Integrated V2I Prototype (IVP): V2I Hub

Tab 1: Overview

In order to bring infrastructure components into the Connected Vehicle architecture, you need software that will facilitate the exchange of data in a format that can be understood by both vehicles and infrastructure devices The Integrated Vehicle-to-Infrastructure Prototype (IVP), called V2I Hub, takes in data from vehicles via Basic Safety Messages (BSM) in a Society of Automotive Engineers (SAE) standard format and translates the data to a National Transportation Communications for ITS Protocol (NTCIP) that infrastructure components can understand. And vice versa. It translates Signal Phase and Timing (SPaT) data from NTCIP to SAE and sends it to the Roadside Unit (RSU) for broadcast to mobile devices, including vehicles.

V2I Hub is a message handler that acts as a translator and data aggregator/disseminator for infrastructure components of a connected vehicle deployment.

V2I Hub was developed to support jurisdictions in deploying Vehicle-to-Infrastructure (V2I) technology by reducing integration efforts and issues.

* V2I Hub is a software platform that enables connected vehicles to talk to existing traffic management hardware and systems, such as traffic signal controllers, Transportation Management Centers, pedestrian and vehicle detection systems, road weather sensors, and dynamic message signs.
* V2I Hub simplifies integration by translating communication between different standards and protocols.
* Using a modular design, software plugins enable efficient connections to new hardware, custom connections to Transportation Management Centers, and Connected Vehicle (CV) Safety Apps to run on roadside equipment.

Tab 2: Description

The V2I Hub software (formerly known as IVP) will bridge the gap between infrastructure messages and vehicle messages and allow the two systems to effectively communicate. This software is designed to be used by jurisdictions deploying connected vehicle systems. By initializing the system with standard data such as intersection geometry and the correlation between signal phases and lanes, this software will be able to form and transmit Signal Phase and Timing (SPaT) messages that are key to connected vehicle applications such as red light violation warning. Through the use of Traveler Information Messages (TIM), the system can also develop and send infrastructure information about advisory speeds, attributes of physical elements such as bridge heights, and other data that can be used by applications such as curve speed warning and over-height warnings.

The V2I Hub system reduces time needed to create and deploy a roadside based V2I system. The V2I Hub system contains a suite of plugins that are built to handle specific functionality. The output of these plugins will vary, but any plugin that communicates externally will produce a message from the J2735-201509 messages set. Plugins can request to receive data that is being produced by other plugins in the system. For example, a location plugin can create a location message that is then received by the MAP plugin for use it in its processing. Below are a list of plugins and the messages they produce that are included in the V2I Hub system.

* AradaRecieverPlugin – Created for the Arada Commando RSU, this plugin will receive J2735 BSM messages and inject them back into the system for use by other plugins.
* CSWPlugin – The Curve Speed Warning Plugin will monitor J2735 BSM messages at a curve, and send a message to a dynamic message sign when it detects that a vehicle is approaching a curve too fast. The CSW plugin also produces a J2735 TIM message containing the approach zones for the curve to be used by a CSW in-vehicle CV application.
* DMSPlugin – The Dynamic Message Sign (DMS) Plugin will receive messages from other plugins and translate the information to NTCIP 1203 for display on a DMS.
* MAP Plugin – Produces intersection geometry in J2735 MAP format.
* SPAT Plugin – Communicates with a traffic signal controller (TSC) using NTCIP 1202, and creates a J2735 SPAT Message.
* DSRCImmediateForwardPlugin – Sends all J2735 traffic to the RSU for transmission out the DSRC radio.
* LocationPlugin – Communicates with GPS devices producing NMEA GP\* sentences and forwards location information to the V2I Hub system.
* RTCMPlugin – Communicates with a NTRIP network to create J2735 RTCM position correction messages.

V2I Hub is a communication, computation, and processing platform for V2I applications, and providing the functions listed below.

* Message handling across multiple interfaces using SAE J2735 messages:
  + Integrating data from multiple sources and compiling messages for delivery to vehicles and nomadic devices via multiple communication methods.
  + Obtaining and aggregating data from multiple vehicles and nomadic devices, and sending to the Transportation Management Entity.
  + Distribution of Traveler Information Messages (TIM) to local vehicles and devices.
* Examples of local infrastructure-based computation and processing:
  + Local computation of recommended speeds and stopping distances using real time weather and road condition data for crash imminent V2I safety scenarios such as Reduced Speed (Work Zone) Warning and Spot Weather Information Warning.
  + Aggregation of vehicle weather data for efficient communication to Transportation Management Entity for weather-responsive traffic management.
  + Multi-Modal Intelligent Traffic Signal Systems (MMITSS) “intersection level” functions including J2735 Intersection Geometry (MAP) and J2735 Signal Phase and Timing (SPAT) broadcast manager, equipped vehicle tracker, priority request server, and interface to traffic signal controller.

