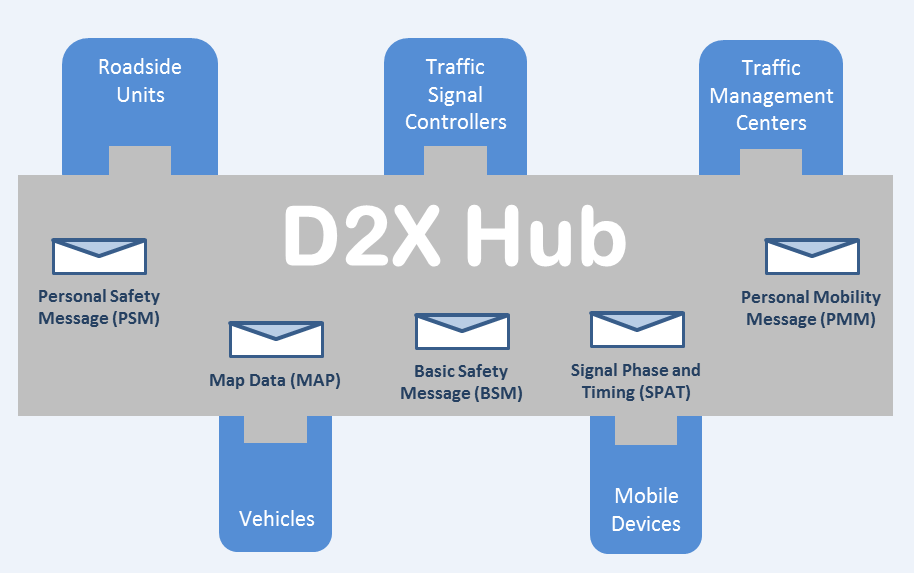
D2X Hub

# Content for Application Page

## Tab 1: Overview

Software applications are needed to allow mobile devices to share data in the formats unique to connected vehicles, roadside infrastructure and other travelers using mobile devices. The Mobile Device-to-Everything (D2X) software platform, called the D2X Hub, translates and coordinates the exchange of data between connected vehicles, ITS infrastructure devices, and travelers using mobile devices. The D2X Hub translates, aggregates and disseminates messages using the software applications within this platform that need to be installed on each connected mobile device, vehicle, ITS and traffic control device, or traffic management system.

