

CONNECTED INTERSECTIONS MESSAGE MONITORING SYSTEMS REQUIREMENTS & PROTOTYPE DEVELOPMENT (CIMMS)

Final Report

February 2024

Prepared by



FINAL

Contents

1.	Introduction	1
1.1	DOCUMENT PURPOSE.....	1
1.2	PROJECT SCOPE.....	1
1.3	REFERENCES	4
2.	System Description	5
2.1	CIMMS INTERNAL ARCHITECTURE	7
2.2	ROADSIDE ARCHITECTURE	8
2.3	USER INTERFACE	10
2.3.1	CIMMS Dashboard	10
2.3.2	Map View	11
2.3.3	Data Selector.....	12
2.3.4	Configuration Page	13
2.4	SYSTEM REPORTS.....	13
2.5	OPERATIONS AND MAINTENANCE.....	14
2.5.1	Software Updates	14
2.5.2	Routine Maintenance	14
2.5.3	User Support and Management.....	15
3.	Project Tasks	16
3.1	PHASE I.....	16
3.1.1	Task 1: Program Management.....	16
3.1.2	Task 2: Stakeholder Engagement.....	16
3.1.3	Task 3: Concept of Operations.....	17
3.1.4	Task 4: System Requirements	17
3.2	PHASE II.....	18
3.2.1	Task 5: Prototype Development.....	18
3.2.2	Task 6: Prototype Demonstration and Evaluation	19
3.2.3	Task 7: Final Report.....	21
4.	Final Outcomes	22
4.1	OUTCOMES.....	22
4.2	LESSONS LEARNED	22
4.3	RECOMMENDATIONS FOR IMPROVEMENTS.....	23
5.	Concluding Remarks	26
Appendix A.	Version History.....	29

Tables

Table 1. Version History.....	29
-------------------------------	----



Figures

Figure 1. Anthem, AZ RSU Locations.....	6
Figure 2: Orem, UT RSU Locations.....	6
Figure 3: CIMMS Internal Architecture	7
Figure 4. Maricopa County DOT CIMMS Roadside Architecture	9
Figure 5. Utah DOT CIMMS Roadside Architecture.....	9
Figure 6. CIMMS User Interface – Dashboard / Landing Page.....	10
Figure 7. CIMMS User Interface – Map View	11
Figure 8. CIMMS User Interface – Data Selector.....	12
Figure 9. CIMMS User Interface – Configuration Page.....	13
Figure 10. Real-Time Network Device Communications Monitoring (Nagios)	25
Figure 11. Networked Device Traffic History (Nagios).....	25

Acronyms

BSM	Basic Safety Message
CBR	Cease Broadcast Recommendation
CI	Connected Intersection(s)
CIP	Connected Intersection Project
CV	Connected Vehicle
ECLA	External Control Local Application
IEEE	Institute of Electrical and Electronics Engineers
IOO	Infrastructure Owner/Operator
ITS	Intelligent Transportation System
MAP	MapData
OBU	Onboard Unit
OEM	Original Equipment Manufacturer
PFS	Pooled Fund Study
PSID	Product Service Identifier
RLVW	Red Light Violation Warning
RSU	Roadside Unit
SAE	Society of Automotive Engineers
SCMS	Security and Credentials Management System
SPaT	Signal Phase and Timing
TSC	Traffic Signal Controller
V2X	Vehicle-to-Everything
V2V	Vehicle-to-Vehicle
V2I	Vehicle-to-Infrastructure

1. Introduction

1.1 DOCUMENT PURPOSE

The Connected Intersection Message Monitoring System (CIMMS) has been developed and deployed alongside two operational connected vehicle environments in Anthem, Arizona (operated by Maricopa County DOT), and in Orem, Utah (operated by Utah DOT).

The purpose of this Final Report is to provide a description of the CIMMS system, recap each task, document the overall outcomes and findings of the project, identify lessons learned, and to provide concluding remarks.

The structure of this document is as follows:

- **Section 1** provides a document overview, and identifies all documents referenced in developing this document.
- **Section 2** describes the internal system architecture, how the CIMMS interfaces with the existing system at both deployment sites, as well as user interfaces, inputs/outputs, system reports, and operations and maintenance.
- **Section 3** provides a brief overview of each project task, the activities undertaken in each, and deliverables that were developed.
- **Section 4** provides an overview of the final outcomes and the lessons learned throughout the development, testing, and integration of the CIMMS. This section concludes with a list of development recommendations to consider for the CIMMS looking forward.
- **Section 5** provides concluding remarks about the CIMMS. The section contemplates the evolution of the CIMMS, discusses the Connected Intersections Implementation Guide (CTI 4501), and the practicality and effort required to make the changes necessary for compliance.

1.2 PROJECT SCOPE

Since the inception of Connected Vehicle (CV) technology, researchers and deployers have sought new and innovative ways to use CV to improve transportation safety, mobility, and efficiency. Most of these efforts have focused on the ability of vehicles to react to the data they receive from other vehicles and from the infrastructure. Forward Collision Warning, Red-Light Violation Warning, Curve Speed Warning, etc., are all examples of critical CV safety applications that utilize CV data and as more data becomes available, these applications continue to mature. In parallel, numerous advances to ensure timely and authenticated data is being provided to the CV environment continue. Robust fiber networks and the investment in the Security Credential Management Systems (SCMS) serve as proof of those investments. The assumption, however, has generally been that once a site had deployed and validated the broadcast messages, the data would remain correct. Two issues arise from this thinking:



- **Accuracy** – As has been the focus of the current CV PFS Connected Intersection Project (CIP), and others before it, validation of message content goes beyond conformance to the SAE J2735 standard. To truly be considered conformant to the needs of OEMs, deployers need to ensure that broadcast messages truly match what is happening at the intersection. For instance, signal indications on the traffic signal must match those in the SPAT message.
- **Consistency and Changes** – The validation of message accuracy is not only needed at the time when CV equipment is deployed, but also throughout active operations of a CV system. Signal timing patterns change, road geometries change, and devices fail – it is important to confirm that these changes are properly accounted for in the CV system simultaneously after the changes are made. Presently, only limited capabilities exist to determine if a device is even operational, so it is not a simple matter to determine if the messages a CV system produces contain data that correctly reflect ground truth.

A precursor to the CIMMS project, the CV PFS Connected Intersections Program (CIP) project¹ focused on validating a site's ability to conform to the newly published ITE CI Design Guidance², guidance which the OEMs agree will uniformly support advanced safety applications, such as RLVW. CIP is a validation of the guidance itself, feedback which will be provided to the industry. It's important to note that in practice, use of the guidance would typically only occur at deployment however, and similarly validate a site's ability to conform to the guidance.

But what happens after a site validates in conformance to the guidance? Agencies don't have the bandwidth or budgets to subject every intersection to the same rigors as the CIP on a repetitive basis or over an extended period of time. Many agencies are also preparing to leverage increasingly larger amounts of Vehicle-to-Everything (V2X) data from intersections and from vehicles, especially as the number of V2X-equipped vehicles increases. The Concept of Operations proposes that this V2X data can be leveraged to continuously validate the correct operation of the infrastructure over a long-term time horizon. The broad goal of the Connected Intersection Message Monitoring System is to evaluate this potential.

Note that the scope for this project (initial implementation of the message monitor) is limited to receiving CV messages from an existing CV system, and using SPaT, MAP, and BSMs (driver behavior that provides a proxy for ground truth conditions) to assess the correctness of data within SPaT and MAP messages. The ability to use driver behavior to infer ground truth is predicated on the fact that a driver's response to traffic control devices and the roadway environment is generally predictable (though not perfect). Thus, it is reasonable to assume the data in SPaT and MAP messages should be consistent with general vehicular movement as evidenced in BSMs. The correctness of data in MAP and SPaT messages was initially cited as being a priority. Limiting the message monitor to SPaT, MAP, and BSMs, as opposed to interfacing with other components in the traffic signal cabinet, simplifies the data and interfaces required between the existing system and the message monitor and minimizes the pre-requisites for the existing system. Any system for which CV data will be assessed should be able to produce and forward CV data to an external system such as the message monitor. Data from other (non-CV) sources at the intersection such as traffic signal controllers, conflict monitors, and sensing equipment (e.g. radar/LIDAR) while useful for assessing SPaT and MAP accuracy, increases

¹ <https://engineering.virginia.edu/labs-groups/cvpfs>

² <https://www.ite.org/ITEORG/assets/File/Standards/CTI%204501v0101-tracked.pdf>

the complexity of the integration and/or may not be available at every intersection. Thus, data from non-CV sources was not included for the initial implementation of the message monitor.

