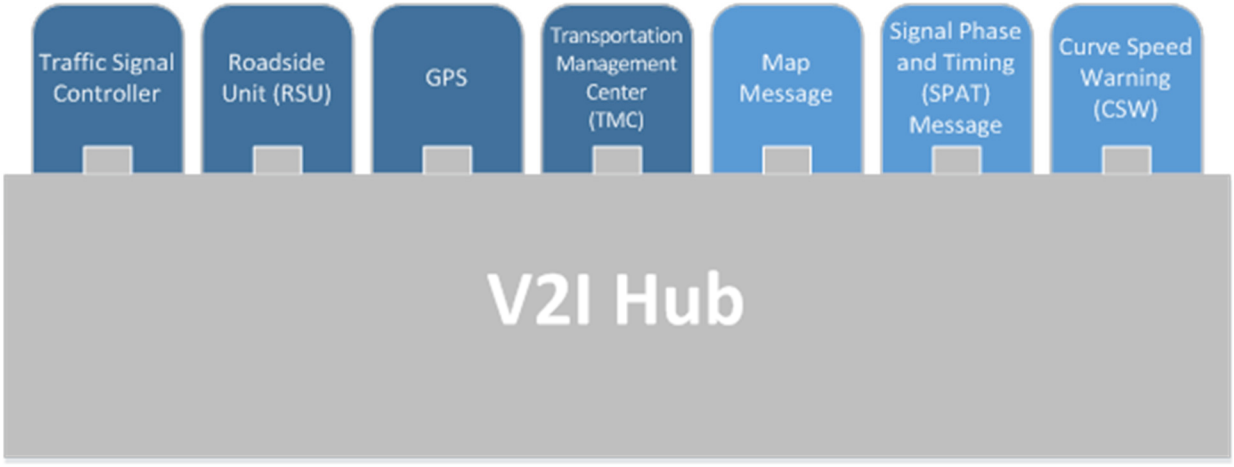
V2I Hub

Plugins

www.its.dot.gov/index.htm

Final Report – July 3, 2018
FHWA-JPO-18-643



U.S. Department of Transportation

Produced by Battelle Memorial Institute under DTFH61-12-D-00040
U.S. Department of Transportation
Office of the Assistant Secretary for Research and Technology
Joint Program Office

Notice

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA-JPO-643		

4. Title and Subtitle	5. Report Date
V2I Hub Plugins	July 3, 2018
	6. Performing Organization Code
	Battelle

7. Author(s)	8. Performing Organization Report No.
Gregory Zink, Greg Baumgardner	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Battelle 505 King Ave Columbus, OH 43201-2693	**11. Contract or Grant No.** DTFH61-12-D-00040

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Federal Highway Administration 1200 New Jersey Avenue, S.E. Washington, DC 20590	Final
	14. Sponsoring Agency Code FHWA

15. Supplementary Notes

16. Abstract

Connected Vehicle technologies help reduce the number of driving related injuries and fatalities by allowing road users to be aware of potential dangerous situations on the road. There are two main types of Connected Vehicles communications, vehicle-to-infrastructure and vehicle-to-vehicle. Vehicle-to-infrastructure communication takes place between vehicles and deployed roadside communication devices, which capture vehicle generated data while providing information pertaining to safety, mobility, and environmental conditions

This document gives descriptions and configuration details for the V2I Hub plugins created during the Integrated V2I Prototype (IVP) and V2I Reference Implementation project.

17. Keywords	18. Distribution Statement
Connected vehicle, V2V, V2I, safety, deployment, V2I Hub, IVP, SPaT, V2I Reference Implementation	Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	48	

Form DOT F 1700.7 (8-72) **Reproduction of completed page authorized**

Table of Contents

List of Tables

Executive Summary

This document contains information about the V2I Hub plugins that were created during the Integrated V2I Prototype (IVP) project and the V2I Reference Implementation project. The IVP project created a prototype software suite, called the V2I Hub, which allows software modules called plugins to create, encode, decode, and share connected vehicle data. The V2I Reference Implementation was a follow on to the IVP project, which took the prototype created, update to the latest standards and made the software a deployable solution. Several plugins were created during those projects and are listed in this document. The chapters below give descriptions of the plugins, along with information about the configuration values and any external configuration files that are needed to execute the plugins. This document is a part of the V2I Hub series of documents produced by the V2I Reference Implementation project. The rest of the documents are listed below. It is suggested to start with a review of the V2I Hub Guidebook.

- V2I Hub Guidebook
- V2I Hub Plugin Programming Guide
- V2I Hub Administration Portal User Guide
- V2I Hub Deployment Guide
- V2I Hub Software Configuration Guide

Chapter 1. Introduction

The *V2I Hub Plugins* document details the V2I Hub Plugins by providing a description of the plugin, and information on each plugin's configuration parameters, and any external configuration files that are needed to execute the plugin. For additional information on how to create a plugin for V2I Hub, refer to the *V2I Hub Plugin Programming Guide* document found in the U.S. DOT's Open Source Application Development Portal (OSADP) repository.

