available resources for processing by the CV. Though in general, CVs contain sufficient processing power to generate the requested data at the required rate.

## 8.4 Queue Advisory / Queue Warning System Behavior

### 8.4.1 Application Fidelity

Traffic delays at a work zone include delays caused by deceleration of vehicles while approaching the work zone, reduced vehicle speed through the work zone, time needed for vehicles to resume freeway speed after exiting the work zone, and vehicle queues formed at the work zone. The Year 2000 Highway Capacity Manual (HCM) [13] defines a Queue as "A line of vehicles, bicycles, or persons waiting to be served by the system in which the flow rate from the front of the queue determines the average speed within the queue. Slowly moving vehicles or people joining the rear of the queue are usually considered part of the queue. The internal queue dynamics can involve starts and stops. A faster-moving line of vehicles is often referred to as a moving queue."

Two levels of in-vehicle QA/QW application fidelity are defined based on information from the infrastructure. In defining the level of fidelity for the QA/QW system, it is assumed that the TMC has the following:

1. Road-level maps of locations that may suffer from congestion / slow down. TMC knows the location of highway splits and merges, exits, and entry ramps and posted speed limits.
2. Detailed information about the planned work zones and planned traffic interruptions, including their locations and speed limits
3. The ability to measure traffic conditions using inductive loops or other roadside sensors and knows their accuracy and latency
4. TMC requested data from EDCM-enabled connected vehicles

#### 8.4.1.1 High-Fidelity

A high-fidelity QA/QW system provides lane specific traffic congestion / queue information to approaching vehicles to inform / warn them to take appropriate action.

**Scenario:** Work zone (reduced capacity)

**Assumptions:** The TMC has lane-level work zone map data that includes:

- Cause and sub cause codes (for informational purpose as defined in (ETSI))
- Number of lanes closed and open for traffic
- Start of work zone (Reference Point in latitude/longitude/elevation)
- Heading – vehicles approaching the work zone
- Lane level map of approach lanes (~600m)
    - Note: the map of approach lanes may need to be extended based on anticipated BoQ position
- Start location of lane closure (start of taper)
- Speed limits
    - Posted speed limit of the roadway
    - Work zone speed limit (reduced speed limit).
- The following is determined by the TMC by aggregating data from CVs, roadside sensors and other sources for each lane:
    - Average speed in queue
    - BoQ position for each lane in latitude, longitude and elevation at better than 100m resolution
    - Optionally the Front of Queue (FoQ) for the in-vehicle application to determine estimated time to go through the queue
    - Shockwave speed of the queue for the in-vehicle vehicle application to determine BoQ position based on its location and speed for generating appropriate information

#### 8.4.1.2 Low-fidelity

A low-fidelity system provides road-level traffic congestion / queue information to approaching vehicles to inform / warn them to take appropriate action.

