Virtual Open Innovation Collaborative Environment for Safety (VOICES™) Distributed Testing Pilot Test 2

March 2026

Publication No. FHWA-HRT-25-053

U.S. Department of Transportation
Federal Highway Administration

Research, Development, and Technology
Turner-Fairbank Highway Research Center
6300 Georgetown Pike McLean, VA 22101-2296

Turner-Fairbank Highway Research Center

FOREWORD

The U.S. Department of Transportation (USDOT) initiated the Virtual Open Innovation Collaborative Environment for Safety (VOICES™) projects from 2020 to 2024 to foster a safe, interoperable, and cooperative transportation ecosystem using distributed testing. Distributed testing capabilities enable geographically distributed physical and simulated test assets to connect over a secure network and interactively test together in realtime. The project focused on developing and demonstrating distributed testing tools for government, industry, and academia to securely collaborate and integrate heterogeneous transportation systems. The two main components of the project were led by the Federal Highway Administration (FHWA), which identified applications for distributed testing, and the Office of the Assistant Secretary of Transportation for Research and Technology, which developed secure networking tools.

The FHWA Pilot 2 test campaign concluded initial efforts to demonstrate the feasibility of distributed testing and involved FHWA plus seven external research collaborators from government, industry, and academia testing various simulated and physical assets all across the United States. The test campaign's main objectives were to successfully cosimulate (connect multiple test sites securely and integrate various simulation platforms) and to demonstrate how distributed testing can potentially advance the technical maturity of applications, such as cooperative driving automation, while ensuring safe collaboration between organizations.

The research team overcame challenges during testing, such as coordinating schedules and solving technical issues between different systems, through careful planning and collaboration. These tests showed how potential problems could be identified earlier in development, thus improving system-of-systems safety in a cost-effective manner. Over two years of experiments, FHWA confirmed that distributed testing is a valuable tool for the surface transportation sector, helping to advance connected transportation systems through collaboration. This report outlines the methodology, outcomes, and potential for further advancements in distributed testing for connected surface transportation applications.

John Harding
Director, Office of Safety and Operations
Research and Development

Recommended citation: Federal Highway Administration, *Virtual Open Innovation Collaborative Environment for Safety (VOICES™) Distributed Testing Pilot Test 2* (Washington, DC: 2025) https://doi.org/10.21949/1521554

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. FHWA-HRT-25-053	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Virtual Open Innovation Collaborative Environment for Safety (VOICES) Distributed Testing Pilot Test 2		5. Report Date March 2026	
		6. Performing Organization Code:	
7. Author(s) Andrew Loughran (ORCID: 0009-0006-1572-9322), Danielle Chou (ORCID: 0009-0001-6134-1971), and Robin Laqui (ORCID: 0009-0004-8967-0699)		8. Performing Organization Report No.	
9. Performing Organization Name and Address Leidos 1750 Presidents Street Reston, VA 20190		10. Work Unit No.	
		11. Contract or Grant No. Contract 693JJ321D000010 / Task order 693JJ322F00260N	
12. Sponsoring Agency Name and Address Office of Safety and Operations Research and Development Federal Highway Administration 6300 Georgetown Pike McLean, VA 22101-2296		13. Type of Report and Period Covered Technical Report; September 2023–May 2025	
		14. Sponsoring Agency Code HRTM-30	
15. Supplementary Notes The contracting officer's representative was Danielle Chou (HRSO-50; ORCID 0009-0001-6134-1971).			
16. Abstract This report documents the results and technical performance analysis of the distributed testing pilot test 2 (Pilot 2) for the FHWA Virtual Open Innovation Collaborative Environment for Safety (VOICES) project. This pilot connected multiple test sites over a secure network and demonstrated how different models and simulation platforms can cosimulate together. This testing showcased how distributed testing can potentially improve the technical maturity of different applications and help ensure that algorithms and software from a variety of sources can work together safely to answer research questions important to the government, industry, and academia.			
17. Key Words Distributed testing, VOICES, TENA, simulation, cosimulation, CARLA®, cooperative driving automation		18. Distribution Statement No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22161. https://www.ntis.gov	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 102	22. Price N/A