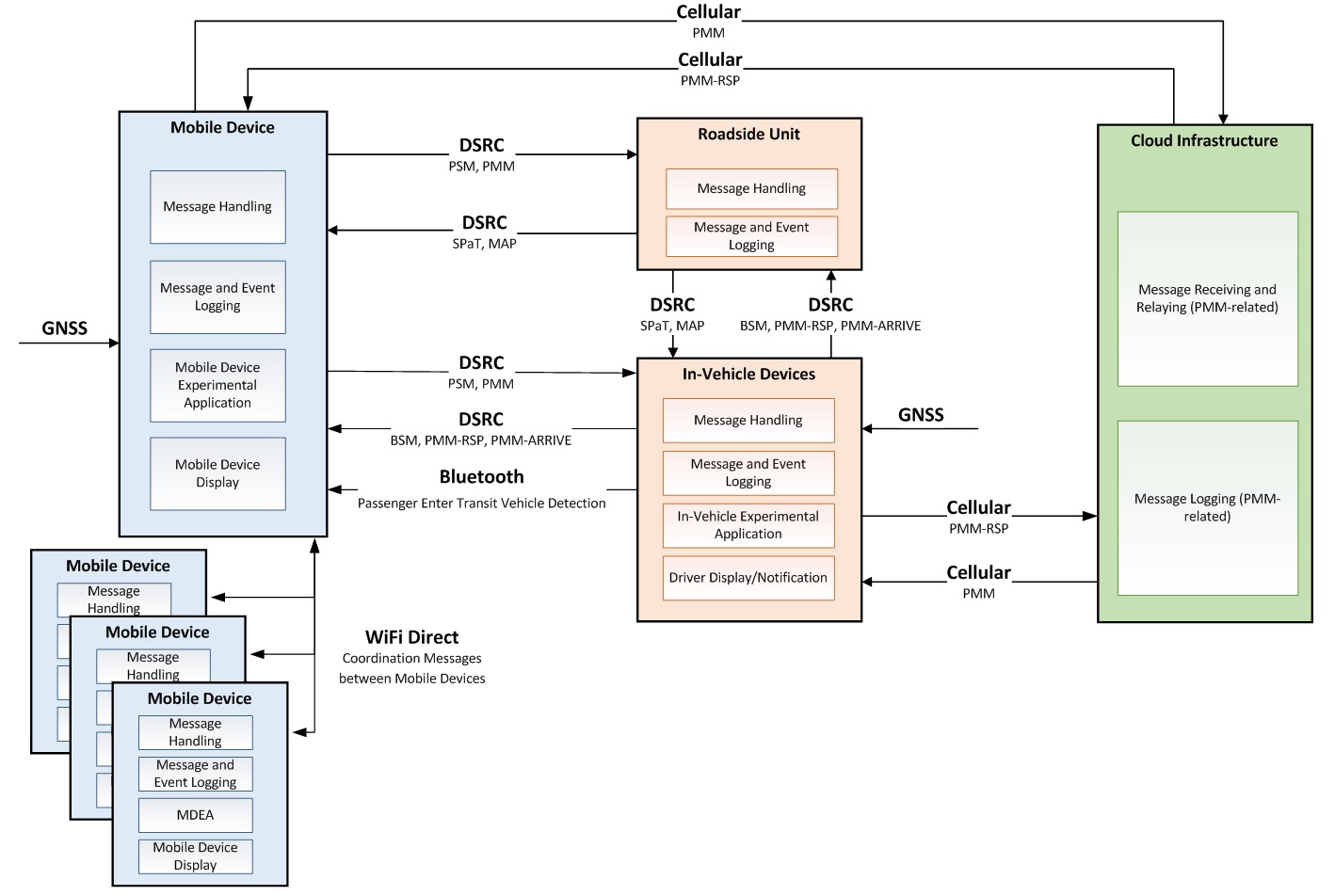
The D2X Hub supports the exchange and translation of several new types of messages to enhance the safety and mobility of travelers using connected mobile devices. These messages include a Personal Safety Message (PSM), a Personal Mobility Message (PMM) and a variety of other messages (e.g., PMM Response (PMM-RSP), PMM Arrival (PMM-ARRIVE)) used to coordinate message exchange between connected applications (mobile devices, connected vehicles and roadside units). The D2X Hub intakes these new messages and translates their contents to facilitate communication with connected vehicles (following the Society of Automotive Engineers (SAE) vehicular message standards) and ITS control devices (following the National Transportation Communications for ITS Protocol – NTCIP).



## Tab 2: Description

The D2X Hub provides a platform to facilitate agencies’ initiatives of data exchange between mobile devices, traffic management systems, ITS devices and vehicles. The D2X Hub is a message handler, supporting the translation of data between different industry standards used by connected vehicles, devices, or components. The D2X Hub uses a modular component design to facilitate the integration of its applications with the hardware or applications specific to different traffic management systems, ITS devices, connected vehicles, or other mobile devices to support this data exchange.

The four main components of the D2X Hub are: (1) the Mobile Device, (2) the In-Vehicle Device, (3) the Roadside Unit and (4) the Cloud Infrastructure, which are identified in the D2X System Architecture Diagram. The architecture diagram also depicts the messages and communication media that are supported to enable this exchange between these components.



**D2X System Architecture Diagram**

As part of the proof of concept testing of the D2X Hub functionality, the connected Mobile Device or phone used by a pedestrian or traveler, was equipped with an application, called the Mobile Device Experimental Application (MDEA), which provides travelers with an interface to request a ride, receive updates on when the ride will arrive and receive safety notifications when the MDEA identifies a safety condition (e.g., connected traveler in path of vehicle) based on the BSMs sent by the VEAs on connected vehicles. The In-Vehicle Experimental Application (VEA) was installed on a smart device allowing it to exchange data with other connected vehicles (which, for the purpose of these tests were assumed to be a taxi or transit vehicle), connected mobile devices, and ITS Devices. The integration of the VEA as a plugin to the D2X Hub allows vehicles to respond to travel requests, indicate passenger pick up locations, receive warnings of nearby travelers and send safety notifications to connected travelers. A Roadside Unit was installed and integrated with the traffic signal controller allowing it to receive, log and send messages (e.g., SPaT, MAP) via DSRC. The D2X Hub’s Cloud Infrastructure supports the exchange of mobility messages over the cellular network when connected elements are too far apart to utilize DSRC media.