Chapter 2. Dedicated Short Range Communication Message Manager

Description

The Dedicated Short-Range Communication (DSRC) Message Manager, known as the DSRC Message Manager, is a plugin which sends data to a DSRC 5.9 GHz Roadside Unit (RSU) compliant with the latest RSU 4.1 specification. The DSRC Message Manager is a plugin in the V2I Hub who monitors all messages in the hub, and packages those flagged for DSRC transmission. The message sent by the DSRC Message Manager to the RSU is transmitted using the immediate forward protocol described in Appendix B, which sends a message to the RSU over a User Data Protocol (UDP) network connection. The data is then packaged and signed by the RSU and transmitted for other CV DSRC radios to use.

Configuration

The DSRC Message Manager can be configured to send messages to four different destinations. A destination is defined by a distinct IP address and port pair. This means that the DSRC Message Manager can support sending messages to four different RSUs if either the messages are the same or the RSU allows all traffic on one port. Some RSUs will only allow a single Provider Service Identifier (PSID) per immediate forwarded application. Examples of this include the Arada and Lear RSUs. These RSUs need an instance of their immediate forward application per PSID. RSUs that require multiple immediate forward applications to transmit (i.e. an application for PSID 0x8002 and a different application for PSID 0x003) will require configuration of multiple destinations to send all requested messages. The DSRC Message Manager can be configured to send messages to four different destinations. Table 1 below shows the configuration variables, description, default values for the DSRC Message Manager V2I Hub plugin.

Table 1. DSRC Message Manager Configuration Values

Key	Default Value	Description
Messages_Destination_1	{ "Messages": [{ "TmxType": "SPAT-P", "SendType": "SPAT", "PSID": "0x8002" }, { "TmxType": "MAP-P", "SendType": "MAP", "PSID": "0x8002" }] }	JSON data defining the message types and PSIDs for messages forwarded to the DSRC radio at destination 1.
Messages_Destination_2	{ "Messages": [ ] }	JSON data defining the message types and PSIDs for messages forwarded to the DSRC radio at destination 2.
Messages_Destination_3	{ "Messages": [ ] }	JSON data defining the message types and PSIDs for messages forwarded to the DSRC radio at destination 3.
Messages_Destination_4	{ "Messages": [ ] }	JSON data defining the message types and PSIDs for messages forwarded to the DSRC radio at destination 4.
Destination_1	192.168.25.10:4589	The destination UDP server(s) and port number(s) on the DSRC radio for all messages specified by Messages_Destination_1.
Destination_2	0	The destination UDP server(s) and port number(s) on the DSRC radio for all messages specified by Messages_Destination_2.
Destination_3	0	The destination UDP server(s) and port number(s) on the DSRC radio for all messages specified by Messages_Destination_3.
Destination_4	0	The destination UDP server(s) and port number(s) on the DSRC radio for all messages specified by Messages_Destination_4.
Signature	False	True of False value indicating whether the RSU should sign the messages being transmitted.

The message destination configuration values take a JSON formatted string as their input. That JSON string contains the information needed to format and send the appropriate immediate forward message to the paired destination. Messages_Destination_1 contains the messages formatting information for Destination_1 and so forth. The JSON string is described below:

- TmxType – internal message type used by the V2I Hub messages router
  - SPAT-P, MAP-P, TIM, BSM, RTCM are the standard J2735 message types used
    - SPAT-P and MAP-P specify USER encoding from the ASN.1. The transition to J2735 2015 allowed for UPER and DER encoded SPAT and MAP. All other messages were DER encoded.

- SendType – the message type string used to populate the immediate forward message
  - SPAT, MAP, TIM, BSM, RTCM are the standard J2735 message types used.
- PSID – the PSID value used to populate the immediate forward message.

Chapter 3. MAP

Description

The MAP plugin creates J2735 MAP messages in J2735 2016 format. Inputs into the plugin can be created using the U.S. DOTs ISD Message Creator. The MAP plugin accepts inputs resulting in an Extensible Markup Language (XML) format that describes the intersection with approach and egress lanes. More information on the XML MAP file format can be found in Appendix D. The ISD Message Creator (https://webapp.connectedvcs.com/isd/) is a MAP and SPaT creation tool for intersections in the U.S. DOT CV Tools Library (https://webapp.connectedvcs.com/). This online tool aids in the creation of J2735 2016 MAP files, which can be used as inputs for the V2I Hub MAP plugin. In addition to accepting XML directly, the input needed by the MAP plugin can be created by using the export tool by setting the Message Type dropdown to MAP, and copying the JavaScript Object Notation (JSON) from the MAP Data text block into a file with a .json extension. The JSON format is a lightweight data exchange format that is human readable and easy for machines to serialize and de-serialize. Alternately, the input needed by the MAP plugin can be created by using the tool to export binary (UPER Hex) data. If this option is chosen, a .txt extension should be chosen and the tight encoding option is recommended so that space conserving mode X-Y offset values are utilized rather than lengthy latititudes/longitudes.

Configuration

The MAP plugin can be configured to send different MAP files based on the action value received by the SPaT plugin. The frequency to which the MAP Plugin can transmit its output can be configured by the Frequency configuration value, in milliseconds. The default is to send MAP data at one second intervals. Table 2 shows the configuration values for the MAP plugin.

Table 2. Map Configuration Values

Key	Default Value	Description
Frequency	1000	The frequency to send the MAP message in milliseconds.
MAP_Files	{"MapFiles":[{"Action":0, "FilePath":"GID_Telegraph-Twelve_Mile_withEgress.xml"}]}	JSON data defining a list of MAP files. One MAP file for each action set specified by the TSC.