V2I Hub was designed to support a variety of connected vehicle safety applications. The table below lists many of these applications (although plugins to support all of these applications have not currently been developed).

|  |  |
| --- | --- |
| **Dynamic Mobility Applications**   * + INFLO Speed Harmonization (SPD-HARM)   + INFLO Queue Warning (Q-WARN)   + RESCUME Incident Zone (INC-ZONE) - Low latency comm for V2V; High latency comm for V2I   + FRATIS – High latency communications   **Multi-Modal Intelligent Traffic Signal System**   * Intelligent Traffic Signal System * Transit Signal Priority * Pedestrian Mobility * Freight Signal Priority * Emergency Vehicle Priority   **AERIS Applications**   * Eco-Signal Operations * Eco-Traffic Signal Timing * Eco-Approach and Departure at Signalized Intersections * Eco-Traffic Signal Priority * Connected Eco-Driving * Dynamic Low Emissions Zones * Dynamic Eco-Lanes | **Transit Applications**   * Pedestrian Crossing Warning (PCW)   **V2I Safety Applications**   * Red-Light Violation Warning (RLVW) * Stop Sign Gap Assist (SSGA) * Curve Speed Warning (CSW) * Stop Sign Violation Warning (SSVW) * Railroad Crossing Violation Warning (RCVW) * Spot Weather Information Warning (SWIW) * Oversize Vehicle Warning (OVW) * Reduced Speed Zone Warning (RSZW) – Speed Reduction and Lane Closure Advisories * Reduced Speed Zone Warning (RSZW) – Lane Closure Alerts & Warnings   **Road Weather Connected Vehicle Applications**   * Enhanced Maintenance Decision Support System (MDSS). * Information for Maintenance and Fleet Management Systems. * Weather-Responsive Traffic Management. * Motorist Advisories and Warnings. * Information for Freight Carriers. * Information and Routing Support for Emergency Responders. |
| Source: Battelle | |

Tab 3: Release Notes

* Version 2.3
* Uses J2735 R41 (2015) message set
* Contains API, Core, MAP Plugin, SPaT Plugin, CSW Plugin, DMS Plugin, RTCM Plugin, and Location Plugin

Hardware Requirements for Installation and Running V2I Hub:

* Intel Core i3 processor
* 4GB of memory
* 10 GB of HD space
* Ubuntu 14.04 LTS with packages listed below

Security and Passwords:

* V2I Hub is middleware that runs on Linux Ubuntu 14.04 LTS. It is recommended that appropriate security and firewall settings be used on the computer running Linux, including conforming to your agency’s security best practices and IT protocols.

Operational Requirements:

* Traffic signal controller producing an NTCIP 1202 message over Ethernet (Econolite ASC/3 with firmware 2.58 or greater), which is used by the SPAT Plugin.
* 3.0 specification RSU with GPS attachment, which is used to transmit data over Dedicated Short Range Communications (DSRC).
* XML input files for the MAP and SPaT plugins specific to the deployment intersection.

The V2I Hub software application was developed using c and c++ and requires Ubuntu 14.04 LTS with the following packages installed via apt-get:

* cmake
* gcc-4.8
* g++-4.8
* libboost1.55-dev
* libboost-thread1.55-dev
* libboost-regex1.55-dev
* libboost-log1.55-dev
* libboost-program-options1.55-dev
* libboost1.55-all-dev
* libxerces-c-dev
* libcurl4-openssl-dev
* libsnmp-dev
* libmysqlclient-dev
* libjsoncpp-dev
* uuid-dev
* libusb-dev libusb-1.0.0-dev
* libftdi-dev
* swig
* liboctave-dev
* gpsd libgps-dev
* portaudio19-dev libsndfile-dev
* libglib2.0-dev libglibmm-2.4-dev
* libpcre3-dev
* libsigc++-2.0-dev
* libxml++2.6-dev
* libxml2-dev
* liblzma-dev

The following must be compiled and installed from the included source in TMX\_OAM/Externals directory.

1. Mysql-connector-c\_\_-1.1.3: Run the following from your TMX\_OAM/Externals directory:
   1. tar xzf mysql-connector-c++-1.1.3.tar.gz
      1. cd mysql-connector-c++-1.1.3
      2. cmake .
      3. make
      4. sudo make install
      5. cd ..
      6. rm -rf mysql-connector-c++-1.1.3
2. J2735r41: Run the following from your TMX-OAM/Externals directory:
   1. ./setupAsnJ2735\_r41.sh

Compilation Instructions

Each project contains its own cmake file. Downloaders will need to compile the projects in TMX/Core first. Start with the TmxUtils project, then the TmxApi, followed by the TmxCore. After those are built, you can compile any of the projects from the Plugins directory.

To compile a project:

* Change directory to the Debug or Release subfolder for the project (cd Debug)
* Run cmake (cmake ..)
* Run make (make)
  + This will compile the project
* To create a deployment package type make package

Set Up and Configuration Instructions

Instructions can be found in the *V2I Hub Sample Set-up Guide*.

Tab 4: Documentation

* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub System Requirement****s*
  + This document defines the requirements for the V2I Hub platform and message handlers recommended for development.
* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub Interface Control Document (ICD)***
  + This document captures the external interfaces necessary to support the V2I Hub platform and related messages between the platform and a variety of infrastructure devices, including traffic signal controllers, back-office systems, mobile devices, etc.; organized around the V2I applications supported by the platform.
* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub Design***
  + This report documents the System Design Document (SDD) for an Integrated V2I Prototype platform that is a key research activity within the Dynamic Mobility Applications program.
* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub Plugin Creation Manual***
  + Instructions for creating plugins for V2I Hub.
* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub Troubleshooting Guide***
  + Information on potential issues and resolutions.
* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub Sample Set-up Guide***
  + Information on the required hardware and software, hardware connections, and configuration for a sample deployment of V2I Hub to generate intersection geometry, Signal Phase and Timing (SPaT), and position correction messages.
* *Integrated Vehicle-to-Infrastructure Prototype (IVP):* ***V2I Hub Deployment Checklist and Guidance***
  + Information outlining the deployment process, activities, and supporting guidance.

Tab 5: Discussion

Main Discussion

Issue Tracker

Tab 6: Similar Applications

To be populated from metadata file