Other needs that have been identified during this process, such as the ability to assess message performance, generic message requirements, the correctness of position correction information in the RTCM message, and the impact of position correction data on the performance of message monitor algorithms that use BSM data are documented, but not considered in the initial implementation of the message monitor. However, the message monitor will be designed in a way that allows the modification/addition of interfaces and algorithms so that data from other sources can be utilized and so other needs can be addressed in future system development efforts.



1.3 REFERENCES

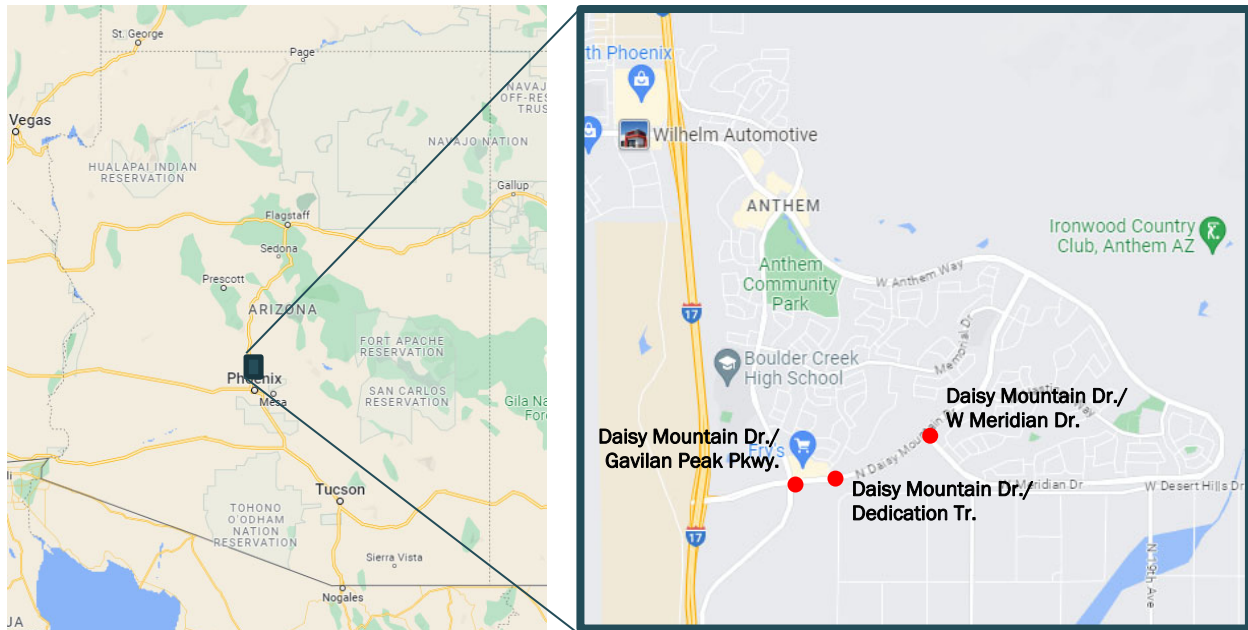
This section contains documents and literature utilized to gather input for this document.

- Connected Intersection Message Monitoring System-Concept of Operations 8/2023 Update
- Connected Intersection Message Monitoring System-System Requirements 12/2022 Final
- Connected Intersection Message Monitoring System-Software Design 4/2023 Final
- Connected Intersection Message Monitoring System-System Integration Design 10/2023 Update
- Connected Intersection Message Monitoring System-System Test Plan 8/2023 Final
- Connected Intersection Message Monitoring System-Assessment Report 12/2023 Draft
- CV PFS Map Guidance Document 2023
<https://engineering.virginia.edu/labs-groups/cvpfs>
- SAE J2735 2016-03. V2X Communications Message Set Dictionary. 2016
https://www.sae.org/standards/content/j2735_201603/
- SAE J2735 2020-07. V2X Communications Message Set Dictionary. 2020
https://www.sae.org/standards/content/j2735_202007
- SAE J2735 2022-11. V2X Communications Message Set Dictionary. 2022
https://www.sae.org/standards/content/j2735_202211
- CTI 4501 v01 – Connected Intersections (CI) Implementation Guide 2021
<http://www.ite.org/pub/76270782-B7E4-7F75-BC72-D5E318B14C9A>
- CTI 4001 v01 - Roadside Unit (RSU) Standard 2021
<http://www.ite.org/pub/764FB228-0F6C-BA02-6D7B-16A86B1F8108>
- SAE J2945/B Recommended Practices for Signalized Intersection Applications (Work in progress)
<https://www.sae.org/standards/content/j2945/b/>
- Operational Data Environment open-source decoder for MAP/SPaT/BSM 2021
<https://github.com/usdot-jpo-ode/jpo-ode> (latest version)
- CTI 4502 v01.00 – Connected Intersections Validation Report 2022
<https://www.ite.org/pub/?id=59A8D354-F7B1-6A18-6FCC-1CECE6ACDE5B>
- Connected Intersections Program, Connected Vehicle Pooled Fund Study
<https://engineering.virginia.edu/labs-groups/cvpfs>

2. System Description

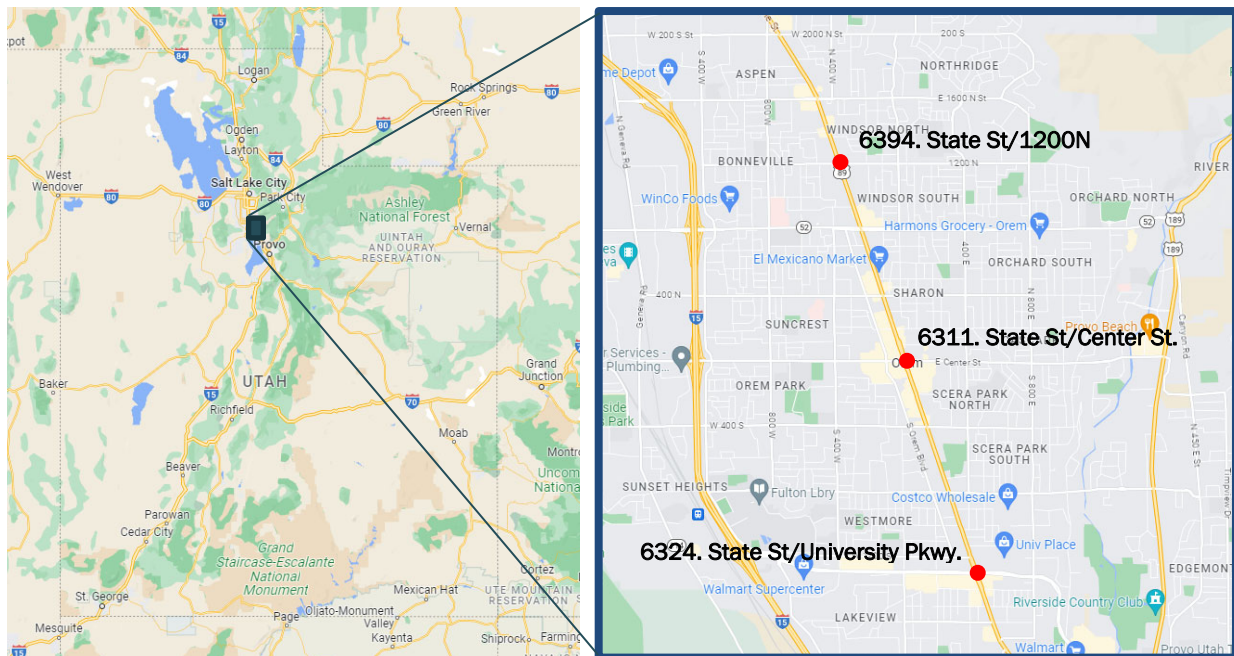
Two instantiations of the CIMMS proof of concept system were deployed. One instance interfaced with 3 network-connected Codha MK5 RSUs in Anthem, AZ (Figure 1), and the second instance interfaced with 3 Kapsch dual-mode RSUs deployed in Orem, UT (Figure 2). For the implementation in Anthem, the CIMMS was installed on virtual machines provisioned by MCDOT on a network-connected server. For the implementation in Orem, UDOT utilized a remote server hosted by Panasonic to host the virtual machines that CIMMS was installed on.

This section provides a description of the CIMMS system including the CIMMS internal architecture, the location of the CIMMS with respect to each system's existing architecture, as well as the CIMMS user interface and system outputs.



Note: Icons shown represent RSUs that interface with the CIMMS. It does not show all operational RSUs in the area.

Figure 1. Anthem, AZ RSU Locations



Note: Icons shown represent RSUs that interface with the CIMMS. It does not show all operational RSUs in the area.

Figure 2: Orem, UT RSU Locations

2.1 CIMMS INTERNAL ARCHITECTURE

The CIMMS internal architecture is comprised of several sub-components that perform various processes to meet the system requirements and software design, specified in documentation developed as part of this project. The internal architecture is provided in Figure 3, followed by a description of each subcomponent.