The MAP plugin uses a JSON value for the configuration key MAP_files which contains the MAP file to load associated with an action id. The action id is associated to an action plan in the Traffic Signal Controller (TSC). The SPaT plugin will transmit the current action plan that it retrieves from the TSC to the V2I Hub, where the MAP plugin will receive it. The MAP plugin will load the appropriate MAP based on the action id. If an action id is supplied by the SPaT plugin doesn't have a corresponding item in the MAP_Files list, the MAP file associated with the first action 0 will be loaded. The description for the MAP_Files JSON is below.

- Action – the action id paired with the MAP file. The action id is associated to an action plan in the TSC. When this action id is active, the corresponding MAP file will be loaded.
- FilePath – Relative file name for the MAP file. This file will need to be located in the MAP plugin directory.
    1. Default MAP plugin directory is /var/www/plugins/MAP/

Chapter 4. Signal Phase and Timing

Description

The Signal Phase and Timing (SPaT) plugin communicates with the traffic signal controller (TSC) and creates J2735 SPaT messages conforming to the J2735 2016 format. The communication with the traffic signal controller is done using NTCIP communication over UDP receiving a bit encoded byte array. The SPaT plugin only sends the necessary message to the TSC to enable and disable the sending of the UDP bit encoded stream. This plugin's main function is to receive the UDP bit encoded stream from the TSC and create the J2735 SPaT message.

Configuration

The SPaT plugin can be configured communicate and receive signal phase and timing information from a single traffic signal controller. Table 3 shows the configuration values for the SPaT plugin.

Table 3. SPaT Configuration Values

Key	Default Value	Description
Intersection_Id	1	The intersection id for SPaT generated by this plugin.
Intersection_Name	Intersection	The intersection name for SPaT generated by this plugin.
SignalGroupMapping	{"SignalGroups":[{"SignalGroupId":1,"Phase":1,"Type":"vehicle"}, {"SignalGroupId":2,"Phase":2,"Type":"vehicle"}, {"SignalGroupId":3,"Phase":3,"Type":"vehicle"}, {"SignalGroupId":4,"Phase":4,"Type":"vehicle"}, {"SignalGroupId":5,"Phase":5,"Type":"vehicle"}, {"SignalGroupId":6,"Phase":6,"Type":"vehicle"}, {"SignalGroupId":7,"Phase":7,"Type":"vehicle"}, {"SignalGroupId":8,"Phase":8,"Type":"vehicle"}, {"SignalGroupId":22,"Phase":2,"Type":"pedestrian"}, {"SignalGroupId":24,"Phase":4,"Type":"pedestrian"}, {"SignalGroupId":26,"Phase":6,"Type":"pedestrian"}, {"SignalGroupId":28,"Phase":8,"Type":"pedestrian"}]}	JSON data defining a list of active SignalGroups and phases.
Local_IP	192.168.25.20	The IPv4 address of the local computer for receiving Traffic Signal Controller Broadcast Messages.
Local_UDP_Port	6053	The local UDP port for reception of Traffic Signal Controller Broadcast Messages from the TSC.
TSC_IP	192.168.25.50	The IPv4 address of the destination Traffic Signal Controller (TSC).
TSC_Remote_SNMP_Port	501	The destination port on the Traffic Signal Controller (TSC) for SNMP NTCIP 1202 v2 communication.

The SignalGroupMapping contains a list of active phases from the signal controller and their corresponding signal groups in the J2735 SPaT message along with their phase type of either vehicle, pedestrian or overlap. Each intersection can be given a friendly name to include in the description of the SPaT message. The Intersection_Name configuration value contains the information that will be populated into that description. The Local_IP configuration value contains the local IP address of the V2I

Hub on the same network as the traffic signal controller. This is the same IP address that you would use to configure the traffic signal controller as the endpoint to send the UDP SPaT stream. Paired with the Local_IP is the Local_UDP_Port. This is the port on the local V2I Hub used to receive the UDP SPaT stream. For most of the traffic signal controllers, this is either a non-configurable value set to 6053, or a configurable value defaulted to 6053. This configuration value should not need to be modified for the SPaT plugin to operate correctly. The last two configuration values, TSC_IP and TSC_Remote_SNMP_Port, are used by the V2I Hub to send Simple Network Management Protocol (SNMP) requests. The TSC_IP address contains the IP address of the TSC. The port for which the TSC listens for SNMP commands is configured by the TSC_Remote_SNMP_Port. Again, the default for the TSC_Remote_SNMP_Port shouldn't need to be modified.

Chapter 5. RTCM

Description

The Radio Technical Commission for Maritime Services (RTCM) is an international standards organization that has created the RTCM 10402.3 RTCM Recommended Standards for Differential Global Navigation Satellite System (GNSS) Service v2.3 and RTCM 10403.3 Differential GNSS Service 3. The RTCM plugin uses the Networked Transport of RTCM via Internet Protocol (NTRIP) to receive the RTCM real time correction information, create the corresponding RTCM J2735 2016 message, and transmit it into the V2I Hub. For best results, it is suggested that an RTCM base station be within 20km of the V2I Hub. This plugin supports either RTCM v 2.3 or RTCM v 3.3 encoding, but it is up to the deployment personnel to ensure that the appropriate message types are supported by the NTRIP caster used, including Differential GPS (DGPS) and Real-time Kinematics (RTK).