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

SI* (MODERN METRIC) CONVERSION FACTORS				
APPROXIMATE CONVERSIONS TO SI UNITS				
Symbol	**When You Know**	**Multiply By**	**To Find**	**Symbol**
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yard	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
		NOTE: volumes greater than 1,000 L shall be shown in m^3		
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2,000 lb)	0.907	megagrams (or "metric ton")	Mg (or "t")
		TEMPERATURE (exact degrees)		
°F	Fahrenheit	5 (F-32)/9 or (F-32)/1.8	Celsius	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m^2	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa
APPROXIMATE CONVERSIONS FROM SI UNITS				
Symbol	**When You Know**	**Multiply By**	**To Find**	**Symbol**
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.314	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or "metric ton")	1.103	short tons (2,000 lb)	T
		TEMPERATURE (exact degrees)		
°C	Celsius	1.8C+32	Fahrenheit	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m2	0.2919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	2.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

*SI is the symbol for International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380. (Revised March 2003)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ANL	Argonne National Laboratory
API	application programming interface
AV	automated vehicle
BSM	basic safety message
CAV	connected and automated vehicle
CDA	cooperative driving automation
CDF	cumulative distribution function
DCN	data collection note
EM	execution manager
ETSI	European Telecommunications Standards Institute
FHWA	Federal Highway Administration
IMCP	Internet control message protocol
IP	Internet protocol
IPsec	IP security
ORNL	Oak Ridge National Laboratory
OST-R	Office of the Assistant Secretary of Transportation for Research and Technology
PSA	Personal Signal Assistant
ROS	Robot Operating System
SDO	Stateful Distributed Object
SDSM	Sensor Data Sharing Message
SIT	systems integration test
SPaT	signal phase and timing
SR	solo run
SRC	source
STOL	Saxton Transportation Operations Laboratory
TCP	Transmission Control Protocol
TDCS	TENA Data Collection System
TENA OM	TENA Object Model
TENA	Test and Training Enabling Architecture
TL	traffic light
TSC	traffic signal controller
TTS	Traffic Technology Services
UCLA	University of California, Los Angeles
UDP	user datagram protocol
USDOT	United States Department of Transportation
V2X	vehicle-to-everything
VOICES	Virtual Open Innovation Collaborative Environment for Safety
VPN	virtual private network

EXECUTIVE SUMMARY

The United States Department of Transportation (USDOT) recognizes that the transportation ecosystem is evolving, with increasingly complex systems and a multitude of actors entering the market. Many of these actors come from different sectors (e.g., automotive, infrastructure owner or operator, technology, software, etc.) and have different technical vocabularies and differing technical objectives. Thus, challenges in coordination and cooperation of transportation elements can arise. To facilitate the development of a safe, interoperable, and increasingly cooperative and coordinated transportation ecosystem, USDOT executed two Virtual Open Innovation Collaborative Environment for Safety (VOICES™) projects from 2020 to 2024. These projects sought to establish a mixed-reality, geographically distributed testing capability for potential surface transportation users across government, industry, and academia.

Distributed testing enables multiple organizations to plug simulated or physical test assets into a secure network from any location and interact with other users' systems in realtime. This approach allows users to safely interact and integrate with black box systems, thereby protecting intellectual property while reducing barriers to collaboration.

The first VOICES project, overseen by the Office of the Assistant Secretary of Transportation for Research and Technology (OST-R), focused on building the secure networking tools used for distributed testing. Meanwhile, the second VOICES project, overseen by the Federal Highway Administration (FHWA), focused on identifying and demonstrating applications that can potentially benefit from distributed testing. These demonstrations resulted in a greater understanding of distributed testing capabilities and created software tools that could be used by others interested in distributed testing.

Pilot 2 marks the conclusion of the VOICES proof-of-concept project overseen by FHWA. The Pilot 2 test campaign consisted of three test events jointly conducted by FHWA with seven external research collaborators from government, industry, and academia. The research team executed the test events with a mix of existing simulated and physical test assets developed and owned by the various research collaborators. Researchers categorized the main goals of Pilot 2 into cosimulation and cooperative driving automation (CDA) application objectives.