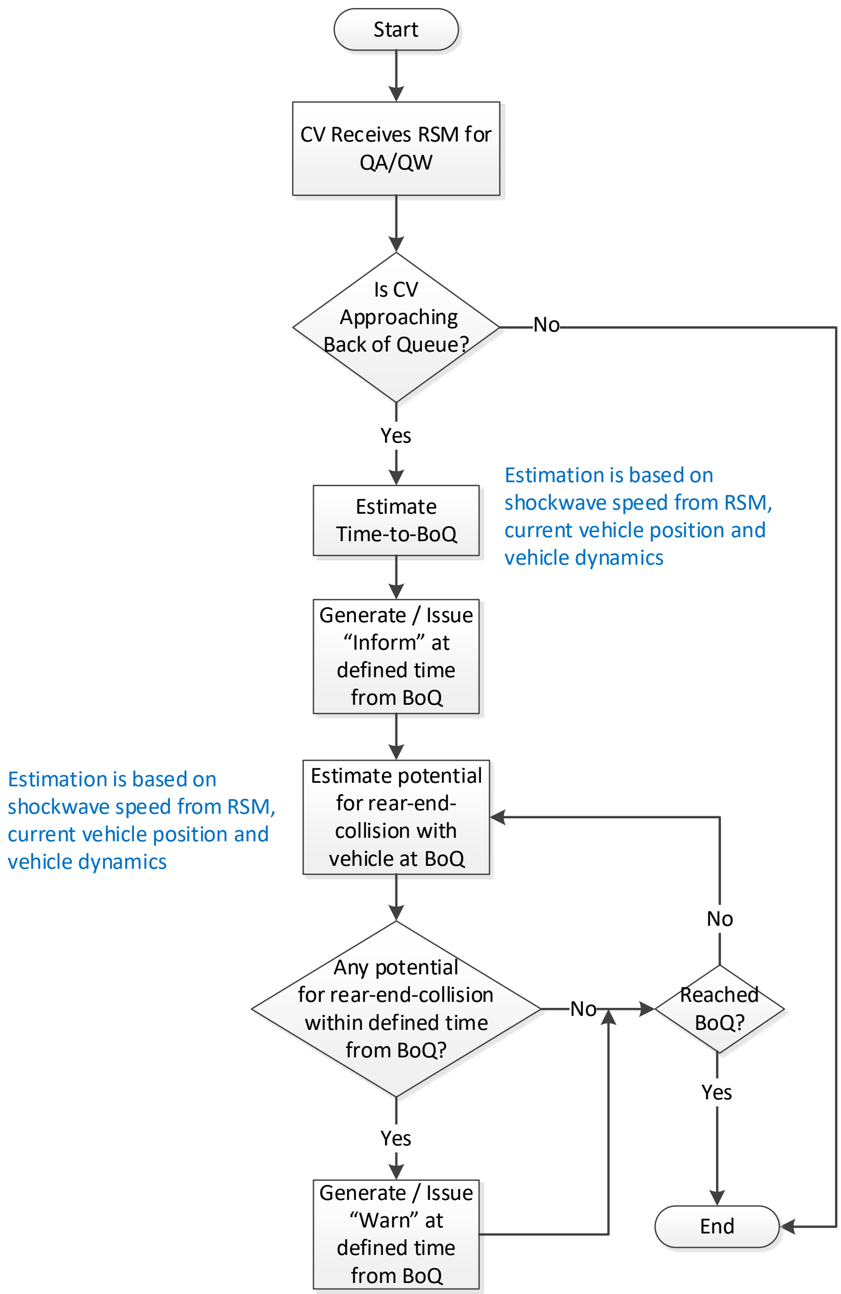
**Scenario:** Reduced roadway capacity caused due to an incidence or other anomaly

**Assumptions:** Traffic congestion leading to a queue is under unknown situation or at an unknown location and TMC has following limited information:

- Only road-level information
- Number of lanes
- Normal speed limit
- Limited information about traffic congestion from third-party sources
- The following is determined by the TMC
    - Traffic backup/congestion from external sources (e.g., police report, reported by callers, etc.)
    - Average speed of vehicles in queued / congested area (road level)
    - Back of Queue (congestion) – approximate latitude/longitude/elevation may be at 1/10th mile resolution

## 8.5 In-Vehicle QA/QW Application

Based on the level of fidelity of the information provided in the RSM from the TMC, the in-vehicle application combines the RSM information with vehicle dynamics data to inform the driver of the upcoming situation and is followed by a warning, if warranted. A high-level application flowchart for a host vehicle (HV) is shown in Figure 49.