Configuration

The RTCM plugin can be configured to create J2735 2016 RTCM correction messages from one NTRIP network. The network information is configured by the Endpoint IP and Endpoint Port. The network authentication information is supplied by the Username and Password configuration values. The NTRIP mountpoint to receive the RTCM stream is configured using the Mountpoint configuration values.

Table 4 contains the configuration values for the RTCM plugin. It is recommended to specify the RTCM version (2.3 or 3.3) to use for encoding and decoding purposes, although it is possible that the system can automatically determine which version through trial and error. The default values, not including the username and password, are used to retrieve RTCM 2.3 data from the state of Ohio's NTRIP network.

Table 4. RTCM Configuration Values

Key	Default Value	Description
Endpoint IP	156.63.133.118	NTRIP caster endpoint IP address
Endpoint Port	2101	NTRIP caster endpoint port
Username	username	NTRIP caster authentication username
Password	password	NTRIP caster authentication password
Mountpoint	ODOT_RTCM23	NTRIP caster mountpoint
RTCM Version	Unknown	Specify the expected RTCM message version (2.3 or 3.3) coming from the caster. Use Unknown to auto detect the version, which is done using trial and error, thus may be slow.
Route RTCM	False	Route the RTCM messages created from NTRIP internally for use by other plugins.

Chapter 6. Dynamic Message Sign

Description

The Dynamic Message Sign (DMS) plugin communicates with a DMS using National Transportation Communications for ITS Protocol (NTCIP) 1203 v3. The plugin receives requests for messages to be displayed from other plugins in the V2I Hub, and sends the appropriate message string from the configuration to the DMS for display. The plugin configures the DMS with four messages that can be called up using their index. This reduces the delay in displaying the message. The four messages can be configured using the configuration values. When the messages are changed, the updated message list is sent to the DMS. A limitation of the plugin is that if a clear display is needed, the plugin must use one of the four messages, reducing the number of messages to three.

Configuration

The DMS plugin can be configured to display four messages on a single DMS. The DMS must be connected to the same network as the V2I Hub. The DMS IP Address configuration value contains the IP address of the DMS on the network. The DMS Port is the port used by the DMS for receiving the NTCIP communications. The Message 01, Message 02, Message 03, and Message 04 configuration values contain the string used to configure the sign for message display. The default values showcase messages that worked when testing with the ADDCO DH250 sign. The “[nl]” key word in the message tells the messages to start on a new line. The configuration in Table 5 shows that the DMS is configured to display a blank page when message with id 01 is sent, 25 MPH when message id 02 is sent, Slow Down when message id 03 is sent, and Curve Ahead when message id 04 is sent. Configuration values Enable Sign Simulator, Enable DMS, Sign Sim IP Address, and Sign Sim Port are not shown or described because they are deprecated.

Table 5. DMS Configuration Values

Key	Default Value	Description
DMS IP Address	192.168.25.30	The IP address of the NTCIP Dynamic Message Sign.
DMS Port	9090	The port of the NTCIP Dynamic Message Sign.
Force Message ID	-1	Immediately activates the message ID specified, then resets back to -1.
Message 01		The text to display on the sign for ID 01 with any formatting (see NTCIP 1203).
Message 02	[jl3][pt15o0]25[nl]MPH	The text to display on the sign for ID 02 with any formatting (see NTCIP 1203).
Message 03	[jl3][pt15o0]SLOW[nl]DOWN	The text to display on the sign for ID 03 with any formatting (see NTCIP 1203).
Message 04	[jl3][pt15o0]CURVE[nl]AHEAD	The text to display on the sign for ID 04 with any formatting (see NTCIP 1203).

Chapter 7. Curve Speed Warning

Description

The Curve Speed Warning (CSW) plugin transmits information pertaining to an approaching sharp curve, monitors basic safety messages (BSM), and requests display of messages on a DMS when conditions are met. The CSW application monitors and transmits geometry leading up to the entrance of the curve using four zones. The zones are transmitted with the recommended speed for each zone using the J2735 TIM messages in J2735 2016 format. The zones for the CSW message lead up to the entrance of the curve and are numbered from one to four, with one being the closest zone to the entrance of the curve. This means a vehicle driving towards the curve will enter zone four first, then three and so on until the vehicle enters the curve. The zones are directional, meaning that the vehicle traveling towards the curve will interact with the zones, but the vehicle traveling away from the curve will not. To give CSW alerts in both directions, zones will need to be created for each direction of travel. The four zones are used by an in-vehicle application which can give different alerts or warnings to the driver depending on zone and speed. When the CSW plugin detects that a BSM is going over the recommended speed in zones three through one, the CSW plugin will sending a DMS request to display “Slow Down” on the DMS.

Configuration

Configuration for the CSW plugin is done using the following configuration values found in Table 6. The frequency of TIM transmission by the CSW plugin can be changed using the Frequency configuration values. This value is defaulted to 1000 milliseconds, or one second. The TIM geometry zones are configured using the MapFile configuration value. This value points to an xml file on the V2I Hub, in the CSW plugin’s home directory, that contains the configuration of the zones in xml format. This xml format is like that of the MAP plugin. The default location for the xml file is /var/www/plugings/CswPlugin. More information of the structure of the CSW input xml format can be found in Appendix C.

Table 6. CSW Configuration Values

Key	Default Value	Description
Frequency	1000	The frequency to send the TIM in milliseconds.
MapFile	IVP_TF_CSW.xml	The CSW xml input file.
Snap Interval	300	The interval in milliseconds to keep a vehicle within a zone before allowing it to transition out of all zones.
Vehicle Timeout	2000	Timeout in milliseconds when a vehicle is removed from all zones if a BSM has not been received.