The cosimulation goals were primarily to connect multiple test sites over a secure network and demonstrate how different models and simulation platforms can cosimulate together. The application goals were centered on accelerating the technical maturity of CDA while supporting the use of standard messages to ensure multiple organizations can work together safely. Overall, the research team designed Pilot 2 to be the most complex set of distributed test events executed to date in the transportation sector.

During the test campaign's development, researchers encountered and resolved many challenges. These lessons learned are comprehensively outlined in this report. One of the main challenges was simply scheduling time with each participating organization to meet for development and actual testing. This challenge was overcome by planning well in advance to accommodate busy schedules and reserve time for testing.

Test event participants and their development teams also encountered and solved individual technical challenges related to systems interacting with one another. When participants test with other systems, assumptions can be challenged, and components can be tested more thoroughly than when testing alone. Initially staging these distributed tests in a cosimulation environment exposed technical challenges that otherwise might not have been discovered until much later in the development process, potentially causing participants to save on costs by expediting necessary testing and design changes.

FHWA has conducted nearly two years of distributed testing experiments and has proven that the capabilities are applicable and extensible to the surface transportation sector. This report outlines the objectives, methodologies, and outcomes of the Pilot 2 test campaign. Additionally, the report identifies how distributed testing may advance understanding of connected surface transportation applications, such as CDA applications, to help mature a connected surface transportation ecosystem with government, industry, and academic participation.

CHAPTER 1. PILOT 2 OVERVIEW

PURPOSE

This report documents the Virtual Open Innovation Collaborative Environment for Safety (VOICES™) Distributed Testing Pilot 2 test campaign (referred to as Pilot 2) executed and overseen by the Federal Highway Administration (FHWA) from Fall 2023 to Spring 2024.[1] This report details Pilot 2 test objectives, methodologies, results, assessments against objectives, and lessons learned. This report also discusses analysis results in the context of this test and FHWA's broader history of distributed testing experience.

BACKGROUND

VOICES Program

To facilitate the development of a safe, interoperable, and more cooperative and coordinated transportation ecosystem, the U.S. Department of Transportation (USDOT) executed two VOICES projects from 2020 to 2024.[1] These projects sought to establish a mixed-reality, geographically distributed test capability that allows multiple entities to simultaneously cosimulate and test together to collaboratively solve problems.

Distributed testing enables multiple organizations to plug simulated or physical test assets anywhere in the world into a secure network and interact with other organizations' systems in realtime. This approach allows users to safely interact and integrate with black box systems, thereby protecting intellectual property while reducing barriers to collaboration.

USDOT's dual VOICES projects were executed by FHWA and OST-R. The VOICES project overseen by OST-R focused on building the secure networking tools used for distributed testing and was executed by the MITRE Corporation, who has independently operated and maintained the VOICES-branded networking portal since September 2024.[2]

Meanwhile, the Saxton Transportation Operations Laboratory (STOL) executed the FHWA VOICES project and focused on identifying and demonstrating applications that would benefit from distributed testing on the VOICES portal.[2,3] These demonstrations resulted in a greater understanding of distributed testing capabilities and created software tools that could be used by transportation entities interested in distributed testing.

This report focuses on the VOICES portal rather than the VOICES projects. FHWA continues to maintain an interest in distributed testing to solve real-world integration and interoperability issues for new and emerging technologies and systems.

Distributed Testing Technologies

Distributed testing enables multiple sites to simultaneously and interactively cosimulate and test together in realtime using a mix of physical and simulated test elements to solve problems collaboratively.

Distributed testing requires four components, as follows:

- A secure test network to connect all test collaborators.
- A common language and interfaces that users can develop adapters for and use with any other users on the platform.
- A common scenario that helps all participants define a shared reality. This scenario definition can include digital maps, configuration files, and input files.
- Test elements, which can be physical or simulated elements under human or simulated control. All elements must have digital representations during the testing.