The D2X Hub utilizes several messages that are a part of the SAE J2735 r63 specification: BSM, MAP, SPaT and the recently added PSM. In addition, a new DSRC message was created to support the functionality of travel coordination; the Personal Mobility Message (PMM) has variable data fields to support new or updated requests (PMM), responses to requests (PMM-RSP), and ride arrival messages (PMM-Arrive).

The D2X Hub testing and acceptance included the functionality of the software applications and the ability to exchange and translate data in messages with different formats. Traveler safety testing dealt with strategies to detect if a connected traveler (e.g., traveler using a connected mobile device that could include pedestrians, bicyclists, traveler in a vehicle) was a pedestrian on the curb, in the roadway, or was a passenger inside a vehicle. Multiple tests experimented with a variety of technologies and strategies to determine the connected traveler’s location and status.

Traveler coordination testing addressed questions related to the ability of travelers using connected mobile devices to coordinate their trips with groups of connected travelers to minimize and consolidate the number of messages these devices need to send and receive within those groups. For example, if a number of connected travelers were waiting at a transit stop and each traveler was using ride-coordination technologies, they would each generate and transmit the same number and type of messages. In situations where several connected travelers coordinated with the same targeted vehicle (taxi or transit bus) and itinerary (pick-up location, pick-up time and destination location), the connected travelers formed a group allowing for the coordination and consolidated of the number of messages these devices needed to send and receive, thereby reducing the load on the communication media and each individual device.

## Tab 3: System Components and Messages:

This section provides details on the components that comprise the D2X Hub and the new messages that were created.

Mobile Device Experimental Application (MDEA)

The Mobile Device Experimental Application (MDEA) is the application that resides on the smartphone of the connected traveler. It has three (3) primary sub-modules:

* Pedestrian Safety Monitor – to monitor incoming BSM messages and transmit PSM messages.
* Ride Request Monitor – to manage PMM messages and travel groups.
* MDEA User Interface – to manage user interaction with Safety and Ride monitors.

In-Vehicle Experimental Application (VEA)

The In-Vehicle Device Experimental Application (VEA) is the application that resides on the bus/taxi Onboard Unit (OBU). It has three (3) primary sub-modules:

• VEA Pedestrian Monitor – to transmit BSM messages and monitor PSM messages.

• VEA Ride Request Monitor – to monitor PMM messages and respond to them.

• VEA User Interface – to manage user interaction with Safety and Ride monitors.

Roadside Experimental Application (REA)

The Roadside Experimental Application (REA) is the application that resides on the Roadside Unit (RSU), located nearby a passenger pick-up spot. It serves to receive and log information, as well as transmit SPaT & MAP messages. SPaT was simulated for the purpose of the proof of concept testing. It has two (2) primary sub-modules:

• REA Monitor – to log BSM, PSM and PMM messages received via DSRC interfaces.

• SPaT & MAP message generator – to broadcast simulated SPaT and MAP messages.

Cloud API

The PMM Message Receiver Web API module provides a Web API interface that is utilized by both the mobile devices and the in-vehicle devices to exchange ride request information. The Web API consists of functions that allow requests to be added to the system, checking for pending requests, and responding to requests.

A Microsoft Azure SQL Database module provides database functionality so that the Web Service is able to create and modify database entries. The database is leveraged to route PMM messages between connected travelers and vehicles prior to being within DSRC range.

Mobile DSRC Message Handler

Mobile phones are not yet equipped with DSRC communication technology. In order to provide this functionality, the mobile device is paired (via Bluetooth) to a portable Arada radio. The Arada Portable DSRC radio is responsible for providing all the DSRC communication needs of the mobile device. The transmission and receipt of all messages sent over DSRC is processed, decoded, and formatted on this platform. Software residing on the Arada is customized to provide functionality to the MDEA to control what messages are sent out. The Arada also receives and passes on incoming DSRC messages from other devices.

PSM – Personal Safety Message

The PSM message is transmitted at a configurable frequency by the mobile device when a connected traveler is not within a vehicle. By sending a PSM, the mobile device broadcasts its location, direction of travel and group information (if applicable). Connected travelers within an ad-hoc travel group stop sending individual PSMs since their safety is represented by the ad-hoc travel group leader’s mobile device. PSM data format is as specified in SAE J2375:2016.