Snap Interval and Vehicle Timeout config are used by the vehicle monitoring algorithm portion of the CSW plugin. The CSW will timeout a vehicle that it has not received an update from after the value in Vehicle Timeout, which is defaulted to two seconds. The snap interval is used to reduce jitter in the GPS position, and requires that 300 milliseconds elapse before transitioning a vehicle out of a zone.

Chapter 8. Message Receiver

Description

Most DSRC RSUs have an application that forwards certain messages from the radio to a configured device. The stored data is sent in J2735 2016 format, and the most common data type sent is the BSM. The Message Receiver Plugin was written to take the forwarded J2735 2016 messages such as BSM from an RSU and send it into the V2I Hub system for other applications, like the CSW plugin, to use in its processing. The Message Receiver can work with any device that forwards J2735 messages or V2I Hub JSON encoded messages over UDP, including simulation software or future communication devices which support the same standard. Indeed, this plugin is used to receive the simulated BSMs and SRMs coming from the V2I Hub Vehicle Simulation Tool. See the V2I Hub Vehicle Simulation Tool User Guide for more details.

Configuration

The configuration of the Message Receiver plugin requires only the IP address for the V2I Hub device receiving the messages to be set for the IP configuration value, along with the Port to open on that V2I Hub device. The plugin is already set up to receive J2735 2016 messages from any number of 4.1 compliant RSUs or simulation software, but those RSUs and the simulation software must separately be configured with these IP and Port values to route the incoming messages directly to the V2I Hub Message Receiver Plugin. The RouteDSRC flag is used for simulation testing in the lab and allows BSMs generated by other plugins to also be sent out of the RSU. This configuration values should not be changed and kept at false for most tests and for all deployments. There are also three configuration parameters for accepting BSM and/or SRM messages from the V2I Hub Vehicle Simulation Tool. Incoming BSMs from the simulation software can also be used to create simulated GPS locations.

Table 7. Message Receiver Configuration Values

Key	Default Value	Description
IP	127.0.0.1	IP address for the incoming message network connection. The default value only accepts connection from the localhost, and must be changed to receive from external subnets.
Port	26789	Port for the RSU network connection.
RouteDSRC	false	Set flag to true for the message to be sent out the DSRC radio by the DSRC Message Manager. This flag would only be used in testing or simulation. For example, having an RSU broadcast BSMs from simulation software.
EnableSimulatedBSM	true	Accept and route incoming BSM messages from a V2I Hub Vehicle Simulation Tool.
EnableSimulatedSRM	true	Accept and route incoming SRM messages from a V2I Hub Vehicle Simulation Tool.
EnableSimulatedLocation	true	Accept and route incoming GPS location messages from a V2I Hub Vehicle Simulation Tool.

Chapter 9. Operational Data Environment

Description

The Operational Data Environment (ODE) is a scalable data acquisition and distribution solution created by ITS JPO. The ODE plugin allows transmission of BSMs received in the V2I Hub to the deployed ODE solution. In a deployed V2I Hub system, the BSM would be received by the RSU and forwarded to the V2I Hub. Once in the V2I Hub, the BSM would be received by the ODE plugin and then sent to the ODE using the set configuration. The ODE solution can also create TIM information based on the data it receives. The ODE plugin will query the ODE for TIM messages for the deployed V2I Hub's location, using the latitude and longitude obtained by GPS or set in configuration during deployment. Any TIM messages retrieved by the ODE plugin will be packaged and sent through the V2I Hub and ultimately out the RSU.

Configuration

The ODE Plugin Receiver can be configured to send BSM messages and receive TIM messages from one ODE instance using the configuration in Table 8. The ODEIP contains the configuration value for the IP address for the deployed ODE, and the ODEPort is the port used for transmission

Table 8. ODE Configuration Values

Key	Default Value	Description
ODEIP	192.168.55.43	IP address for the ODE network connection.
ODEPort	26789	Port for the ODE network connection.

Chapter 10. Command Plugin

Description

The V2I Hub Command Plugin is used solely for communication to and from V2I Hub Admin Portal connections. Using web socket connections from any number of browsers, the Command Plugin writes periodic updates to each web portal for display, as well as handling incoming requests for change from the administrative users of the web portal. The options include software changes, configuration changes and message and event displays. Refer to the V2I Hub Admin Portal Guide for more details.

Configuration

The configuration values in Table 9 are used to set up security and performance of the V2I Hub Command Plugin. The default mode of operation is with a Secure Socket Layer (SSL) around the web socket connection. This secured web socket mode should always be used in deployment, and the appropriate certificates must be installed to the SSLPath location. Because messages are processed in batch by the Command Plugin, the number of milliseconds between each cycle will affect both the performance of the plugin and the display update intervals for each of the web portals. The DownloadPath is a location where installable components such as new or updated plugins contained in a .zip are temporarily extracted to. Therefore, the Linux user running the Command Plugin process must have write permissions to this directory for the feature to work properly.