FHWA's distributed testing has focused on integrating system components and demonstrating applications that can potentially benefit from distributed testing capabilities. These demonstrations have sought to illustrate four basic capabilities, as follows:

- To show that different models, tools, and simulation platforms can successfully interact in realtime within a common cosimulation environment. This capability allows users to leverage existing tools and investments to reduce development resources when collaborating with others.
- To advance understanding of connected surface transportation applications by identifying and resolving conflicts between road elements. This capability can help researchers and developers iterate faster among themselves and may strengthen, inform, and potentially accelerate the development of norms and standards.
- To collect digital assets (maps, scenarios, and models) that can be shared across the surface transportation community. This capability reduces research barriers to entry, especially during collaboration.
- To strengthen relationships across government, academia, and industry. Working closely with a variety of research collaborators can provide improved mutual understanding of emerging technologies and enable each entity to play to respective strengths and roles rather than fulfill all research roles.

Previous FHWA Distributed Testing Demonstrations

Prior to Pilot 2, FHWA conducted several basic demonstrations of distributed testing capabilities. These demonstrations included the systems integration test 1 (SIT-1), conducted in Fall 2022, and pilot 1, conducted in Spring 2023.[4,5] FHWA conducted pilot 1 with research collaborators from Econolite®; Nissan; and the University of California, Los Angeles

(UCLA).[5] The test was conducted using a commercially available cloud-based network tool. Econolite and UCLA ran their respective traffic signal controller (TSC) and vehicle models, whereas FHWA's STOL team ran the CARMA Platform℠—a full-stack, CDA software—as software in the loop in the CARLA® simulation environment, demonstrating an array of models.[6,7] Pilot 1 was the first USDOT distributed testing event to include heterogeneous models from multiple organizations.[5]

PILOT 2 TEST CAMPAIGN

FHWA executed Pilot 2 with seven external organizations: Argonne National Laboratory (ANL), Econolite, Mcity, Oak Ridge National Laboratory (ORNL), Traffic Technology Services® (TTS), UCLA®, and Volkswagen (VW)®. The test campaign consisted of three individual test events; researchers collaboratively crafted all the events, with input from all collaborators.

Built on the technical foundation laid by pilot 1, the Pilot 2 test campaign significantly increased the complexity of tests performed to date. Pilot 2 marked the first time a mix of simulated and physical test assets interacted in realtime across multiple sites. The number of connected sites doubled compared to pilot 1, and the most complex of the test events connected five heterogeneous connected and automated vehicle (CAV) models and two physical test assets. Meanwhile, Pilot 2 also motivated the team to streamline the overall execution of distributed testing to better coordinate with so many test collaborators. Additionally, the research team added methods for improving the ease of connectivity, increasing data throughput, and consistency of test execution during Pilot 2.

Test Objectives

The primary test objectives for Pilot 2 consisted of two areas, as follows:

- Cosimulation objectives:
    - Connect multiple test sites using a cloud-based network.
    - Demonstrate how different models and simulation platforms can work together.
    - Identify areas where having simulation standards can potentially be beneficial: Develop a library of digital test assets, including scenarios, maps, adapters, and analysis tools.
- Application objectives:
    - Accelerate the technical maturity of safety- and efficiency-enhancing CDA applications.
    - Test different algorithms from various sources to ensure these algorithms can work together safely. Identify scenarios where deconfliction is needed.
    - Support the use of standard message formats.

- Work together to define a shared set of research questions important to government, industry, and academia.

Test Events

Pilot 2 consisted of three distinct test events. Each event sought to answer separate research questions, and the events were designed to collect data to help answer the questions. Following are more details:

- Event 0: Connectivity test:
    - Research question: Can multiple sites simultaneously cosimulate and interact with simulated vehicles controlled by microscopic traffic simulation across a cloud-based network?
    - Challenge: Connect and establish a concept of operations for running a test across eight sites located across the United States and even by remote operation from the United Kingdom.
- Event 1: Interoperable vulnerable road user sensing message test:
    - Research question: How compatible are two different vulnerable road user-related message standards?
    - Challenge: Lay groundwork to study the performance impacts of disparate U.S. and European cooperative perception messaging standards.
- Event 2: Collaborative ecodriving test:
    - Research question: Does surrounding automated vehicle (AV) traffic challenge the assumptions each vehicle makes when optimizing for ecodriving?
    - Challenge: Integrate existing tools that researchers have already invested in and developed, as opposed to completely rewriting models in a new simulation environment.
    - Challenge: Successfully cosimulate with multiple independently developed vehicle models (UCLA's OpenCDA, Mcity's TeraSim™, Mcity's Autoware®-based teleoperated live vehicle, FHWA's CARMA Platform, an ANL vehicle on a chassis dynamometer test stand, and ORNL's CARLA-based model). (See references 6–10.) Such testing reflects the varied ecosystem into which a multitude of actors and stakeholders must integrate products.
    - Challenge: Accelerate the use of operating norms and standards to facilitate interoperability.