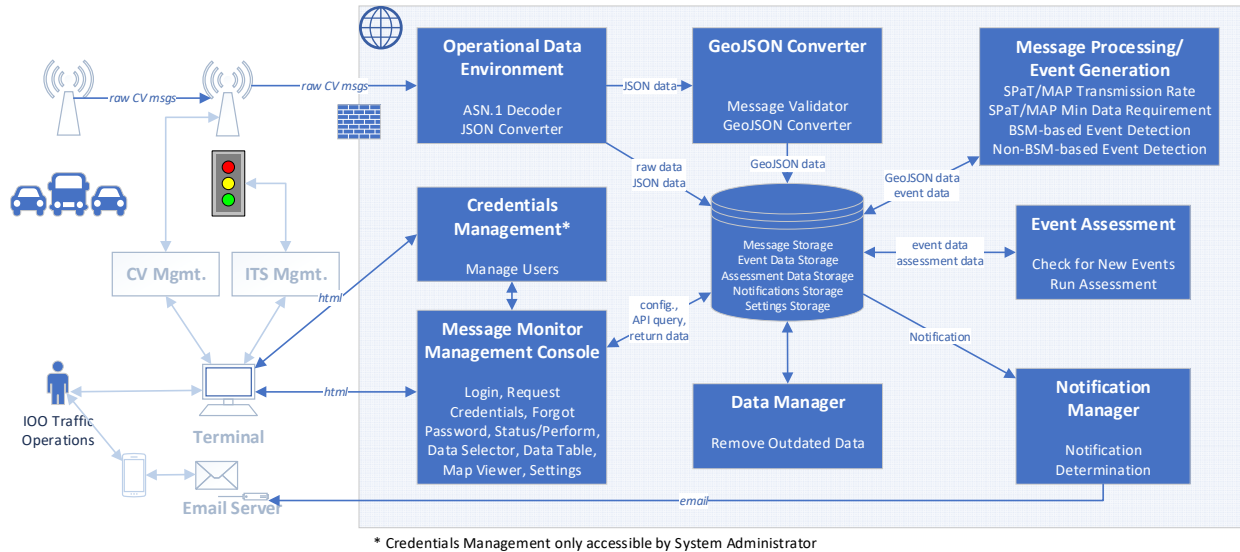


Figure 3: CIMMS Internal Architecture

Operational Data Environment (ODE) - The Operational Data Environment (ODE) is an open-source, scalable, message ingest and decoding solution for connected vehicle data. For CIMMS, the ODE is used to receive Unaligned Packed Encoding Rules (UPER)-encoded ASN.1 SPaT, MAP, and BSM messages forwarded from intersections. The ODE then decodes that data into JSON and forwards it to other CIMMS components.

GeoJson Converter - The GeoJson Converter ingests JSON encoded SPaT and MAP data from the ODE and converts it to a GeoJson encoded MAP and SPaT format (Processed MAP, Processed SPaT). The GeoJson Converter also performs message validation and normalization to streamline further processing. Note: The BSM is not converted to geoJSON.

Message Processing / Event Generation - This component utilizes algorithms to process MAP and SPaT data from the GeoJson Converter as well as BSMs from the ODE to perform continuous validation on connected intersections. User-specified parameters are used as inputs to most algorithms. It generates "events" which are stored on the database.



Event Assessment – The Event Assessment module performs assessments on the event data that are output by Message Processing/Event Generation. The assessment typically involves the aggregation of event data and the comparison of this calculated value against user-specified parameters to determine if a notification should be issued (see Notification Manager). The event assessment module generates “assessments” which are stored on the database.

Data Storage (Mongo DB) - The CIMMS application records all data in a central mongoDB database. This database includes data generated by the ODE, GeoJSON Converter, Message Processing/Event Generation, and the Message Monitor Management Console.

Credentials Management (Keycloak) - CIMMS runs a private Keycloak server to use for credential management and user validation. This server is used to provide authentication services to the Message Monitor Management Console and to track user roles and permissions within CIMMS.

Notification Manager – The Notification manager reads the data in an assessment, determines if a notification is needed, and if so, handles generating email alerts from the CIMMS system. Generated emails are then passed to an SMTP server for distribution to users. The specified email server may be a dedicated email server, or a 3rd party service that handles message distribution such as Send-Grid or Postmark.

Data Manager. The Data Manager monitors the data stored on the database, and automatically removes data that is older than the user-specified parameter.

Message Monitor Management Console – The message monitor is the primary portal for users interacting with the CIMMS system. It provides capabilities to graphically view MAP, SPaT, and BSM messages, query event and assessment data stored in the database, generate and download reports, and configure parameters used by the Message Processing/Event Generation and Event Assessment algorithms.

2.2 ROADSIDE ARCHITECTURE

The roadside architecture essentially captures how the CIMMS interfaces with RSUs and/or other roadside processing equipment at each deployment site. The Anthem CIMMS interfaces with 3 Codha MK5 RSUs and the Orem CIMMS interfaces with 3 Kapsch dual-mode RSUs and co-located external control local application (ECLA) devices. Due to architectural and functional variations between the roadside components between both sites, the CIMMS interfaces differently with each existing system.

Maricopa County DOT. In Maricopa County, SPaT and MAP are sent to the same IP Address and port. The ODE must identify and separate SPaT and MAP messages from the single feed before processing. BSMs are sent to the same IP Address as SPaT and MAP, but to a dedicated port. Figure 4 depicts the Maricopa County roadside architecture.

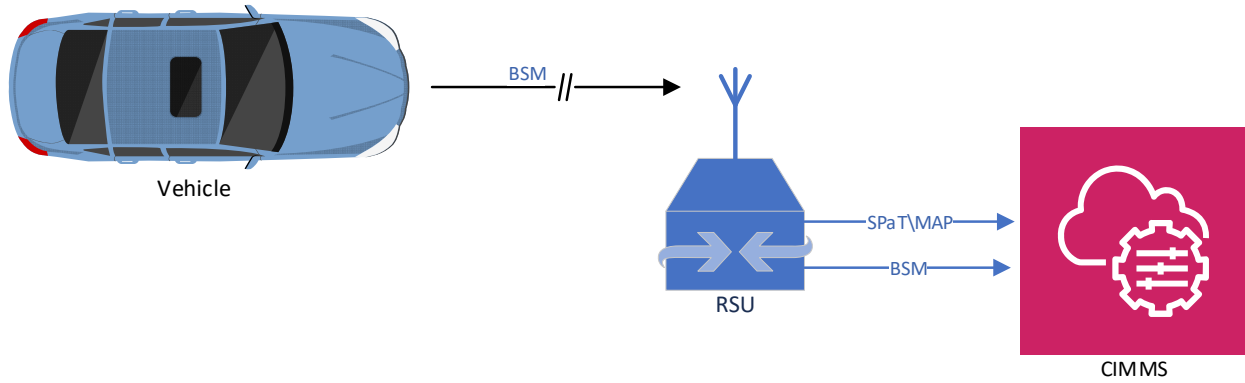


Figure 4. Maricopa County DOT CIMMS Roadside Architecture

Utah DOT. In Utah, SPaT, MAP, and BSMs are sent to dedicated ports. The RSU immediately forwards BSMs to the CIMMS, while the ECLA (which stores and manages the broadcast of MAP payloads, and also generates the SPaT payload using traffic signal controller data) forwards MAP and SPaT messages to the CIMMS. Figure 5 depicts the UDOT roadside architecture.