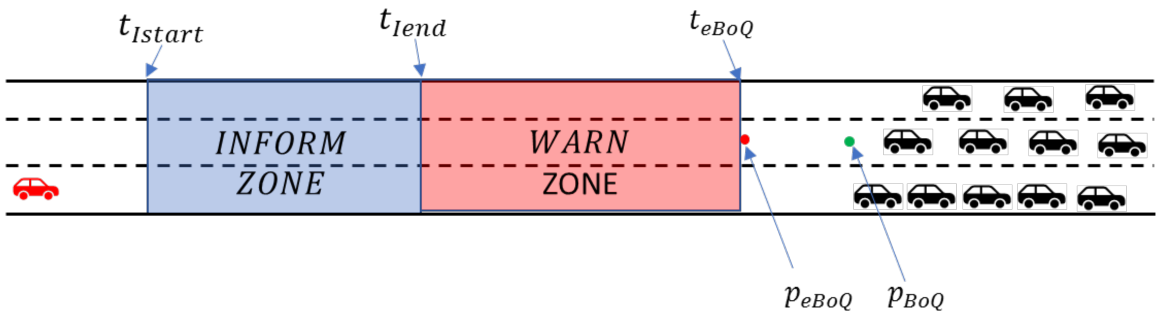



*Source: Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 49: Flow Diagram for In-Vehicle QA/QW Application**

### 8.5.1 Estimating Inform and Warn Distances from BoQ

When a vehicle approaches the BoQ, the "Inform" zone is based on a configurable pre-defined time (distance) for the host vehicle to reach the BoQ and the distance from the start of the zone. The pre-defined time to reach the BoQ considers vehicle dynamics (vehicle type, laden vs. unladen, vehicle speed, appropriate deceleration rate, etc.) based on the shockwave speed provided in the RSM and an estimate of driver perception reaction time. The start of "Warn" zone is generated based on the estimated time (distance) to the BoQ based on shockwave speed, vehicle dynamics and vehicle position (lane or road level as available). Figure 50 illustrates the "Inform" and "Warn" zone concept. In the figure, the start of "inform" zone is indicated at $t_{Istart}$ and ending at $t_{Iend}$ and the "Warn" zone starts at the end of "Inform" zone until reaching the estimated BoQ.




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 50: Estimation for "Inform" and "Warn" Zones**

The distances for the start of the "Inform" and "Warn" zones are based on time as follows:

$$d_{inf} = t_{inf} * v_{hv}$$
$$d_{warn} = t_{warn} * v_{hv}$$

Where;

$d_{inf}$ = Distance to start of inform zone. Inform Zone is a calculated segment of the roadway where the in-vehicle application issues and maintains "Inform" to the operator

$d_{warni}$ = Distance to start of warn zone. Warn zone is a calculated segment of the road where the in-vehicle application issues and maintains "Warn" to the operator

$t_{inf}$ = Configurable time threshold for start of "Inform" zone

$t_{warn}$ = Configurable time threshold for start of "Warn" zone

$v_{hv}$ = Current speed of the host vehicle (*hv*)

$t_{eBoQ}$ = Time (distance) for *hv* to reach estimated BoQ position ($P_{eBoQ}$)

$P_{BoQ}$ = Current position of BoQ

$P_{eBoQ}$ = Estimated BoQ position

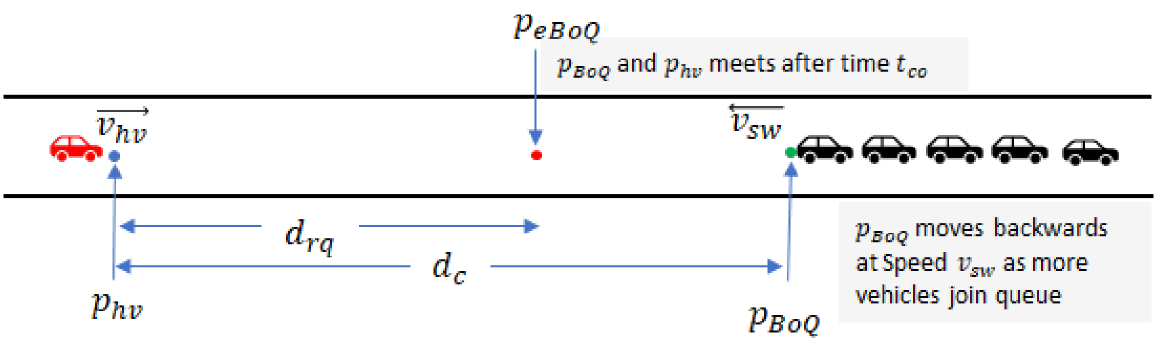
### 8.5.2 Estimating Distance to BoQ

Figure 51 illustrates a scenario where a host vehicle is approaching a region with vehicles in queued state. Since the BoQ point moves at shockwave speed of $v_{sw}$, the *hv* will reach the BoQ at $p_{eBoQ}$. As shown in Figure 51, the distance for *hv* to reach BoQ can be calculated as:

$$d_{rq} = v_{hv} * d_c / (v_{hv} + v_{sw})$$
$$t_{BoQ} = d_{rq} / v_{hv}$$

Where;

$p_{hv}$ = Current position of host vehicle
$p_{BoQ}$ = Current position of BoQ
$p_{eBoQ}$ = Estimated position of BoQ, where $p_{hv}$ meets $p_{BoQ}$
$d_c$ = Computed straight-line distance between the host vehicle and BoQ in meters
$v_{hv}$ = host vehicle speed in m/s
$v_{sw}$ = BoQ Shockwave (queue growth rate) speed in m/s. Positive for upstream and negative for downstream traffic.
$t_{BoQ}$ = Time in s when the HV meets the BoQ
$d_{rq}$ = Distance to estimated BoQ. It is negative if host vehicle speed <= BoQ shockwave speed.




*Source: Vehicle-to-Infrastructure (V2I-2) Consortium*

**Figure 51: Estimating Host Vehicle Distance to BoQ**

## 8.6 In-vehicle QA/QW Application Data

The RSM data elements required to support the in-vehicle QA/QW application are presented in the following sub-sections.

### 8.6.1 Road Safety Message (RSM) Data Elements

The J2945/4 RSM specification under development at SAE consists of two containers: A “Common” container and an “Application” container. The Common container defines data elements that are common to all V2I in-vehicle applications. The Application container defines data elements that are application specific, e.g., Reduced Speed Zone (RSZ) container, Curve container. Following this structure, a ‘Queue’ container was defined for the RSM to support the in-vehicle QA/QW application. The proposed “Queue” container uses many of the existing data elements already defined in RSM and in DSRC J2735 for other applications. Proposed elements for “Queue” container are described in the next section.

#### *8.6.1.1 Queue Container*

- **Road Surface Condition** – Current road condition e.g., dry, wet, snow etc.
- **Queue Status List** – A list of queue status for each applicable lane

- **Associated Lane –** This is an *RSMLane* data type that describes lane geometry. Lane geometry allows the in-vehicle application to determine the lane position of the host vehicle. If the laneID is set to 0, lane-level information is not available, and the application considers road geometry.
- **Queue Ahead Warning –** This Boolean data element indicates the presence or absence of a queue ahead. This element is defined in the RSM under *congestionInfo*. For lanes that do not have a queue, this value would be set to False. It provides additional information to the application for vehicle operator.
- **Traffic Flow –** Provides traffic flow type as additional information. The traffic flow types are described in Subsection 8.1.
- **BoQ Position –** Provides the estimated position of the back bumper of the last vehicle in the queue. The accuracy of this estimation depends on the level of information available to the TMC.
- **BoQ Position Update Time –** The date and time at which the BoQ position information was last updated.
- **BoQ Shockwave Speed –** This element provides the rate at which the BoQ is moving. A negative value indicates rate at which the queue is growing towards the upstream traffic.
- **Front of Queue (FoQ) Position –** Provides the estimated position of the front bumper of first vehicle in the queue.
- **FoQ Shockwave Speed –** This element provides the rate at which the FoQ is moving. A positive value indicates the rate at which the queue is dissipating towards the downstream traffic. The rate of zero indicates the FoQ is stationary.
- **Queue Confidence –** Provides average confidence (in %) of the estimation, queue speed, FoQ and BoQ positions and shockwave speeds.
- **Average Queue Speed –** Indicates the average speed of the vehicles in the queued section of the roadway.