* Fields
  + Date [datetime]: date and time that this message information corresponds to.
  + Latitude [floating point number]: Latitude where traveler is.
  + Longitude [floating point number]: Longitude where traveler is.
  + Elevation [floating point number]: Elevation where traveler is.
  + Position Accuracy [floating point number]: Position Accuracy of location where traveler is.
  + Speed [floating point number]: Speed of traveler.
  + Heading [floating point number]: Heading of traveler.
  + PSM Number of Peds [int]: Number of pedestrians represented by the PSM.
  + PSM Radius of Protection [floating point number]: Radius from Lat/Long that represents the pedestrian’s space. Set in 0.1 meter increments.
  + PSM Path History: The PSM shall specify the pedestrian path history for the last 5 seconds. Lat/Long of location, timestamp.
  + Path Prediction: The PSM shall specify the pedestrian path prediction predicted as a straight line of travel.

PMM – Personal Mobility Message

PmmType

{

New,

Updated,

Arrive,

Response

}

ModeOfTransportType

{

NoPreference = 0,

Transit=1,

Taxi=2,

RideShare=3

}

MobilityNeedsType

{

NoSpecialNeeds=0,

Wheelchair=1

}

PMM messages are transmitted by a connected traveler’s mobile device at a configurable frequency; transmitting when travel arrangement and seat reservation is desired. Once activated, PMMs are continually transmitted by a traveler’s mobile device until a response is obtained from a responding vehicle.  Connected travelers with the same travel arrangements are grouped, and only one PMM request is generated to represent the entire travel group.  Cancellation of a trip is performed by the traveler sending an Update message with zero seats requested.

* Fields
  + GroupId [GUID]: Unique identifier for this travelling group, remains unchanged from first request to final boarding.
  + RequestId [int]: Id within group of iterations of requests, increments with each update.
  + Status [string enum PmmType]: New or Updated.
  + RequestDate [year/month/day/hour/minute]: date and time that request was created.
  + PickupDate [year/month/day/hour/minute]: date and time that the travelers wish to be picked up.
  + Position: Latitude/Longitude where travelers will be awaiting pickup.
    - Latitude [floating point number]
    - Longitude [floating point number]
    - Elevation [floating point number]
  + Destination: Latitude/Longitude where travelers want to travel to.
    - Latitude [floating point number]
    - Longitude [floating point number]
    - Elevation [floating point number]
  + MobilityNeeds: number of travelers included per mobility type
    - Type[string enum MobilityNeedsType]: no special needs, wheelchair, etc
    - Count[int]: number of travelers of this type
  + ModeOfTransport [string enum ModeOfTransportType]:  Preferred method of desired transportation

PMM RSP – PMM Acknowledgement/Response

Transmitted by VEA whenever a PMM request is received to indicate the number of travelers accepted for pickup from this vehicle.

* Fields
  + GroupId [GUID]: Return of GroupId of PMM this PMM-RSP is in response to.
  + RequestId [int] : Return of RequestId of PMM this PMM-RSP is in response to.
  + Status [string enum PmmType]: Response.
  + RequestDate [year/month/day/hour/minute]: date and time that response was created.
  + Position: Latitude/Longitude where vehicle is.
    - Latitude [floating point number]
    - Longitude [floating point number]
    - Elevation [floating point number]
  + IsDSRCEquipped [Boolean]: True if the vehicle is capable of broadcasting and receiving DSRC messages.
  + Eta [integer]: Estimated time (in seconds) until arrival; e.g. zero if arrived.
  + MobilityNeeds: number of travelers accepted per mobility type
    - Type[string enum MobilityNeedsType]: no special needs, wheelchair, etc
    - Count[int]: number of travelers of this type that have been accepted.

PMM ARRIVE – Vehicle Arrival

Transmitted by VEA when the vehicle arrives at the pickup location. The mobile device will then display to the travelers that their ride has arrived.

* Fields
  + GroupId [GUID]: Return of GroupId of PMM this message is in response to.
  + RequestId [int] : RequestId of PMM this PMM-ARRIVE is in response to.
  + Status [string enum PmmType]: Arrive.
  + RequestDate [year/month/day/hour/minute]: date and time that this message was created.
  + Position: Latitude/Longitude of vehicle location.
    - Latitude [floating point number]
    - Longitude [floating point number]
    - Elevation [floating point number]
  + Eta [integer]: Seconds until arrival; e.g. zero if arrived.
  + VehicleDesc [text]: a visible description indicator of the vehicle to aid in human identification of the vehicle.