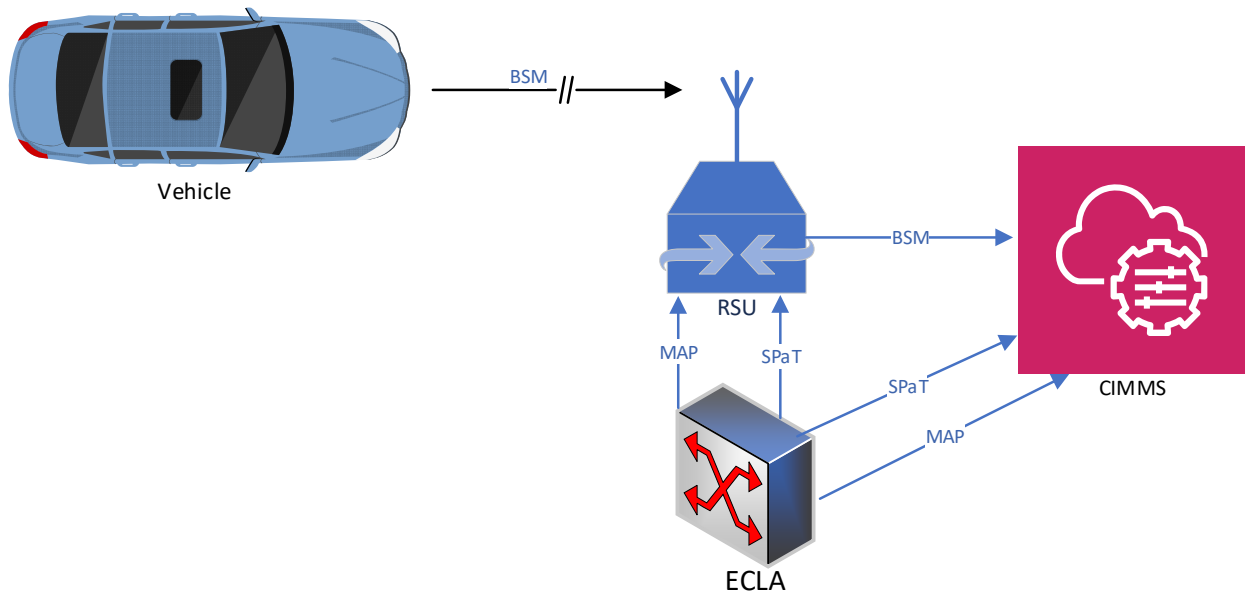


Figure 5. Utah DOT CIMMS Roadside Architecture



2.3 USER INTERFACE

2.3.1 CIMMS Dashboard

The CIMMS dashboard (shown in Figure 6 below) is the main landing page within the CIMMS web application. The graphs on the page depict data from Assessments within the CIMMS system. The table at the bottom of the page shows all of the active notifications for a given intersection.



Figure 6. CIMMS User Interface – Dashboard / Landing Page

2.3.2 Map View

The Map view screen (shown in Figure 7 below) provides users a way to graphically view and debug issues in the SPaT and MAP messages of an intersection. The view can be adjusted temporally to show the intersection states at different times and can also show vehicles that passed through the intersection during the specified time range.

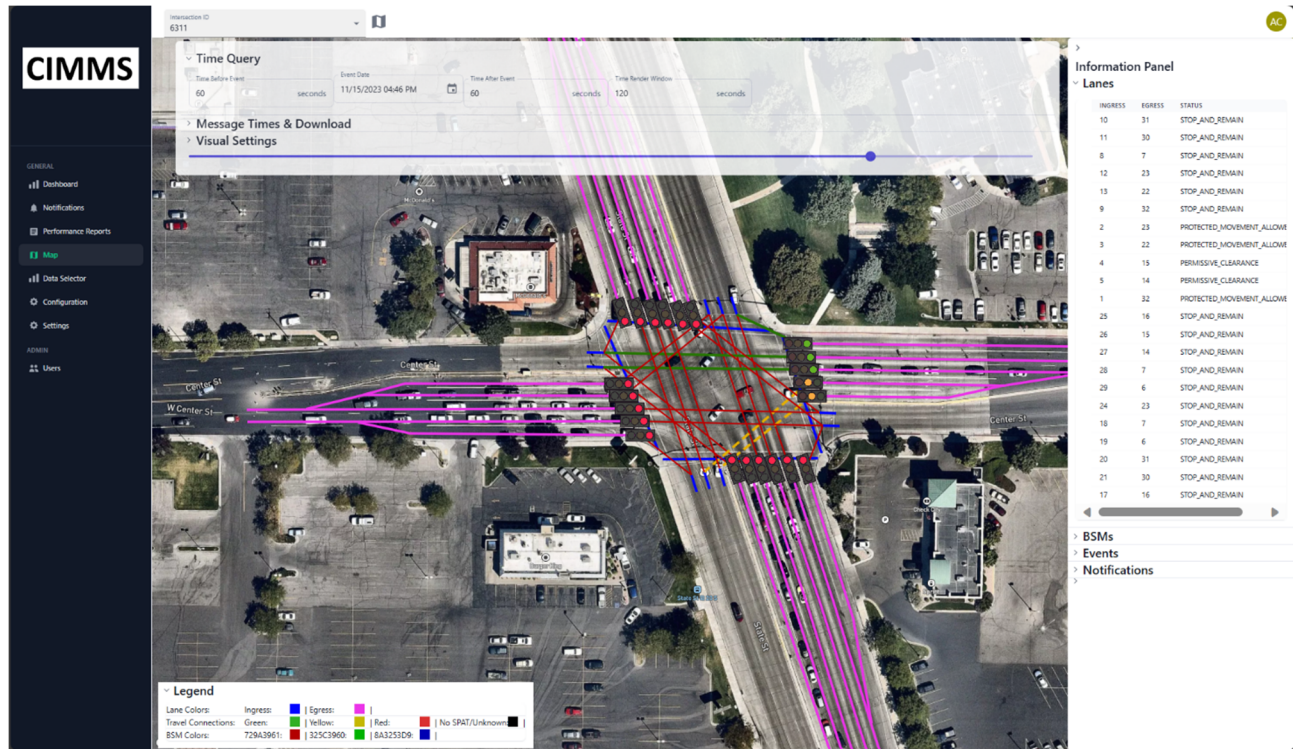


Figure 7. CIMMS User Interface – Map View



2.3.3 Data Selector

The data selector (shown in Figure 8 below) provides the ability for users to query and download historical events and assessments. Historical events and assessments are returned in a human readable JSON format for further offline analysis.

The screenshot displays the CIMMS Data Selector interface. On the left is a dark sidebar with the CIMMS logo and a menu including Dashboard, Notifications, Performance Reports, Map, Data Selector (highlighted), Configuration, Settings, and Users. The main area is titled 'Query' and contains a form with the following elements: an 'Intersection ID' dropdown set to '6311', an 'Event Type' dropdown set to 'All', a 'Start Date' field showing '11/15/2023 04:41 PM', a 'Time Range' field set to '0 minutes', and a list of event types with checkboxes: All, ConnectionOffTravelEvent, IntersectionReferenceAlignmentEvent, LaneDirectionOffTravelEvent, SignalGroupAlignmentEvent, SignalStateConflictEvent, SignalStateEvent, SignalStateStopEvent, TimeChangeDetailsEvent, MapMinimumDataEvent, SpatMinimumDataEvent, MapBroadcastRateEvent, and SpatBroadcastRateEvent. At the bottom of the form are two buttons: 'Query Data' and 'View Counts'.

Figure 8. CIMMS User Interface – Data Selector

2.3.4 Configuration Page

The Configuration page (shown in Figure 9 below) allows CIMMS admins to change properties of the Message Processing/Event Generation algorithms. Updates here can be used to change notification thresholds and modify parameters in how various CIMMS algorithms operate.

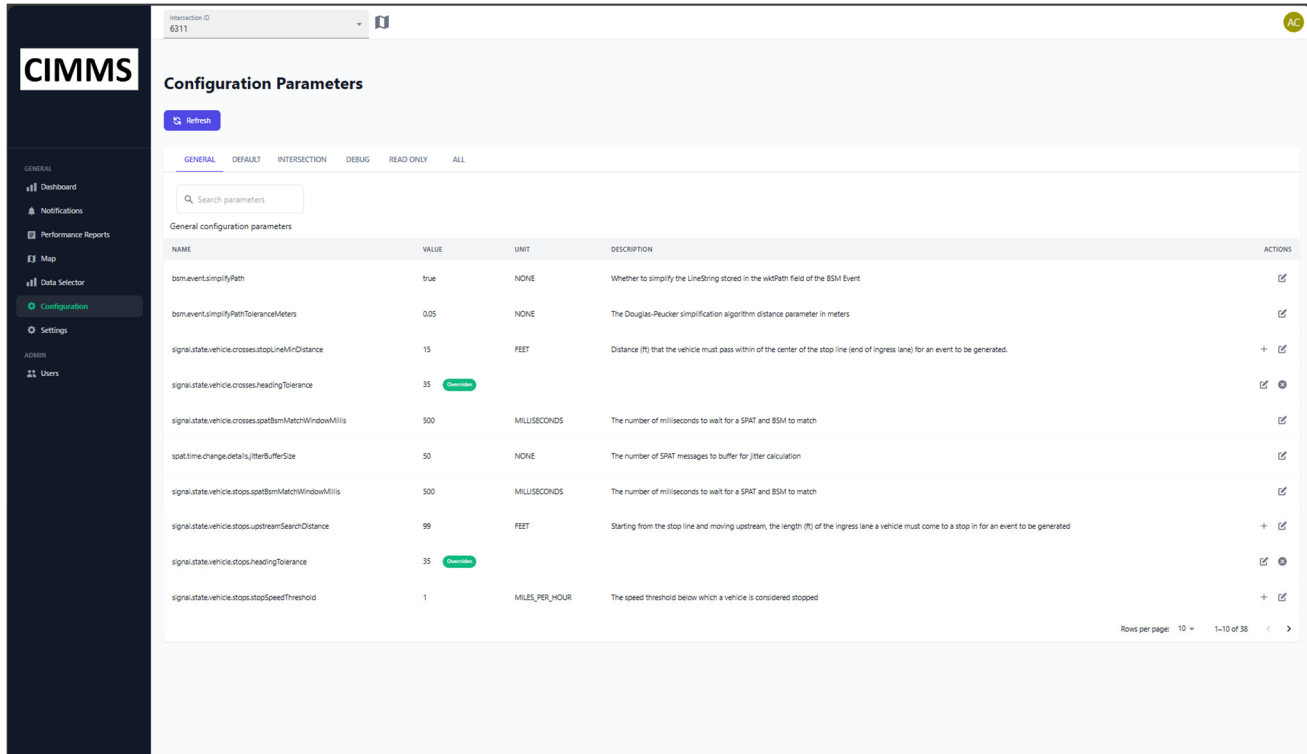


Figure 9. CIMMS User Interface – Configuration Page

2.4 SYSTEM REPORTS

The CIMMS is able to generate summary reports that aggregate event and assessment history. The purpose of these reports is to reveal longer term trends and allow users to validate the effects of updates on intersections. Reports are automatically generated for each day, week, and month and are available for users to download from the CIMMS User interface. Users may also request reports be generated for an arbitrary time range. Reports are designed for easy analysis and contain many graphs and charts to show information about how the intersection is performing. The reports currently contain the following:

- Lane Direction of Travel Event Summary
 - Number of Lane Direction of Travel Events per day
 - Distribution of Distance from centerline
 - Distribution of Vehicle heading error
 - Breakdown of distance from centerline over time for each lane
 - Breakdown of vehicle heading error over time for each lane
- Connection of Travel Event Summary



Final Report

- Number of Connection of Travel Events per day
 - Heatmap of ingress and egress lane pairings
- Signal State Events
 - Number of Stop Line Stop Events per day
 - Number of Stop Line Passage Events per day
 - Number of Signal State Conflict Events per day
 - Number of Spat Time Change Details Events per day
- Intersection Reference Alignment Events
 - Number of Intersection Reference Alignment events per day
- MAP
 - Number of MAP Broadcast Rate events per day
 - Number of MAP Minimum Data events per day
 - List of missing MAP data elements
- SPAT
 - Number of SPaT Broadcast Rate Events per day
 - Number of Spat Minimum Data Events per day
 - List of missing SPaT data elements

2.5 OPERATIONS AND MAINTENANCE

Both the Maricopa County and UDOT deployments are run in Docker images run on three VM's. Left alone, the CIMMS system is designed to operate without significant maintenance. Except for periodic software updates, and manual log cleanup.

2.5.1 Software Updates

To update a CIMMS instances to the latest version of the software, begin by logging into each VM in the CIMMS system. From each VM pull the latest changes from its respective repository from the usdot-jpo github repository. Each repository and its associated component are listed below.

- Ingest VM, jpo-ode, <https://github.com/usdot-jpo-ode/jpo-ode>
- Database VM, jpo-conflictmonitor, <https://github.com/usdot-jpo-ode/jpo-conflictmonitor>
- Application VM, jpo-conflictvisualizer, <https://github.com/usdot-jpo-ode/jpo-conflictvisualizer>

The latest changes can be downloaded by running ```` git pull ````

2.5.2 Routine Maintenance

It is recommended that CIMMS operators routinely log into the CIMMS system and verify that all of the docker images are running and operational. During testing the most common failure of the CIMMS system is filling up all available disk space with logs or data from the CIMMS instance. This was particularly a problem for the UDOT ingest VM, which frequently filled up the 32GB available disk space with Kafka Logs (Recommended system storage requirements have been increased to 50GB for this reason). When logged into the system, verify that there is plenty of available disk space on the VM. If

no disk space is available, consider clearing out old Kafka logs from the Kafka VM in the ODE. Similarly, the Database VM may also fill up all available storage as the database grows. While the CIMMS proof-of-concept system is configured to automatically delete data over time, it is still possible to overrun available drive space if the rate of data creation increases significantly before old data has a chance to get cleared out.

2.5.3 User Support and Management

The CIMMS UI offers limited support for user management and allows for creation and deletion of users. This is sufficient for basic use cases, but is not a comprehensive solution to all possible user management and administrative needs. For example, if an admin wants to reset a user's password, implement a password rotation policy, or change a user's role in the system, they will need to use the CIMMS keycloak admin dashboard. The dashboard is accessible here:

https://<VM_IP_ADDRESS>/admin



3. Project Tasks

3.1 PHASE I

3.1.1 Task 1: Program Management

A kick-off meeting was held on February 5, 2022. A project management plan (PMP) that provided a description of the proposed work, methodology and approach for completing work, schedule, proposed staffing, and contact information proposed channels of communication, and deliverable due dates was developed. The draft PMP was submitted to the CV PFS panel on March 25, 2022. Comments made by PFS members were used to refine the draft content. The final version of the PMP was submitted to the PFS panel on May 2, 2022. Progress reports were delivered on a monthly basis throughout the course of the project, and monthly status calls were also held.

3.1.2 Task 2: Stakeholder Engagement

Note: This task was added to the initial scope provided for this project, and was a precursor to developing the ConOps and System Requirements.

A Stakeholder Engagement Plan (SEP) was developed to guide the level of involvement of the identified stakeholders. Stakeholder engagement for each task was identified, and the preliminary schedule of meetings and interactions with the stakeholders was provided. Three stakeholder engagement efforts were ultimately undertaken during the systems engineering development for the CIMMS, as described below:

- **Survey.** During the development of the draft ConOps, an online survey was developed to get feedback on potential use cases for the proof of concept CIMMS. The survey was distributed to PFS members as well as a list of external stakeholders. During the survey timeframe (May 20 through June 6, 2022), 38 responses were received. The survey results were used to validate the user needs in the ConOps. A summary of survey responses was provided in an Appendix in the ConOps.
- **ConOps Webinar.** After the draft version of the ConOps was completed, a ConOps webinar was conducted on June 24, 2022. This webinar was open to both PFS members and external stakeholders. A high-level walkthrough of the ConOps and survey results was provided, and attendees were provided an opportunity to ask questions and provide input that could be used to shape the final version of the ConOps.
- **System Requirements Webinar.** After the draft version of the System Requirements was completed, a System Requirements webinar was conducted on August 12, 2022. This webinar was open to both PFS members and external stakeholders. A walkthrough of select system requirements was provided, and attendees were provided an opportunity to ask questions and provide input that could be used to shape the final version of the System Requirements.

3.1.3 Task 3: Concept of Operations

The first step in the systems engineering documentation for the CIMMS project was the development of a Concept of Operations (ConOps). The ConOps provided a general description of the current system, a justification for changes, and provided the concept for the CIMMS within the context of the existing system. The description of the CIMMS included several detailed algorithms that indicated explicitly how data inputs would be processed to generate outputs that would be useful for CIMMS users in determining potential issues with V2X messages. While this level of detail is not typically provided in a ConOps, it was deemed necessary for the CIMMS project to demonstrate that the CIMMS concept was feasible, providing the CV PFS members with confidence to proceed with the next steps of developing system requirements, and system development activities. The ConOps also provides use cases and scenarios that describe how the system operates from the perspective of system users, and demonstrates how the system provides anticipated system benefits. The draft ConOps was submitted to the CV PFS panel on June 15, 2022.

Once the draft ConOps was completed, a ConOps webinar was held to provide a stakeholder walkthrough and to elicit stakeholder feedback (see section 3.1.2). This feedback along with comments in the document made by PFS members was used to refine the draft content. The final version of the ConOps was submitted to the PFS panel on September 16, 2022. Responses to individual comments within the document were captured in a comment resolution table.

The ConOps was updated twice after the submittal of the final version. The first update (November 18, 2022) was provided to incorporate an additional algorithm for determining the signal event state when a vehicle came to a complete stop at the intersection. This algorithm had not been initially considered, but was included to compliment events generated by the stop line passage algorithm. A second update (August 31, 2023) was provided to correct a few typos that were identified.

3.1.4 Task 4: System Requirements

The second step in the systems engineering documentation for the CIMMS project was the development of the System Requirements. The System Requirements provided a detailed description of the CIMMS, user characteristics, assumptions, constraints, and detailed system requirements. Requirements types include: functional, performance, interface, data, security, policy and regulation, non-functional, and enabling requirements. Each requirement was associated with a functional group, subcomponent, user needs, and a verification method type. Reverse traceability was also provided to check that all user needs were accounted for in the requirements. Systems engineers developing the requirements worked closely with the development team to verify that the intent of each requirement was being clearly communicated in the text of the System Requirements Document. The draft System Requirements was submitted to the CV PFS panel on July 15, 2022.