Table 9. Command Plugin Configuration Values

Key	Default Value	Description
SSLEnabled	True	Enable secure connection using SSL
SSLPath	/var/www/plugins/.ssl	The path to the directory containing the SSL key and certificate files. The files in this directory should be secured from outside access.
SleepMS	100	The length in milliseconds to sleep between processing all messages.
EventRowLimit	50	The maximum number of rows returned for the initial Event Log query
DownloadPath	/var/www/download	The path to the directory where downloaded files will be saved.

Appendix A. Acronyms

ASC	Actuated Signal Controller
CV	Connected Vehicle
BSM	Basic Safety Message
CSW	Curve Speed Warning
DMS	Dynamic Message Sign
DSRC	Dedicated Short-Range Communications
FCC	Federal Communications Commission
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
JSON	Java Script Object Notation
ISD	SDC/SDW Intersection Situation Data
IVP	Integrated V2I Prototype
NTCIP	National Transportation Communications for Intelligent Transportation System Protocol
NTRIP	Networked Transport RTCM via Internet Protocol
OBU	On-Board Unit
ODE	Operational Data Environment
PSID	Provider Service Identifier
RSU	Roadside Unit
RTCM	Radio Technical Commission for Maritime Services
SNMP	Simple Network Management Protocol
SPaT	Signal, Phase, and Timing

SSL	Secure Socket Layer
TIM	Traffic Incident Message
TSC	Traffic Signal Controller
U.S. DOT	United States Department of Transportation
UDP	User Data Protocol
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
XML	Extensible Markup Language

Appendix B. Active File Format

The active file format listed below is from the *DSRC Roadside Unit (RSU) Specifications Document v4.1* Appendix C. This has been included in its entirety as a reference.

```
# Message File Format

# Modified Date: 04/10/2014

# Version: 0.7

Version=0.7

# Message Dispatch Items

#

# All line beginning with # shall be removed in file sent to radio

#

# Message Type

# Values: SPAT, MAP, TIM, (other message types)

Type=<Type>

#

# Message PSID as a 2 Byte Hex value (e.g. 0x8003)

PSID=<PSID>

#

# Message Priority in the range of 0 (lowest) through 7

Priority=<priority>

#

# Transmission Channel Mode

# Allowed values: CONT, ALT

TxMode=<txmode>
```

Allowed values: 172, CCH, SCH(note: “CCH” refers to DSRC Channel 178 and SCH refers to the

#operator configured DSRC Service Channel)

TxChannel=<channel>

#

Transmission Broadcast Interval in Seconds

Allowed values: 0 for Immediate-Forwarding, 1 to 5 for Store-and-Repeat

TxInterval=<txinterval>

#

Message Delivery (broadcast) start time (UTC date and time) in the form:

#"mm/dd/yyyy, hh:mm”

Leave value blank if Immediate Forward mode

DeliveryStart=<mm/dd/yyyy, hh:mm>

#

Message Delivery (broadcast) stop time (UTC date and time) in the form:

#"mm/dd/yyyy, hh:mm”

Leave value blank if Immediate Forward mode

DeliveryStop=<mm/dd/yyyy, hh:mm>

#

Message Signature/Encryption

Signature=<True\False>

Encryption=<True\False>

#

Message Payload (encoded according to J2735 or other definition)

Payload=<DSRC message payload>

Appendix C. CSW XML File Format

The Curve Speed Warning (CSW) plugin uses an XML input file to create the geometry sent in the SAE J2735-2016 Traveler Information Message (TIM) for the curve. The input file contains the speed of the curve, the four (4) zones with their reference point, offsets to create the center line for the zones, and the zone width. Currently, the CSW plugin supports zones containing two (2) points, which create a straight line. The zones are referenced as Zone 1 through Zone 4, with Zone 1 being the closest to the entrance of the curve and Zone 4 the furthest. In the file, Zone 1 is the first region in the region list. As a vehicle approaches the curve, the vehicle will enter Zone 4, Zone 3, Zone 2, and finally, Zone 1. A sample XML file is included along with a description of each node.

Sample Input File

```
<?xml version="1.0" encoding="UTF-8"?>
<Curve xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance">
  <Version>1</Version>
  <CurveID>125</CurveID>
  <SpeedLimit>8.491</SpeedLimit>
  <Approaches>
    <Approach>
      <Regions>
        <Region>
          <ReferencePoint>
            <Latitude>38.9556187319798</Latitude>
            <Longitude>-77.1504409878302</Longitude>
            <Elevation>0</Elevation>
          </ReferencePoint>
          <Width>335</Width>
          <DirectionOfUse>Forward</DirectionOfUse>
          <Nodes>
            <Node>
              <Eastern>0</Eastern>
              <Northern>0</Northern>
              <Elevation>0</Elevation>
            </Node>
            <Node>
              <Eastern>141</Eastern>
              <Northern>-206</Northern>
              <Elevation>0</Elevation>
            </Node>
          </Nodes>
        </Region>
        <Region>
          <ReferencePoint>
            <Latitude>38.9554328674441</Latitude>
            <Longitude>-77.150277082612</Longitude>
            <Elevation>0</Elevation>
          </ReferencePoint>
          <Width>335</Width>
```

```
          <DirectionOfUse>Forward</DirectionOfUse>
          <Nodes>
            <Node>
              <Eastern>0</Eastern>
              <Northern>0</Northern>
              <Elevation>0</Elevation>
            </Node>
            <Node>
              <Eastern>173</Eastern>
              <Northern>-181</Northern>
              <Elevation>0</Elevation>
            </Node>
          </Nodes>
        </Region>
        <Region>
          <ReferencePoint>
            <Latitude>38.9552698871225</Latitude>
            <Longitude>-77.150076690332</Longitude>
            <Elevation>0</Elevation>
          </ReferencePoint>
          <Width>335</Width>
          <DirectionOfUse>Forward</DirectionOfUse>
          <Nodes>
            <Node>
              <Eastern>0</Eastern>
              <Northern>0</Northern>
              <Elevation>0</Elevation>
            </Node>
            <Node>
              <Eastern>256</Eastern>
              <Northern>-155</Northern>
              <Elevation>0</Elevation>
            </Node>
          </Nodes>
        </Region>
        <Region>
          <ReferencePoint>
            <Latitude>38.9551299660347</Latitude>
            <Longitude>-77.1497799675418</Longitude>
            <Elevation>0</Elevation>
          </ReferencePoint>
          <Width>335</Width>
          <DirectionOfUse>Forward</DirectionOfUse>
          <Nodes>
            <Node>
              <Eastern>0</Eastern>
              <Northern>0</Northern>
              <Elevation>0</Elevation>
            </Node>
            <Node>
              <Eastern>382</Eastern>
              <Northern>-138</Northern>
              <Elevation>0</Elevation>
            </Node>
          </Nodes>
        </Region>
      </Regions>
    </Approach>
  </Approaches>
```

```
</Curve>
```