## Tab 4: Release Notes:

This section is intended to identify the hardware, D2X Hub software, operating, and data sharing requirements. In addition, information is provided to support the download, installation and operation of the software components to support the desired exchange of messages and data.

D2X Hub Version 1.0 2/2017.

Contains software components: Mobile Device Experimental Application (MDEA), In-Vehicle Experimental Application (VEA), Roadside Experimental Application (REA), Cloud API, and Mobile DSRC Message Handler.

Messages: Uses J2735 R63 message set with added custom PMM and PSM message types.

## License

This software is intended for use in Connected Vehicle research projects supported by the U.S. Department of Transportation and shall not be used for, or with, commercial products. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>. Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## System Requirements

Following are the computer system requirements for development, installation, and operation of the D2X Hub applications. Please refer to the System Architecture and Design Document (TBD link) for additional information on the D2X Hub system and its design.

Mobile Device Experimental Application (MDEA) – Built on Xamarin and Visual Studio

* PC running Windows 7 or higher
* Visual Studio 2015
* Xamarin 4.3.0.784
* Android-compatible mobile device
* Android version minimum target: API Level 21 – Lollipop
* Xamarin.Android 7.1.0.41

In-Vehicle Experimental Application (VEA) – Built on V2I Hub. V2I Hub is middleware that runs on Linux Ubuntu 14.04 LTS. Please refer to the V2I Hub Documentation on the OSADP: <https://www.itsforge.net/index.php/community/explore-applications/for-search-results#/40/125>

* Battelle Common Computing Platform (CCP) (i.MX6-based single board computer with Cohda RSU could alternatively be employed, using different set of plugins)
* V2I Hub core software v2.3.1 or higher
* V2I Hub LocationPlugin v1.1.0 or higher
* CCPStatusPlugin v0.1.0 or higher
* CohdaInterfacePlugin v0.1.5 or higher

Roadside Experimental Application (REA) – Built on V2I Hub. V2I Hub is middleware that runs on Linux Ubuntu 14.04 LTS. Please refer to the V2I Hub Documentation on the OSADP: <https://www.itsforge.net/index.php/community/explore-applications/for-search-results#/40/125>

* Battelle Common Computing Platform (CCP) (i.MX6-based single board computer with Cohda RSU could alternatively be employed, using different set of plugins)
* V2I Hub core software v2.3.1 or higher
* V2I Hub MAP R41 Plugin v1.1.6 or higher
* V2I Hub SPAT R41 Plugin v1.0.2 or higher
* CCPStatusPlugin v0.1.0 or higher
* CohdaInterfacePlugin v0.1.5 or higher

Cloud API

* PC running Windows 7 or higher
* Visual Studio 2015

Mobile DSRC Message Handler

The Mobile DSRC Message Handler application is written in C and C++ using interfaces supplied by the manufacturer. The development for the application that resides on the DSRC radio uses Arada’s WAVE API version 1.86. The application executes on a Linux operating system using the LocoMate™ tool chain version 1.42. This API includes Security, GPS positioning and SAE J2735 messaging.

## Installation and Removal Instructions

This section provides installation and removal instructions for the D2X Hub applications. Please refer to the System Architecture and Design Document (TBD link) for additional information on the D2X Hub system and its design.

Mobile Device Experimental Application (MDEA)

Download and unzip zip file from OSADP. Open in Visual Studio. Build and deploy to mobile device using Visual Studio.

In-Vehicle Experimental Application (VEA)

Refer to the V2I Hub Documentation (<https://www.itsforge.net/index.php/community/explore-applications/for-search-results#/40/125>) for instructions on installing the V2I Hub core software before proceeding to add the VEA software. The VeaPedestrianMonPlugin and VeaRideRequestPlugin are the two plugins created for D2X Hub and are installed following the same procedure as those for V2I.