- Challenge: Collect simulated and physical test data to assess the energy performance of each ecodriving algorithm. Demonstrate the process by which this evaluation may be done.

Researchers conducted these tests successfully in less than 6 mo. An Mcity-hosted webinar [1] in May 2024 offered an overview and live demonstration.(11) A crucial impact of Pilot 2 was introducing collaborators to the potential of distributed testing for the facilitation of collaborative development and emerging technology integration to improve transportation system safety and efficiency.

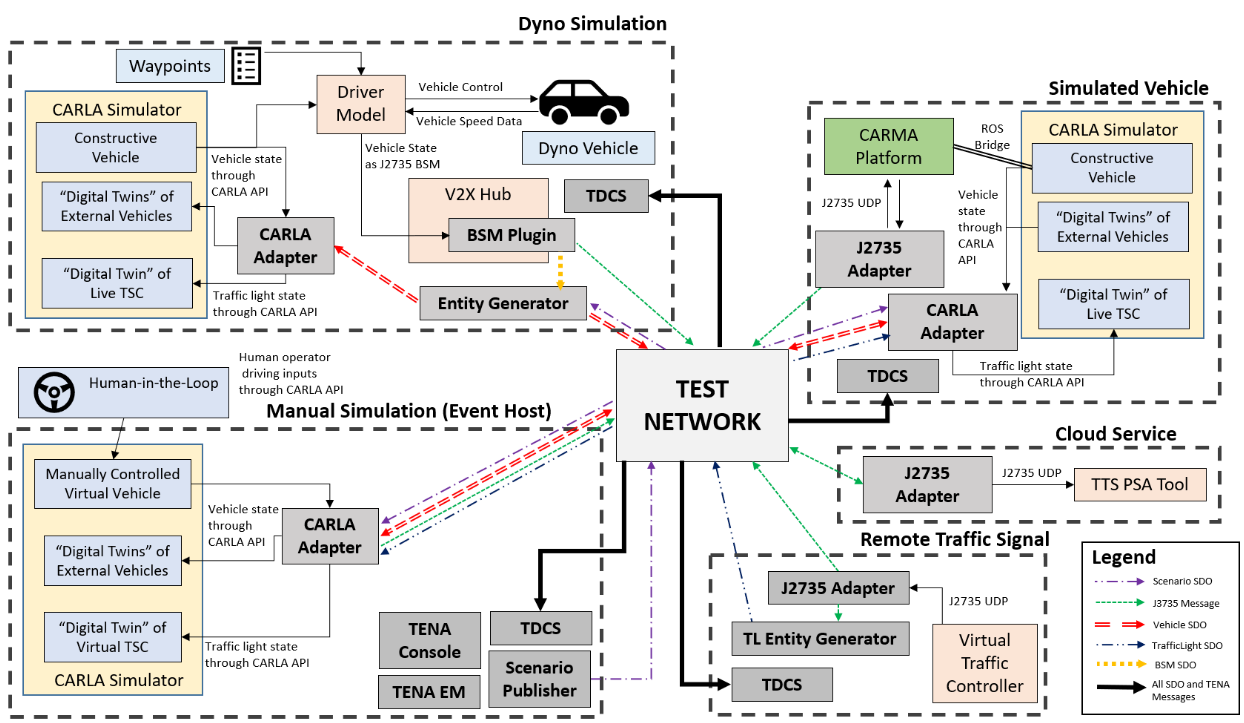
[1]This video includes the views and opinions of outside entities. The views and opinions expressed in this video do not necessarily reflect those of FHWA or the U.S. Department of Transportation (USDOT).