Once the draft System Requirements document was completed, a System Requirements webinar was held to provide a stakeholder walkthrough and to elicit stakeholder feedback (see section 3.1.2). This feedback along with comments in the document made by PFS members was used to refine the draft content. The final version of the System Requirements was submitted to the PFS panel on November



18, 2022. Responses to individual comments within the document were captured in a comment resolution table.

3.2 PHASE II

Upon review of the systems engineering documentation developed as part of Phase I of the CIMMS project, the CV PFS panel elected to continue moving forward with Phase II of the project which included additional systems engineering documentation, development of the CIMMS proof of concept system, as well as demonstration, evaluation, and reporting tasks.

3.2.1 Task 5: Prototype Development

The next step was the development of the software design document, system integration design (SID) document, and the system test plan.

The Software Design Document (SDD) provided a description of the design of the CIMMS such that it allowed for software development to proceed with an understanding of what is to be built and how it is expected to be built. The SDD defined the design stakeholders and how each was expected to interact with and/or benefit from the CIMMS. It also defined the internal architecture, defined the internal and external interfaces (including a detailed user interface design), the structures of data types stored on the CIMMS, algorithms, and risks and contingencies. Minimum system requirements were also provided. The draft SDD was submitted to the CV PFS panel on February 22, 2023. Comments in the document made by PFS members were used to refine the draft content. The final version of the SDD was submitted to the PFS panel on April 4, 2023. Responses to individual comments within the document were captured in a comment resolution table.

The System Integration Plan conveyed specifics of how the Connected Intersection Message Monitoring System (CIMMS) prototype was deployed in each selected environment (Maricopa County DOT and Utah DOT). The draft System Integration Plan was submitted to the CV PFS panel on May 19, 2023. Comments in the document made by PFS members were used to refine the draft content. The final version of the SID was submitted to the PFS panel on June 14. Responses to individual comments within the document were captured in a comment resolution table. An update to the SID was provided on November 28, 2023 to account for a few items that were not known at the time of the submission of the final version.

Finally, a test plan was developed to document test testing approach and the detailed test procedures to be followed during testing to determine if the developed CIMMS was performing as intended. The draft test plan was submitted to the CV PFS panel on July 27, 2023. Comments in the document made by PFS members were used to refine the draft content. The final version of the test plan was submitted to the PFS panel on August 31, 2023. Responses to individual comments within the document were captured in a comment resolution table.

The ConOps, System Requirements, and Software Design were used by developers when developing the CIMMS. Regular meetings between systems engineers and developers were held to provide progress updates, as well as to provide feedback to developers. As functionality was implemented,

testing was performed to verify if requirements and design documentation were being met. Several rounds of preliminary testing and code updates were undertaken to refine functionality and overall efficiency of processes undertaken by the CIMMS. The CIMMS is an open-source software, and is available free to download on GitHub. Links to the three software components required to run the CIMMS are provided below.

- Ingest VM, jpo-ode, <https://github.com/usdot-jpo-ode/jpo-ode>
- Database VM, jpo-conflictmonitor, <https://github.com/usdot-jpo-ode/jpo-conflictmonitor>
- Application VM, jpo-conflictvisualizer, <https://github.com/usdot-jpo-ode/jpo-conflictvisualizer>

At the end of the development and testing process (December 2023), the development team had budget to support additional development activities. Discussions were held between CV PFS panel members, systems engineers, and developers on December 6, 2018 to discuss various functionality updates and new functionality for the CIMMS (discussed in Section 4). As of January 2023, the development team had made updates to functionality related to the data storage scheme, the application programming interface (API), and the user interface (UI).

As of the writing of this final report, the updated data storage scheme reduced the data storage rate by about 67 percent from the initially-developed CIMMS. The reduction of the data storage rate allows for lower overall required storage (which results in reduced cost) or alternatively, allows data to be held in storage over a longer timespan before being overwritten.

API updates allows the application VM to connect directly to Kafka to get a direct feed of messages, and implements a Stomp WebSocket broker for broadcasting intersection messages live. The UI was updated to connect to a WebSocket broker to receive and display live messages. This added functionality allows users to actively monitor SPAT and MAP messages broadcast from RSUs, and begins to show the potential for the CIMMS to not only be used as means of monitoring and assessing message content, but as a CV system management tool.

It is important to note that development activities that took place after the initial development and testing process are not captured in systems engineering documentation – described in Section 3.1.3, Section 3.1.4, and the beginning of this section (Section 3.2.1).

3.2.2 Task 6: Prototype Demonstration and Evaluation

The System Integration Plan was used in to inform the integration process. Most integration activities took place virtually. This included the provisioning of virtual machines, VPN access (Maricopa County only), installation of the CIMMS software on the VMs, and configuration of the RSUs to forward CV messages to the CIMMS (data ingest subsystem). The System Integration Plan was updated based on the experience gained during the integration process.

Once CV messages were being ingested by the CIMMS, system testing was performed. Testers created accounts for each CIMMS deployment, and were able to utilize the user interface to visualize messages, view event data, and view assessment data. Data in SPaT, MAP, and BSMs were carefully reviewed to determine when events should be generated, and reviewed the data contained in events



and compared against raw SPaT, MAP, and/or BSMs to check the accuracy of the outputs. Testers also provided general feedback on the user interface to refine functionality and improve useability.

Field testing was performed in Anthem. This field testing served two primary purposes. First, BSMs were not being received by the Anthem CIMMS. The field test provided an opportunity for testers and developers to collaborate to identify the issue at hand and to make changes to the CIMMS to accommodate the observed issue. During the field test, testers were able to confirm that unsigned BSMs were being sent by OBUs, received by RSUs, and were being forwarded to the CIMMS. However, the CIMMS as initially designed did not account for the potential for unsigned BSMs to be forwarded from RSUs. Thus, the CIMMS did not properly identify the message or decode it. Developers were able to make modifications to the data ingest subsystem to accommodate unsigned BSMs. However, it is important to note that while acceptable for testing the proof of concept CIMMS, the ingestion and use of unsigned BSMs is undesirable in production-level CV systems.

Second, field testing in Anthem allowed testers to perform controlled drive-through tests. A data collection plan was developed prior to engaging in test activities in Anthem. This testing allowed testers to compare the observed vehicle trajectory against the BSM data visualized on the CIMMS, determine if the data in the events produced by the CIMMS is accurate based on BSM/SPaT/MAP messages, and finally if the final outcome of the event accurately represented the expected outcome based on the observed actual vehicle trajectory. This testing resulted in the refinement of algorithms that utilized BSMs, and also yielded insight into the potential implications of inaccurate BSM data – that is BSMs that do not accurately represent the vehicle’s actual trajectory affecting the accuracy of data in the events produced by the CIMMS.

Field testing was not performed in Orem, as there were BSM-producing vehicles in the area, and CV messages were being ingested by the CIMMS after the conclusion of integration activities. One important observation stems from the fact that certain types of vehicles were over-represented in BSM data. The design of CIMMS BSM-based algorithms is predicated on a sufficient portion of drivers behaving in an expected manner (e.g. stopping on a red signal indication, driving in the correct direction, entering/exiting the intersection using the correct lanes, etc.). Due to the deployment of CV-enabled priority/preemption applications, a significant portion of the equipped vehicles passing through Orem intersections are transit and emergency vehicles. When pulling into and out of transit stops, transit vehicles deviate from a typical driver’s expected motion along the lane of travel. This was quite apparent from the visualization of BSMs on the CIMMs. Because a significant portion of the equipped vehicles are transit vehicles, this resulted in the potential for the CIMMS to indicate potential issues with ingress and egress lane orientations in the MAP message via the ‘lane direction of travel’ events. Furthermore, there was a particular left turn movement where the CIMMS indicated a large portion of vehicles were crossing the stop line on red. Unlike the behavior of transit vehicles in the prior example, there was not a method of readily determining if the vehicles were emergency vehicles or not by looking at the BSM visualization on the CIMMS. However, using judgement regarding the progression of phasing (using SPaT and MAP visualization), the overall correct behaviors exhibited for all other movements at the intersection, and knowledge that OBU-equipped emergency vehicles are operating in the area, it is not a stretch to conclude that it is the behaviors of emergency vehicles (not an incorrect SPaT/MAP data) that is resulting in the aforementioned outcome.

A System Assessment Report was developed to provide a post-deployment record of overall findings, lessons learned, changes to the CIMMS, and other critical details for the two operational connected vehicle environments in Anthem, Arizona (operated by Maricopa County DOT), and in Orem, Utah (operated by Utah DOT). Most importantly, this document captures recommendations echoed by systems engineers, developers, and PFS members that provide direction for potential development activities moving forward. The draft System Assessment Report was submitted to the CV PFS panel on December 3, 2023. Comments in the document made by PFS members were used to refine the draft content. The final version of the System Assessment Report was submitted to the PFS panel on January 31, 2024. Responses to individual comments within the document were captured in a comment resolution table.