Table 15 lists the QA/QW data elements in the 'Common' and 'Queue' Containers.

**Table 15: Data Elements for In-Vehicle QA/QW Application**

| Common Container | | | | |
|---|---|---|---|---|
| **Data Element** | **Data Type** | **Application Fidelity**<br>R = Required<br>O = Optional | | **Description** |
| | | **High** | **Low** | |
| Message Version | As 'version' in RSM.Version | R | R | For compatibility |
| Event Info | As 'eventInfo' in RSM. EventInfo | R | R | Contains information related to event start/end date/times, recurrence, cause codes |
| Event ID | As 'eventID' in DSRC.TemporaryID | R | R | Randomly assigned temporary ID for the event |
| Cause Code (ETSI) | As 'causeCode' in RSM.CauseCOde | O | O | Cause code from list of event codes defined in ETSI |
| Sub Cause Code (ETSI) | As 'causeCode' in RSM.SubCauseCode | O | O | Sub cause code from list of event codes defined in ETSI |
| Region Info | As 'RegionInfo' sequence in RSM<br>• applicableHeading<br>• speedLimit | O | O | Contains heading, reference point, speed limit, etc., for map matching purpose |
| Roadway Geometry | As 'AreaType' sequence in RSM<br>• roadwayGeometry<br>• broadRegion | O | O | To define regions of interest |

| Queue Container | | | | |
|---|---|---|---|---|
| **Data Element** | **Data Type** | **Application Fidelity**<br>R = Required<br>O = Optional | | **Description** |
| | | **High** | **Low** | |
| Surface Condition | As 'surfaceCondition' in RSM.SurfaceCondition | O | O | Enumerated values describing current road condition |
| Queue Status List | As 'QueueStatusList' | R (lane Level) | R (Road Level)<br>O (Lane Level) | Sequence of queue status for each lane |
| Associated Lane | As 'RSMGeometry' as RSMLane<br>• laneID<br>• laneGeometry | R | R | SEQUENCE (Size (1...10) of RSMLanel<br>Lane ID = 0 indicates road level geometry |
| Queue Ahead Warning | As 'queueAheadWarning' in RSM | O | O | BOOLEAN – Informative |
| Traffic Flow | As 'trafficFlowStatus' | O | O | 0 = Free flow, 1 = Moderate, 2 = Heavy, 3 = Crawling / Standstill |
| Back of Queue (BoQ) | As 'PositionBoQ' | R | R | Back of Queue position in Latitude and longitude: Micro Degrees, Elevation: 10cm |

| Queue Container | | | | |
|---|---|---|---|---|
| BoQ Shockwave Speed | As 'ShockwaveSpeedBoQ' | R | O | In +/- 0.2 m/s |
| BoQ Position Update Time | As 'BoQUpdateTime' in DSRC.DDateTime | O | O | Time when BoQ position was last updated |
| Front of Queue (FoQ) | As 'PositionBoQ' | O | O | Front of Queue position in Latitude and longitude: Micro Degrees, Elevation: 10cm |
| FoQ Shockwave Speed | As 'ShockwaveSpeedFoQ' | O | O | In +/- 0.2 m/s |
| Queue Confidence | As 'PctQueueConfidence' | O | O | List of average queue confidence level in % for all related measurements - speed, FoQ, BoQ, queue build-up rate |
| Average Queue Speed | As 'AverageQueueSpeed' | O | O | In 0.02 m/s - Average speed of vehicles in queue |

# 9 Summary

This report describes the development and implementation of Phase 1of the EDCM Project. EDCM is *a flexible messaging approach* between connected vehicles and TMCs that leverages existing connected vehicle infrastructure and communications mechanisms to identify events and road conditions that potentially impede the safety and mobility of the traveling public. EDCM advances the outcome of research conducted under the prior AMCD Project. Results from this research suggested that a flexible messaging structure may provide an effective mechanism for infrastructure applications to request data from vehicles with the aim of improving roadway operations, and the development of a dynamically reconfigurable messaging scheme based on the needs of infrastructure applications utilizing data received from vehicles would provide additional value.