CHAPTER 2. PILOT 2 TEST ARCHITECTURE AND METHODOLOGY

The Pilot 2 test campaign consisted of three events: Event 0, Event 1, and Event 2. Participants contributed various components from across the United States, and one participant in the United Kingdom controlled a US-based computer remotely. These participants were connected via a virtual private network (VPN) and performed testing by utilizing the distributed testing platform and their own distinct models. This chapter provides details on each participating site and their participating assets, how these sites were connected, and the tools that were used to facilitate distributed testing.

SYSTEM ARCHITECTURE

The exact system architecture varied for each test event in Pilot 2, but the core components were generally similar in nature. All tests took place on the same digital map and used Test and Training Enabling Architecture (TENA) adapters and object models.[12] Figure 1 features an example architecture diagram.



Source: FHWA.
BSM = basic safety message; ROS = robot operating system; TL = traffic light; API= application programming interface; V2X = vehicle-to-everything; UDP = user datagram protocol; EM = Execution Manager.

Figure 1. Diagram. Pilot 2 example system architecture.

Distributed Testing Using TENA Middleware

- TENA is a suite of tools and infrastructure created by the Department of Defense's Test Resource Management Center with the goal of designing common architecture and software to integrate testing, training, simulation, and high-performance computing technologies distributed across many facilities. TENA is built into the core of most distributed testing applications to easily manage publishers and subscribers and distribute event data. The TENA architecture contains multiple tools that were used in Pilot 2 event execution and data analysis. Following are some key TENA applications and terms to assist in understanding how TENA was used in the distributed testing system: (12)

- TENA Middleware: The high-performance, realtime, low-latency communication infrastructure used by applications and tools during the execution of a test event. Provides the core publisher and subscriber functionality for TENA applications and adapters.(12)

- Execution Manager (EM): Manages a test execution by coordinating the joining and resignation of TENA applications.(12) The EM coordinates with arriving and exiting applications to share the publishers and subscribers for each data type.

- TENA Console: A graphical user interface for monitoring and managing a test. The TENA Console connects to an EM to display connected applications and their status, monitor communications between applications, and perform basic network health monitoring.(12)

- TENA Object Model (OM): Customized classes of objects. For Pilot 2, this object model explicitly defined the format of vehicle, traffic signal, SAE® J2735™ message, SAE J3224™ message, and scenario data. (12,13,14)

- TENA Stateful Distributed Object (SDO): A specific instance of a TENA OM with persistence (for example, a vehicle or traffic signal).(12)

- TENA message: A specific instance of a TENA OM with no persistence. Pilot 2 included the generation and distribution of TENA J2735 messages and TENA J3224 messages.(12)

- TENA Adapters: Interfaces built with the TENA Middleware, which connects to the EM and interacts with TENA OMs. TENA Adapters primarily convert information from a non-TENA application into a TENA OM format and access the functions and methods of that OM. TENA Adapters often also work in reverse: receiving OM data, converting it, and sending it back to an application in its native non-TENA format. Adapters are primarily used to create, update, and destroy SDOs and to send or receive TENA messages. Pilot 2 included adapters for CARLA, J2735 messages, and J3224 messages.(7,12)

- TENA Data Collection System (TDCS): Can subscribe to one or more TENA OMs and collects all data into a database. This application was used at each site for Pilot 2 to collect data for network performance analysis, but data collection can also be conducted by only one site for later dissemination to all other sites.(12)

- TENA Playback: Can selectively replay collected TENA SDO updates and TENA messages from a database file on a per TENA OM basis. Pilot 2 used this application to replay collected data in an effort to capture network throughput for each TENA OM per site.[12]

NETWORK ARCHITECTURE

Researchers conducted the Pilot 2 test events with two different network solutions: a cloud-based secure VPN with zero-trust connections provided by Twingate™ and the VOICES portal.[2,15] The cloud-based network provided basic network functionality for Events 0 and 1 while the VOICES portal was in development.[2,15] By Spring 2024, when Event 2 was scheduled, the VOICES portal was ready for testing; thus, the test collaborators used the portal.[2]

Commercial-Off-the-Shelf Network

Events 0 and 1 used Twingate's cloud-based network and TENA Middleware to connect multiple geographically isolated research and development sites to each other.[12,15] The Twingate network architecture is composed of the following main components:[15]

- Controller—A cloud-hosted, team-managed, central coordination service, which stores configurations for user management and authorization.