3.2.3 Task 7: Final Report

A Final Report (this report) was developed to provide a review of project tasks, discuss final outcomes, capture final strategic recommendations regarding the direction of the CIMMS moving forward, and concluding remarks for the CIMMS project. The draft Final Report was submitted to the CV PFS panel on February 2, 2024, with comments requested by 02/23/24. Comments provided by PFS members will be utilized to refine the document. The final version of the final report is anticipated to be submitted to the Panel by 03/08/24 along with a comment resolution sheet describing how each comment was addressed.

A final presentation to review the final report will be developed and provided to the FPS panel by March 22, 2024.



4. Final Outcomes

4.1 OUTCOMES

The following outcomes based on feedback from site leads is documented in the bulleted list below.

- Outputs from the CIMMS are generally useful. Site leads specifically indicated that the immediate feedback that certain algorithms provided were especially useful.
- The ability to visualize replayed SPaT, MAP, and BSM data is useful, regardless if the data was associated with a particular event or not.
- Once installed and integrated, it is relatively simple to add new intersections. The forwarding tables on the RSUs simply need to be adjusted to forward SPaT, MAP, and BSMs to the CIMMS IP address and port. The CIMMS will automatically handle the data being received from the RSU, and allow the user to immediately “see” the intersection in user interfaces.
- BSM data was generally good enough to determine if a lane was oriented in the correct direction and what lane the vehicle is in when it enters and exits the intersection (to determine the event state when a vehicle stopped or crossed the stop line and the connection the vehicle took).
- The actual impacts of position correction on BSM-based algorithm performance is not known. There is no method of determining if the latitude/longitude position data in the BSM has been corrected or not.

4.2 LESSONS LEARNED

The following lessons learned based on testing as well as feedback from site leads is documented in the bulleted list below.

- BSM data is not recommended for determining the positional accuracy of the lane centerline.
- The CIMMS proof-of-concept algorithms rely on drivers behaving in a typical fashion. The makeup of the population of CVs (producing BSMs that are received by RSUs and forwarded to the CIMMS) has the ability to impact results.
 - Buses pulling out of and back into a lane of travel (to serve a transit stop) did not always align well with lane geometries in the MAP message. Because buses comprised a significant portion of equipped vehicles, the CIMMS initially indicated potential issues with lane alignment. Algorithm parameters were loosened to account for this behavior, but this is not ideal for long-term operations.
 - A particular movement at an intersection saw a greater-than-expected percentage of vehicles crossing on a red signal indication. While this would typically point to a potential issue with SPaT and/or MAP, it could not be ruled out that this was the result of emergency vehicles that were over-represented for the turning movement.
- Notifications that can be provided in real-time (e.g. intersection reference alignment, event state conflict, time change details conflict) are far more preferable to notifications that require

a minimum number of events to occur (i.e. BSM-based events) before the notification is issued. This can especially be an issue when the penetration rate of CVs is low or non-existent.

- Site leads have also indicated the ability to view ingested data live in the visualization would be helpful for field testing and management purposes.

4.3 RECOMMENDATIONS FOR IMPROVEMENTS

Throughout the course of the design, development, deployment, and testing of the CIMMS, discussions between the CV PFS panel and the design and development team yielded a number of noteworthy improvements and considerations for future CIMMS development – detailed in the subsections below:

- **Additional Ground Truth Data Sources.** It should be considered if the CIMMS should interface with additional external systems so that it has access to additional ground truth data sources (in addition to the CV data that it currently ingests). These new sources of ground truth data have the potential to simplify and/or streamline the ability to perform checks of certain data elements. One potential external source of data that was discussed included data coming directly from traffic signal infrastructure including but not limited to the signal controller, conflict monitor, or other equipment that can sense the current signal state. However, this would need to be weighed against the increased complexity of development/integration/configuration activities that this would likely entail to account for potential variations in these data sources and ground conditions between sites.
- **Data Storage.** One limiting factor to deploying the CIMMS more broadly (adding RSUs) are its data storage requirements. Some measures have already been taken (see Section 3.2.1 for details) to reduce the storage requirements. Additional measures that could be taken include handling duplicated results, which typically result from the fact that the same event can be issued for each individual message. It is recommended that certain algorithms be performed on a periodic basis (e.g. every minute), and an aggregate result is provided in the output. De-duplication is expected to have the greatest impact on the following event types: Intersection Reference Alignment Event, Signal Group Alignment Event, SPaT/MAP Minimum Data Event, Signal State Conflict Event.
- **Data Overwrite.** The CIMMS proof-of-concept utilizes a time-based data overwriting scheme. This scheme has the potential for the CIMMS to overrun available drive space for data storage. Another recommended improvement would be to implement a memory-based data overwriting scheme, where data that has been stored the longest is deleted when a pre-defined storage limit is reached. Consideration should also be given to allowing different values to be provided for different data types. This would allow an agency to prioritize retention of certain types of data, depending on the specific data needs (and limitations) of the agency. For example, if limited storage capacity is available, and there is a high-priority need to store event and/or assessment data (e.g. for validation purposes), then the amount of memory for raw message storage could be reduced (since raw messages have a high memory requirement), allowing for increased amount of event and/or assessment data to be retained.
- **Additional Message Monitoring functionality.** There are various new algorithms that could be developed to check compliance with other CTI 4501 requirements. Should there be continued



interest in the CIMMS, it is recommended that stakeholder outreach be performed to prioritize the next phase of development activities. Potential items to consider include, but are not limited to: RTCM message requirements, event state transitions, revision counter, comparison of pedestrian movements against vehicular movements, subsequent movement event information, consideration of revocable/enabled lanes, more complex intersection geometries.

- **CV Device Health and Status Monitoring.** CIMMS deployment sites have indicated interest in the CIMMS providing CV device management functionality, which includes a device monitoring capability as well as monitoring of network connectivity between devices. Nagios is an example of a network device monitoring tool that can assist in detecting problems and mitigating future issues before they ultimately affect end users. Figure 10 and Figure 11 provide screenshots that demonstrate the types of visualizations that Nagios can provide for monitoring real-time and historical status of devices as well as network traffic between devices.

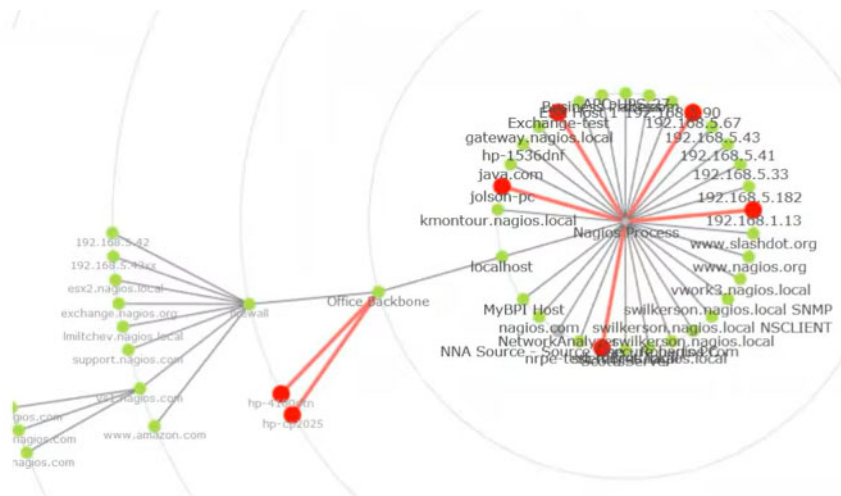


Figure 10. Real-Time Network Device Communications Monitoring (Nagios)

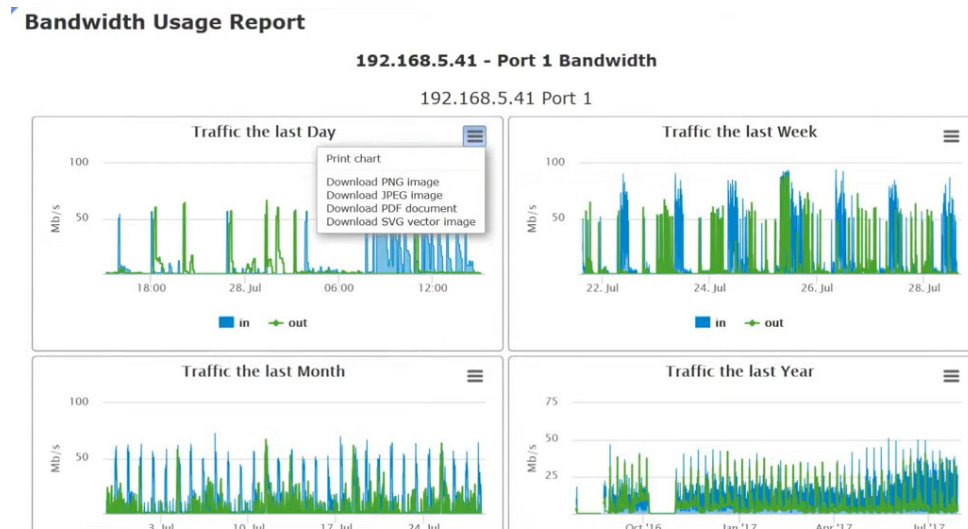


Figure 11. Networked Device Traffic History (Nagios)