The technical development of the overall EDCM system was split into two phases:

- Phase 1 laid the technical foundation for the EDCM concept.
  - Stakeholder outreach efforts were conducted to support developing an EDCM Concept of Operations [2] developed to ensure that the system design addressed a broad set of use cases
  - QA/QW System and In-Vehicle Application Requirements [12] were developed and a system architecture was defined, including the message structures, communications protocols and system interfaces required to implement the initial use cases
  - A bench-test reference system was created to verify end-to-end functionality of the proposed flexible messaging structure
- Phase 2 was not pursued due to a reprioritization of federal research funds. Results from Phase 1 suggest additional research towards adoption of EDCM is warranted.

**Stakeholder Engagement**

VDOT served as a representative for the IOOs, collaborating with the EDCM team throughout the duration of the project. The purpose of stakeholder engagement was to gather information from end users to aid in the development of the EDCM system. In particular, there were five areas in which the team sought feedback to ensure that EDCM would support the needs of the user:

- The overall EDCM architecture and Concept of Operations which enables a TMC to request vehicle data and receive responses from vehicles
- Specific support for the QA/QW and CWZ applications using the EDCM framework
- Brainstorming of additional use cases of interest to an IOO and their TMC
- Appropriateness of a reference in-vehicle system architecture to support the IOO needs for selected applications

A series of workshops were held with several VDOT operational districts covering a broad range of operational requirements. A common set of use cases that an EDCM system could facilitate was identified:

- Queue Advisory / Queue Warning
- Work Zones
- Traffic Management
- Crash & Near Crash Detection
- Weather
- Road Hazards
- Active Traffic Management

However, the priority of different use cases varied as a function of the unique roadways, geography, traffic distribution, facilities, and population density that exist within a district. The differences highlight the need for flexibility in the message structure to allow the individual TMCs to tailor the messages to gather data that is most relevant and useful for their area and at the particular moment in time (needs may change over time).

A common need identified from every workshop was to ensure the information from CV to the TMC operators would be actionable and reduce their workload by minimizing the extent to which an operator needed to analyze or interpret the raw data. Participants saw significant value in the potential of having access to more high-quality data than is currently available, in a timelier manner, and over a larger coverage area.

**EDCM System and Flexible Messaging**

The EDCM system operates within the larger CV environment, which includes supporting communication infrastructure, security protocols and privacy management techniques required for EDCM to function. It enables a TMC to request information from CVs equipped with EDCM capabilities in specified areas regarding current conditions at varying rates and time of day. EDCM-equipped CVs then provide vehicle dynamics and status data in response using a flexible messaging schema.

The EDCM System uses XML to define query and response message exchanges between a TMC and equipped vehicles. The messaging schema is based on a well-defined data dictionary, known to both the connected vehicle and the infrastructure system. In this initial design, 23 different vehicle data elements or status information can be requested in a query. A CV can be queried to get basic information such as vehicle type and position; vehicle dynamics information such as location, speed, acceleration, yaw; vehicle status information such as activation of exterior lights, ABS, traction control or stability control systems; and weather related information such as wiper position and exterior air temperature. The query message provides flexibility to request CVs to respond with a single instantaneous value or composite /averaged value, at different rates or based on distance traveled, respond when the CV is within specified geographical region (geofence) and/or when certain conditions such as vehicle type, dynamics behavior or status are met.