The following plugins should be running on the VEA CCP:

|  |  |
| --- | --- |
| Plugin Name | Version |
| CCPStatus | 0.0.4 |
| CloudInterfacePlugin | 0.0.29 |
| Cohda DSRC TxRx | 1.0.5 |
| CohdaInterface (Enable Bsm=true) | 0.1.1 |
| EventLoggingPlugin (log level Info) | 0.0.3 |
| LocalLogging | 0.5.0 |
| Location | 1.1.0 |
| TMX UI Information | 1.0.0 |
| TMX UI MobileDevicesUI | 1.0.0 |
| VeaPedestrianMonPlugin | 0.0.1 |
| VeaRideRequestPlugin | 0.0.1 |

Roadside Experimental Application (REA)

Refer to the V2I Hub Documentation (<https://www.itsforge.net/index.php/community/explore-applications/for-search-results#/40/125>) for instructions on installing the V2I Hub core software, which provides the D2X Hub REA.

The following plugins should be running on the REA CCP:

|  |  |
| --- | --- |
| Plugin Name | Version |
| CCPStatus | 0.0.4 |
| Cohda DSRC TxRx | 1.0.5 |
| CohdaInterface(Enable Bsm=false) | 0.1.1 |
| EventLoggingPlugin | 0.0.3 |
| LocalLogging | 0.5.0 |
| Location | 1.1.0 |
| Map Plugin | 1.1.6 |
| Spat Plugin | 1.0.2 |

Cloud API

Download and unzip zip file from OSADP. PublishProfiles have not been included. Please recreate these to point to your Azure resources where you wish to deploy.

Mobile DSRC Message Handler

Refer to the V2I Hub Documentation for instructions on installing the V2I Hub core software before proceeding to compile the Mobile DSRC Message Handler (Arada) software. The TMX\_OAM directory must already be in place.

Compilation Instructions

* Change directory to the MIPS-Release subfolder for the project (cd MIPS-Release).
* Run cmake (cmake ..) to build the makefile.
* Run make (make) to compile the project.

Installation Instructions

* Transfer the compiled executable to the /var/bin directory of the Arada LocoMate device using standard Linux tools (e.g. scp).
* If the project contains a file named “treerepair.xml”, this file must also be transferred to the /var/bin directory of the LocoMate device.
* The user interface to the LocoMate™ ME is through a telnet connection using hardwired Ethernet. Once a user has logged into the device, a command line interface is used to configure the device using standard Linux commands.
* A command line interface is provided by Arada (i.e. cli) that allows the user to manage the operation of the DSRC radio and its external interfaces, including automatic startup of the application. Refer to the LocoMate User’s Guide for details.

## Operating Instructions

Please refer to the System Architecture and Design Document (TBD link) for additional information on the D2X Hub system and its design. Due to the experimental nature of the system and its applications, Operating Instructions were not required or developed.

## Related web sites

The software is distributed through the USDOT's JPO Open Source Application Development Portal (OSADP), <http://itsforge.net/>

## Tab 5: Documentation

* Concept of Operations. Sharing Data Between Mobile Devices, Connected Vehicles and Infrastructure, Task 3: Concept of Operations Technical Memorandum – FINAL: <http://ntl.bts.gov/lib/60000/60800/60820/FHWA-JPO-16-422.pdf>
* System Requirements. Sharing Data Between Mobile Devices, Connected Vehicles, and Infrastructure, Task 3: System Requirements Specifications (SyRS) – FINAL: <http://ntl.bts.gov/lib/60000/60800/60821/FHWA-JPO-16-423.pdf>
* System Architecture and Design. Sharing Data between Mobile Devices, Connected Vehicles, and Infrastructure, Task 4: System Architecture and Design Document – FINAL: 508 PDF delivered
* Proof of Concept Experimental Plan. Sharing Data between Mobile Devices, Connected Vehicles and Infrastructure, Task 5: Prototype Proof of Concept Field Demonstration Experimental / Field Demonstration Site Plan – FINAL: 508 PDF delivered
* Acceptance Test Plan. Sharing Data between Mobile Devices, Connected Vehicles and Infrastructure, Task 6: Prototype Acceptance Test Plan – FINAL: 508 PDF delivered
* Acceptance Test Summary Report. Sharing Data between Mobile Devices, Connected Vehicles and Infrastructure, Task 6: Prototype Acceptance Test Summary Report – DRAFT: Word doc delivered
* Proof of Concept Test Report. Sharing Data between Mobile Devices, Connected Vehicles and Infrastructure, Task 8: Prototype Proof of Concept Test Report – DRAFT: Word doc soon to be delivered

## Tab 6: Discussion

## Tab 7: Related Applications