- Performance Metrics and Operational Metrics.** The CIMMS could also be used to perform computations on CV data it ingests to generate real-time and/or historical aggregated performance metrics and operational metrics, such as message counts, CV traffic volumes (by lane, movement, etc.), speeds, or notifications for events (e.g. priority/preemption events, hard braking events, red light running rates, etc.).
- Alternative Architecture.** Alternative architectures should be considered and tested to determine if there may be a more cost-effective or efficient method for running the CIMMS. This could be an especially important consideration as the number of intersections scales up. One particular architecture that was discussed was utilizing roadside processing equipment at each intersection to perform initial data ingestion and processing. Systems that support the CIMMS database and user interface would remain centrally located to allow for a consolidated management interface.
- SCMS Validation for RSU Operations and OBU Misbehavior Detection.** The nature of the CIMMS is to assess the accuracy of data in CV messages. The logging of event and assessment data could be used as a basis for demonstrating some level of competency (or the absence of certain types of issues). While the current focus of the CIMMS is on the accuracy of CV messages produced by CV infrastructure, future versions may also assess the accuracy of data in messages produced by vehicles, such as the BSM. This is often referred to as misbehavior detection and may involve the use of sensing equipment on the roadside to produce ground truth information which can be used to assess accuracy of data in BSMs. When inconsistencies are noted, a misbehavior report could be sent to an SCMS.



5. Concluding Remarks

The CIMMS proof of concept was engineered and developed with SPaT and MAP interoperability in mind. CTI 4501 Connected Intersection Implementation Guide represents an industry consensus on the message requirements that must be met for production CVs to uniformly support advanced safety applications between jurisdictions. Industry leaders have had time to understand the requirements in this document and have shown interest in learning about the ability for their current system to conform to these requirements. There is a shared concern that one-time testing is not sufficient to determine if an intersection is adhering to CTI 4501 requirements. This type of testing cannot feasibly be performed over an extended period to determine if the intersection is continuously meeting these requirements. Furthermore, this type of testing will not readily identify issues with messages after changes to the roadway geometry or signal timing plan. Thus, the ability to continuously monitor CV messages from intersections in an automated fashion was conceptualized.

The CIMMS was developed as a proof-of-concept system focused on message monitoring – it considered a limited number of use cases and utilized relatively straightforward data analysis and data management techniques to demonstrate that such a system could be used to provide effective outputs for determining if certain data elements in SPaT and MAP messages is accurate.

At the time the CIMMS was conceptualized, there was a desire to minimize the complexity of systems that interface with the CIMMS to simplify the development and integration processes. This was sufficient for the demonstration of the CIMMS proof-of-concept. CIMMS users found value in the immediate feedback that certain algorithms provided. Algorithms that required multiple vehicle passages through the intersection to produce a reliable output were less desirable, as this generally took some amount of time. In order to provide more immediate feedback for these algorithms, other ground truth data sources are needed. One potential source of this data is the intersection conflict monitor. While typically used to detect and respond to improper or conflicting signal indications and improper operating voltages in a traffic controller assembly, the conflict monitor is a resource that can provide data regarding the current state for all traffic signal channels as well as the programmed channel conflicts. While this may increase the complexity of the CIMMS integration process (and configuration), users are likely to appreciate the immediate feedback that the CIMMS will be able provide using data from other external sources.

The CIMMS was initially designed to monitor for potential issues with SPaT and MAP messages over the long term, after initial testing was completed. However, users found that the immediate feedback provided by the proof-of-concept CIMMS was useful for initial testing activities as well. Given its success, CIMMS users and CV PFS observers began to consider other ways the processing capabilities of the CIMMS could be utilized to provide additional capabilities. The next generation of the CIMMS could consider a more complete list of use cases, use more elegant algorithms, or more efficiently store data. Furthermore, in addition to message monitoring, the platform that the CIMMS is built upon has the potential to perform other automated data-based tasks such as live message monitoring, RSU management/configuration, generation of operations data, and generation of performance measures.

With outputs from the proof-of-concept CIMMS in hand (along with the results of other testing activities performed on behalf of the CV PFS), CV system managers have become more aware of interoperability issues in SPaT and MAP message being broadcast from intersections. As the CIMMS functionality evolves and is able to more holistically assess compliance with CTI 4501 requirements, this awareness of interoperability issues will only continue to grow.

With this knowledge, CV system managers are starting to determine what changes need to be made to address compliance issues. Some issues can be resolved through a simple change such as a configuration modification, utilizing the tools that are currently available³. Other issues may require technically complex modifications to existing functionality or modifications to existing tools to comply with CTI 4501. To date, there has been minimal motivation on the behalf of vendors or developers to address these shortcomings⁴. It is expected that a dedicated engineering and development effort may be needed to demonstrate the feasibility of meeting more demanding requirements.

The need for the industry to make a concerted effort to make such changes is not new. The development of one of the more substantial advances in the CV space, the Traffic Signal Controller Broadcast Message (TSCBM, also known as the *SPaT Blob*), required a dedicated research and development team. This functionality has since become a standard output in modern-day traffic signal controllers and is still widely used to generate SPaT messages in today's CV deployments. Devising solutions to address issues pertaining to compliance with CTI 4501 will also likely require an open, collaborative, multi-disciplinary process, similar to the development of the SPaT Blob or the CIMMS system itself. Once a functioning solution is developed, it is anticipated that signal controller manufacturers, as well roadside processing device vendors (which may include, but is not limited to RSUs), can leverage the developments in their products to produce interoperable messages required to be compliant with CTI 4501.

At the same time, standards developers need to more carefully consider the practicality of making system changes and modifying operations to comply with CTI 4501, especially for the SPaT message. Traffic signal controllers have long catered to the needs of traffic engineers to provide ever-advancing traffic control capabilities at intersections. The types of data used internally within a traffic signal controller involves adherence to signal timing plans. It will have knowledge about aspects of operations such as the signal state of each phase, time windows for the next signal state (based on minimum end time, maximum end time, passage time, and vehicle detections), but may have limited knowledge about what the next phase will be or exactly when this transition will occur until immediately before the transition. The ability to accommodate concepts such as Assured Green Period (AGP), which alleviates this concern, will likely run into resistance from traffic engineers to preserve efficiency. AGP could negatively impact traffic flow at intersections that already experience congested or near-congested conditions. Furthermore, the traffic signal controller generally has no knowledge of the intersection geometry, if/when movements are protected or permissive, or which movements (i.e. connections in the MAP message) are controlled by which phase(s). More data than what is

³ The Connected Intersections Program (CIP) is currently evaluating issues that can be addressed by changes at this scale.

⁴ One notable exception to this is the USDOT update of the MAP generation tool, [ISD Message Creator](#), expected in 2024.

Engagement with USDOT staff indicates the population of certain CTI 4501-required elements that could not be accommodated by the existing tool may potentially be accommodated in the update.



immediately available to today's traffic signal controller is needed to populate a fully compliant CTI 4501 SPaT message.

That being said, there is a need to find a balance between what requirements are needed for interoperability, the capability of CV equipment and other roadside equipment to provide the information necessary to comply, and the practicality of implementing some of the concepts in CTI 4501. While it is within reason for compliance with CTI 4501 to eventually be achieved, it is highly recommended for standards developers to introduce multiple levels of interoperability within CTI 4501 to provide deploying agencies with a preliminary target that can be reached using today's technology while developments (both evolution of the CTI 4501 standard and roadside equipment functionality updates) continue to be made. It would be understood that intersections that meet the preliminary target requirements could provide the data necessary for a "basic" version of RLVW, and that the full intended scope of the RLVW application could not be reached until all requirements can be met.

Finally, it is important to consider the impacts of future releases of CTI 4501. Contributors to CTI 4501 are currently discussing updates, which will be made available in the next release sometime in 2024. These updates will provide further clarification and interpretation of requirements in the current version. It is too early to precisely determine what the exact changes will be at this time. Thus, once released, it will be important to consider if current CIMMS algorithms will need to be modified to properly align with the updated requirements. Furthermore, it is important to note that CTI 4501 currently only focuses on RLVW, and no other safety applications, though it is expected that interoperability guidance for other applications may not be far behind.

Appendix A. Version History

Table 1. Version History

Version Number	Date	Author(s), Agency	Summary of Changes
0.1	12/15/2023	WSP	Outline
1.0	1/31/2024	WSP	Initial Draft
2.0	2/29/2024	WSP	Final