**Reference Bench Test for Engineering Evaluation**

To help inform the design and verify the concept and design of the EDCM, the team developed an end-to-end system to allow the creation and transmission of QM within the context of how a TMC might operate. Based on the feedback received during the stakeholder engagement, six use cases were selected to represent the conditions of interest to the stakeholder and included queue detection, hard braking, late merge (in the context of work zone monitoring), weather events, general traffic monitoring and road segment evaluation. The client application allowed QMs to be configured and sent to an OBU emulator in vehicles. The emulator was fully functional and monitored data from the vehicle CAN bus to be able to compose and transmit RMs based on the trigger and response conditions defined in the different QMs as described above. In addition to providing a means to create the QMs, the server and client applications received and displayed the information from the roadway. Within the client application, the user display provides the means to see the movement of vehicles several ways. In addition to the real time motion on a map display, a persistent marker display provides a recent historic display of RMs from the CV. Each marker is color coded to map to the QMs which are tied to a particular use case to quickly identify what event is associated with a given marker. In addition, by monitoring the associated RMs based on use case, the client can create, and display alerts based on an aggregate of data for a given message type in a given area to provide the TMC operator an alert when certain conditions exist on the road such as a slow down by 20 vehicles in a given location in the last two minutes.

To verify the design and operation, two vehicles were instrumented and used to drive a test course with scripted events corresponding to the different use cases. Live video from the vehicles was sent to the simulated TMC console to allow for simultaneous recording of the live video and the display of the client application. This data, along with log files from the vehicles and servers, was analyzed to demonstrate the complete operation of the system from QM creation to RM data display in a TMC type environment.

**QW/QW System Design and Requirements**

The QA/QW system design and in-vehicle application requirements were developed in cooperation with a companion CV PFS V2I Queue Advisory/Warning Project. The QA/QW system operates under an EDCM framework which provides a flexible messaging structure that facilitates a dynamically adjustable two-way data exchange between a TMC and enabled CVs. In QA/QW system, the TMC sends specific QMs to equipped CVs to determine the location and characteristics of speed reduction(s) caused by traffic congestion / queue formation. When specified conditions are met, CVs return RMs containing the requested data to the TMC, where the information is combined with other road-side sensor and/or third-party data. Once traffic congestion / queue formation is detected, the TMC returns the information to all CVs in the area using RSMs for the vehicle to determine appropriate action. This information may also be distributed to non-connected vehicles using Dynamic Message Signs or other conventional techniques.

Two levels of in-vehicle QA/QW applications were defined based on the resolution of information available from the infrastructure. In a high-fidelity QA/QW system, the TMC provides lane specific traffic congestion / queue information to approaching vehicles. In situations where this is not available, low-fidelity traffic congestion / queue information is provided at road-level to approaching vehicles.

**Suggested Future Work**

Phase 1 research established the EDCM system architecture and verified successful operation under controlled conditions. Future work should move implementation of the EDCM concept to real-world operation involving an actual TMC implementation in conjunction with a fleet of equipped CVs. High priority use cases such as QA/QW and CWZ should be implemented and refined to address the needs of IOO and OEM stakeholders. This effort could potentially be integrated to USDOT's Cooperative Automation Research Mobility Applications (CARMA) Cloud [14] efforts to provide a national reference implementation for TMCs.

# 10 References

[1] Advanced Messaging Concept Development (AMCD) Project Vehicle-to-Infrastructure Program: Final Report, www.its.dot.gov/index/htm, Final Report – June 30, 2017, FHWA-JPO-18-620.

[2] Event Driven Configurable Messaging (EDCM) Concept of Operations, CAMP Internal Report, Vehicle-to-Infrastructure 2 (V2I-2) Consortium, June 2020.

[3] W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures, https://www.w3.org/TR/xmlschema11-1/ .

[4] Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C Recommendation 26 November 2008, https://www.w3.org/TR/xml/ .

[5] Extensible Markup Language (XML), W3C Information and knowledge domain, https://www.w3.org/XML/ .

[6] XML Essentials, https://www.w3.org/standards/xml/core .

[7] Event Driven Configurable Messaging (EDCM), XML Schema, CAMP Internal Report, Vehicle-to-Infrastructure 2 (V2I-2) Consortium, June 2020.

[8] H. Mahmassani, H. Rekha, E. Hubbard, D. Lukasik. Concept Development and Needs Identification for Intelligent Network Flow Optimization (INFLO), Final Report – March 15, 2012 FHWA-JPO-13-011.