Curve Node

```
<?xml version="1.0" encoding="utf-8"?>
<Curve xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <Version>1</Version> ← Version of the curve file
      <CurveID>125</CurveID> ← Curve ID
      <SpeedLimit>8.491</SpeedLimit> ← Speed Limit of Curve in m/s
      <Approaches> </Approaches> ← List of Approach Nodes
</Curve>
```

Approach Node

```
<Approach>
      <Regions> </Regions> ← List of Region Nodes
</Approach>
```

Region Node

Each region of the curve XML is information for one of the zones. Each Region should be formatted like the one below, and contain a reference point, width, and node list. Each node contains offset in decimeters from the reference point to create the center line for the region (or zone).

```
<Region>
      <ReferencePoint> ← Used in creating the center line for the region with the nodes
            <Latitude>38.9556187319798</Latitude> ← Latitude in decimal degrees
            <Longitude>-77.1504409878302</Longitude> ← Longitude in decimal
degrees
            <Elevation>0</Elevation> ← Elevation in meters (optional, set to 0 if not used)
      </ReferencePoint>
      <Width>335</Width> ← Width in decimeters (ex 335 = 3.35 meters)
      <DirectionOfUse>Forward</DirectionOfUse> ← Direction of use should always be
forward
      <Nodes>
            <Node>
                  <Eastern>0</Eastern> ← Eastern offset in centimeters for node 1
                  <Northern>0</Northern> ← Northern offset in centimeters for node 1
                  <Elevation>0</Elevation> ← Elevation offset in centimeters for node 1
            </Node>
            <Node>
                  <Eastern>141</Eastern> ← Eastern offset in centimeters for node 2
```

```
                <Northern>-206</Northern> ← Northern offset in centimeters for node 2
                <Elevation>0</Elevation> ← Elevation offset in centimeters for node 2
            </Node>
        </Nodes>
</Region>
```

Appendix D. MAP XML File Format

The MAP plugin uses an XML input file as one option to create the SAE J2735-2016 MAP message for the intersection. The input file contains a reference point for the intersection, and approach and egress lanes. Each lane is defined by a set of nodes, which have offsets. The offset for the first node is from the reference lane, and subsequent nodes are from the previous node. A sample input file is shown below, and a description of the file follows. Each approach lane needs a connection to an egress lane with the signal group from the related SPaT message controlling the maneuver that connects the lanes.

Sample Input File

```
<J2735.GID.blob>
  <Version>1</Version>
  <IntersectionID>1001</IntersectionID>
  <Elevation>false</Elevation>
  <Resolution>centimeter</Resolution>
  <Geometry>
    <ReferencePoint>
      <Latitude>38.954997</Latitude>
      <Longitude>-77.149386</Longitude>
    </ReferencePoint>
    <Approach>
      <Lane Number="2">
        <Type>Vehicle</Type>
        <Attributes>14</Attributes>
        <Width>305</Width>
        <Nodes>
          <Node Number="1">
            <Eastern>-1540</Eastern>
            <Northern>320</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="2">
            <Eastern>-1020</Eastern>
            <Northern>500</Northern>
            <Elevation>0</Elevation>
          </Node>
        </Nodes>
        <Connections>
          <Connection>
            <LaneNumber>106</LaneNumber>
            <Maneuver>1</Maneuver>
            <SignalGroup>2</SignalGroup>
          </Connection>
      </Lane>
      <Lane Number="6">
        <Type>Vehicle</Type>
        <Attributes>2</Attributes>
        <Width>305</Width>
        <Nodes>
          <Node Number="1">
```

```
            <Eastern>1450</Eastern>
            <Northern>-300</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="2">
            <Eastern>1550</Eastern>
            <Northern>-270</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="3">
            <Eastern>1520</Eastern>
            <Northern>-170</Northern>
            <Elevation>0</Elevation>
          </Node>
        </Nodes>
        <Connections>
          <Connection>
            <LaneNumber>107</LaneNumber>
            <Maneuver>1</Maneuver>
            <SignalGroup>6</SignalGroup>
          </Connection>
        </Connections>
      </Lane>
    </Approach>
    <Egress>
      <Lane Number="106">
        <Type>Vehicle</Type>
        <Attributes>2</Attributes>
        <Width>305</Width>
        <Nodes>
          <Node Number="1">
            <Eastern>1450</Eastern>
            <Northern>-650</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="2">
            <Eastern>1550</Eastern>
            <Northern>-270</Northern>
            <Elevation>0</Elevation>
          </Node>
        </Nodes>
      </Lane>
      <Lane Number="107">
        <Type>Vehicle</Type>
        <Attributes>2</Attributes>
        <Width>305</Width>
        <Nodes>
          <Node Number="1">
            <Eastern>-1420</Eastern>
            <Northern>690</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="2">
            <Eastern>-1030</Eastern>
            <Northern>490</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="3">
            <Eastern>-850</Eastern>
```

```
            <Northern>380</Northern>
            <Elevation>0</Elevation>
          </Node>
          <Node Number="4">
            <Eastern>-1100</Eastern>
            <Northern>530</Northern>
            <Elevation>0</Elevation>
          </Node>
        </Nodes>
      </Lane>
    </Egress>
  </Geometry>
</J2735.GID.blob>
```

