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 J2735-2016 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.

* CSW Plugin – 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.
* DMS Plugin – 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.
* DSRC Immediate Forward Plugin – Sends all J2735 traffic to the 4.1 RSU for transmission out the DSRC radio.
* Message Receiver Plugin – Receives all J2735 traffic incoming from the 4.1 RSU for consumption by other V2I Hub plugins.
* Location Plugin – Communicates with GPS devices producing location information and optionally the NMEA GP\* sentences for the V2I Hub system.
* RTCM Plugin – 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 3.1
* Uses J2735 R63 (2016) message set
* Contains API, Core, DSRC Message Manager Plugin, 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 16.04 LTS with packages listed below

Security and Passwords:

* V2I Hub is middleware that runs on Linux Ubuntu 16.04 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.
* 4.1 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.

Prerequisites

The V2I Hub software application was developed using c and c++ and requires Ubuntu 16.04 LTS with the following packages installed via apt-get:

* cmake
* gcc-5
* g++-5
* libboost1.58-dev
* libboost-thread1.58-dev
* libboost-regex1.58-dev
* libboost-log1.58-dev
* libboost-program
* options1.58-dev
* libboost1.58-all-dev
* libxerces-c-dev
* libcurl4-openssl-dev
* libsnmp-dev
* libmysqlclient-dev
* libjsoncpp-dev
* uuid-dev
* git
* libusb-dev
* ibusb-1.0.0-dev
* libftdi-dev
* swig
* liboctave-dev
* gpsd libgps-dev
* portaudio19-dev
* libsndfile-dev
* libev-dev
* libuv-dev
* libglib2.0-dev
* libglibmm-2.4-dev
* libpcre3-dev
* libsigc++-2.0-dev
* libxml++2.6-dev
* libxml2-dev
* liblzma-dev
* dpkg-dev

Run the following command to install prerequisites via apt-get:

$ sudo apt-get install cmake gcc-5 g++-5 libboost1.58-dev libboost-thread1.58-dev libboost-regex1.58-dev libboost-log1.58-dev libboost-program-options1.58-dev libboost1.58-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 libsndfile1-dev libglib2.0-dev libglibmm-2.4-dev libpcre3-dev libsigc++-2.0-dev libxml++2.6-dev libxml2-dev liblzma-dev dpkg-dev libmysqlcppconn-dev libev-dev libuv-dev git

Compilation Instructions

To Compile the V2I Hub software

Run the following from the “src” directory

$ cd tmx

$ cmake .

$ make

This will create a lib and bin folder in the TMX directory. The lib folder will contain libtmxapi.so and libtmxutils.a. The bin directory will contain the main tmxcore application along with zip files for each plugin.

- Install the libraries and includes

$ sudo make install

The library path to the new libraries needs to added to the LD\_LIBRARY\_PATH variable for the libraries to be found. This can be done in a per session basis with the command line.

$ export LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:/usr/local/lib

When the unit is rebooted this variable will not be set. To add the path at bootup modify the correct file in the /etc/ld.so.conf.d/ directory and add a line with the path “/usr/local/lib/”. For an Intel 64 bit system the correct file is “x86\_64-linux-gnu.conf” but this will vary based on your platform. After the file is modified run this command to update the paths.

$ sudo ldconfig

The V2I Hub supplied plugins have a dependency on a version of libwebsockets that is newer than the installable package that comes with Ubuntu 16.04. Therefore, a custom version of the software needs to be downloaded and compiled locally before compiling the V2I Hub plugins. Note this requires the GIT tool for checking out the latest version of the source code.

$ cd <some tmp dir>

$ git clone <https://libwebsockets.org/repo/libwebsockets>

$ cd libwebsockets

$ cmake -DLWS\_WITH\_SHARED=OFF .

$ make

$ sudo make install

The new libwebsockets static library should now be available in /usr/local to build against.

Now, run the following from the V2I-HUB directory

$ cmake .

$ make

This will create a bin directory that contains the plugin executable, as well as a directory for each plugin. However, a V2I Hub plugin must be packaged in a ZIP file to be installed to a system. In order to package up any one of the plugins from the V2I-HUB directory, do the following:

$ ln -s ../bin <PluginName>/bin

$ zip <PluginName>.zip <PluginName>/bin/<PluginName> <PluginName>/manifest.json

The binary and the manifest file are the minimum number of files needed for any V2I Hub plugin. It is possible some specific plugins require more files from the sub-directory to be included in the installable ZIP.

Installation Instructions

* install lamp-server
  + $ sudo apt-get install lamp-server^
  + enter a root password (i.e. ivp)
* install database
  + modify the install\_db.sh script. Modify the value for DBROOTPASS to the password that was used for root during the previous step
  + save the script
  + execute the script using the following commands
  + $ chmod +x install\_db.sh
  + $ sudo ./install\_db.sh
* To setup a service to start tmxcore on Ubuntu copy the tmxcore.service file to the “/lib/systemd/system/” directory. Execute the following commands to enable the application at startup.
  + $ sudo systemctl daemon-reload
  + $ sudo systemctl enable tmxcore.service
  + $ sudo systemctl start tmxcore.service

Set Up and Configuration Instructions

The CommandPlugin plugin must be running to access the Administration Portal. Follow the instructions above to build the CommandPlugin.zip package and then refer to Chapter 3 of the V2I Hub Administration Portal User Guide for installation and configuration instructions.

Instructions can be found to install additional plugins in the V2I Hub Software Configuration Guide.

Administration Portal

The Administrator Portal can be launched by opening the v2i-webportal/index.html file with either Chrome or Firefox. Further instructions for hosting the portal on a web server can be found in the V2I\_Hub\_AdministrationPortalUserGuide.pdf.

*NOTE: The MAP plugin will need an input file in order to run. A sample input file for Turner Fairbank has been included in this deployment in the Sample MAP Input folder.*

Copy sample MAP input file

* $ sudo cp Sample MAP Input\ STOL\_MAP.xml /var/www/plugins/MAPr41/
* $ cd /var/www/plugins/MAP/
* $ sudo chmod 644 STOL\_MAP.xml
* $ sudo chown www-data STOL\_MAP.xml
* $ sudo chgrp www-data STOL\_MAP.xml
* cd src

Tab 4: Documentation

* ***V2I Hub Guidebook***
  + Provides an overview of infrastructure connectivity equipment supporting CAV technologies and explains the hardware and software required for the V2I Hub deployment.
* ***V2I Hub Deployment Guide***
  + Provides a detailed approach for deploying the V2I Hub vehicle to infrastructure solution at a signalized intersection. It walks through each phase of the deployment step-by-step to ensure a successful implementation.
* ***V2I Hub Plugins***
  + Provides descriptions and configuration details for the V2I Hub plugins created during the Integrated V2I Prototype (IVP) and V2I Reference Implementation project.
* ***V2I Hub Plugin Programming Guide***
  + Provides details on how to create a plugin for customized V2I Hub software development.
* ***V2I Hub Software Configuration Guide***
  + Provides instructions and examples on how to configure the plugins for infrastructure V2I Hub deployment with MAP and SPaT.
* ***V2I Hub Administration Portal User Guide***
  + Provides an overview of operation and configuration of the Administration Portal used for the deployment of the V2I Hub.
* ***V2I Hub Vehicle Simulation Tool User Guide***
  + Describes the hardware and software configurations required, and provides step-by-step instructions, for creating scenarios or visually playing back previously recorded trip data using the V2I Hub Vehicle Simulation Tool.

Tab 5: Discussion

Main Discussion

Issue Tracker

Tab 6: Similar Applications

To be populated from metadata file