[9] Denny Stephens, Thomas Timcho, Theodore Smith (Battelle), Kevin Balke, Hassan Charara, Srinivasa Sunkari (TTI), Technical Report on Prototype Intelligent Network Flow Optimization (INFLO) Dynamic Speed Harmonization and Queue Warning, Final Report – June 19, 2015 FHWA-JPO-15-213.

[10] K. Balke, H. Charara, S Sunkari. Report on Dynamic Speed Harmonization and Queue Warning Algorithm Design, Final Report – February 28, 2014 FHWA-JPO-14-168.

[11] Geza Pesti, V2I Queue Advisory/Warning Applications – Concept of Operations, Texas A&M Transportation Institute 400 Harvey Mitchell Parkway South, Suite 300 College Station, TX 77845-4375, unpublished report.

[12] Queue Advisory & Queue Warning (QA/QW) System and In-Vehicle Application Requirements, CAMP Internal Report, Vehicle-to-Infrastructure 2 (V2I-2) Consortium, June 2020.

[13] Kittelson, Wayne; Ruehr, Erik. The year 2000 Highway Capacity Manual. Institute of Transportation Engineers, ITE Journal; Washington Vol. 72, Issue 6, (Jun 2002): 42-45.

[14] United States Department of Transportation (USDOT) Federal Highway Administration, CARMA Platform;

https://highways.dot.gov/research/operations/CARMA-Platform

# Appendix – A: Table of Supported Vehicle Data in QM

| Data Elements | Description | QM Trigger |
|---|---|---|
| | -- Vehicle Braking System -- | |
| abs | Anti-lock braking system | Yes |
| auxBrake | Auxiliary brake | Yes |
| brakeApplied | Brake applied | Yes |
| brakeBoost | Brake boost | Yes |
| panicBrake | Panic brake | Yes |
| vehBrakeList | List of all brake status separated by comma | |
| vehBrakeStatus | Status of each braking system | |
| | -- Vehicle Acceleration -- | |
| latAccel | Lateral acceleration | Yes |
| longAccel | Longitudinal acceleration | Yes |
| vertAccel | Vertical acceleration | Yes |
| vehAccelList | List of all accelerations separated by comma | |
| vehAccelStatus | Status of each acceleration | |
| | -- Vehicle Traction -- | |
| scs | Stability control system | Yes |
| traction | Vehicle traction | Yes |
| | -- Vehicle Speed -- | |
| speedChangeMps | Speed change in m/s | Yes |
| speedChangePct | Speed change in % | Yes |
| speedMps | Speed in m/s | Yes |
| | -- Vehicle Maneuver -- | |
| steeringWheelAngle | Steering wheel angle | Yes |
| yawRate | Yaw rate | Yes |
| | -- Vehicle Position/Location -- | |
| gfRegionEntryExitStatus | Vehicle Entry/Exit status of vehicle location in geofenced region | |
| headingDeg | Vehicle heading angle | Yes |
| tolerance | Tolerance for vehicle heading angle | Yes |
| pos3D | Vehicle position (Latitude, Longitude and Elevation) | |
| pos3DList | Vehicle position list separated by comma | |
| pos3MD | Vehicle position in micro degrees | |
| pos3MDList | Vehicle position list in micro degrees separated by comma | |
| vehPos | Vehicle position (Latitude, Longitude, Elevation, Heading and Heading Tolerance) | |
| | -- Vehicle Exterior Parameters -- | |
| extAirTempC | Exterior air temperature in Celsius | Yes |

| Data Elements | Description | QM Trigger |
|---|---|---|
| extLightList | List of exterior lights separated by comma | |
| extLightStatus | List of status of each exterior light | |
| fogLight | Fog lamp | Yes |
| hazardLight | Hazard light | Yes |
| highBeam | High beam | Yes |
| normalBeam | Normal beam | Yes |
| wiperPos | Wiper position | Yes |