J2735.GID.blob Node

```
<J2735.GID.blob>
      <Version>1</Version>   ← Version of the curve file
      <IntersectionID>1001</IntersectionID>   ← Intersection ID (needs to match SPAT)
      <Elevation>false</Elevation>   ← Boolean indicating if elevation is included
      <Resolution>centimeter</Resolution>   ← Node and elevation offset resolution
      <Geometry>   ← Intersection geometry
      </Geometry>
</J2735.GID.blob>
```

Geometry Node

```
<Geometry>
      <ReferencePoint>   ← Reference point location
            <Latitude>38.954997</Latitude>   ← Latitude in decimal degrees
            <Longitude>-77.149386</Longitude>   ← Longitude in decimal degrees
      </ReferencePoint>
      <Approach />   ← List of approach lanes
      <Egress />   ← List of egress lanes
</Geometry>
```

Approach and Egress Node

Approach and Egress have the same format. Lanes in the egress node will not have the connections node.

```
<Approach>
      <Lane Number="2">
```

```
            <Type>Vehicle</Type>
            <Attributes>14</Attributes>
            <Width>305</Width>
            <Nodes />
</Approach>
```

Nodes Node

```
<Nodes>
      <Node>
            <Eastern>0</Eastern> ← Eastern offset in centimeters for node 1
            <Northern>0</Northern> ← Northern offset in centimeters for node 1
            <Elevation>0</Elevation> ← Elevation offset in centimeters for node 1
      </Node>
      <Connections>   ← List of connections (Approach Only)
            <Connection>
                  <LaneNumber>107</LaneNumber>   ← Connecting lane number
                  <Maneuver>1</Maneuver>   ← Connecting maneuver bitmapped int
                  <SignalGroup>6</SignalGroup>   ← Connecting signal group
            </Connection>
      </Connections>
<Nodes>
```

Appendix E. References

1. U.S. DOT. (2009, September 25). IEEE 1609 - Family of Standards for Wireless Access in Vehicular Environments (WAVE). In *United States Department of Transportation*. Retrieved from https://www.standards.its.dot.gov/factsheets/factsheet/80

2. National Marine Electronics Association. (2008, November). NMEA 0183 Standard. In *National Marine Electronics Association*. Retrieved from https://www.nmea.org/content/nmea_standards/nmea_0183_v_410.asp

3. U.S. DOT. (2009, September 18). NTCIP 1202 - Object Definitions for Actuated Traffic Signal Controller Units. In *United States Department of Transportation*. Retrieved from https://www.standards.its.dot.gov/Factsheets/Factsheet/22

4. U.S. DOT. (2009, September 18). NTCIP 1203 - Object Definitions for Dynamic Message Signs (DMS). In *United States Department of Transportation*. Retrieved from https://www.standards.its.dot.gov/Factsheets/Factsheet/23

5. U.S. DOT. (2016, October 31). DSRC Roadside Unit (RSU) Specifications Document v4.1. In *United States Department of Transportation*. Retrieved from http://www.fdot.gov/traffic/Doc_Library/PDF/USDOT%20RSU%20Specification%204%201_Final_R1.pdf

6. Radio Technical Commission for Maritime Services. (n.d.). RTCM 10402.3 RTCM Recommended Standards for Differential GNSS (Global Navigation Satellite Systems) Service, Version 2.3. In *Radio Technical Commission for Maritime Services*. Retrieved from http://www.rtcm.org/about.html

7. U.S. DOT. (2009, September 25). SAE J2735 - Dedicated Short Range Communications (DSRC) Message Set Dictionary. In *United States Department of Transportation*. Retrieved from https://www.standards.its.dot.gov/Factsheets/Factsheet/71

8. U.S. DOT. (2016, March). SAE J2945/1 On-board Minimum Performance Requirements for V2V Safety Communications. In *United States Department of Transportation*. Retrieved from https://www.standards.its.dot.gov/Standard/531

U.S. Department of Transportation
ITS Joint Program Office – HOIT
1200 New Jersey Avenue, SE
Washington, DC 20590

Toll-Free “Help Line” 866-367-7487

www.its.dot.gov

FHWA-JPO-18-643

U.S. Department of Transportation