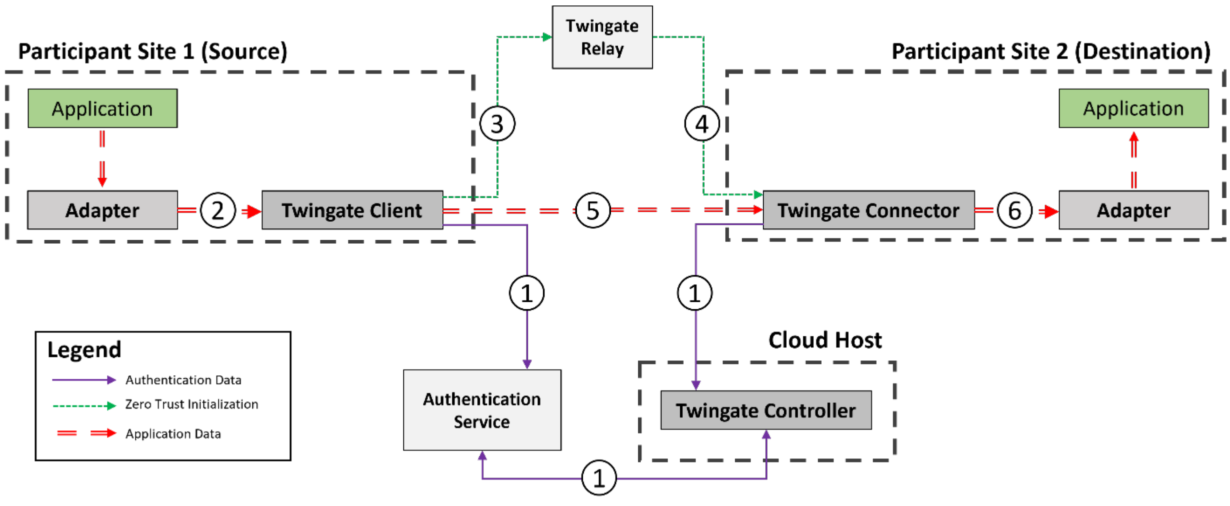
- Client—A software application installed on user devices that allows users to provide authentication and receive authorization and direct connectivity to network resources.

- Connector—An application, installed wherever resources must be assessed, that serves as a gateway for Clients to connect to network resources.

- Relay—A cloud-hosted, Twingate-managed registration point for Clients and Connectors to establish zero-trust connections.

The Twingate product includes cloud-based configuration and authentication functionalities, which are provided to the network. To ensure security, each simulation site must be authenticated to connect to the network. This authentication is carried out at each site through an instance of the Client software installed at each site, the controller in the cloud, and the cloud-based authentication service.[15]

Twingate's Controller contains all the configurations for the network. These configurations establish network resources that Clients can connect to and dictate what Clients can access what resources. Figure 2 shows an example network architecture, with a one-way communication path for one adapter to communicate with another adapter. For Pilot 2's Events 0 and 1, the research team configured all sites with Twingate Clients and Connectors, so all sites could send and receive data between one another.[15]



Source: FHWA.

Figure 2. Diagram. Example cloud-based network architecture.

Before communication can be established, the Client software at the source site (in this case, participant site 1) and the Connector at the destination site (in this case, participant site 2) must authenticate with the network using an approved authentication service, with the Connector authenticating through the Controller (figure 2). Once authenticated, an application at the source site will attempt to communicate with an application at the destination site. If this application is authorized to access that resource, the Client will make a request to the Relay for a direct connection to the appropriate Connector. The Relay will redirect the Client to the appropriate Connector via a zero-trust secure tunnel. At this point, the Client and Connector have a secure direct connection to each other and may begin exchanging data. Once the data reaches the Connector, it will be forwarded to the appropriate resource—in this case, the destination site adapter.[15]

VOICES Portal

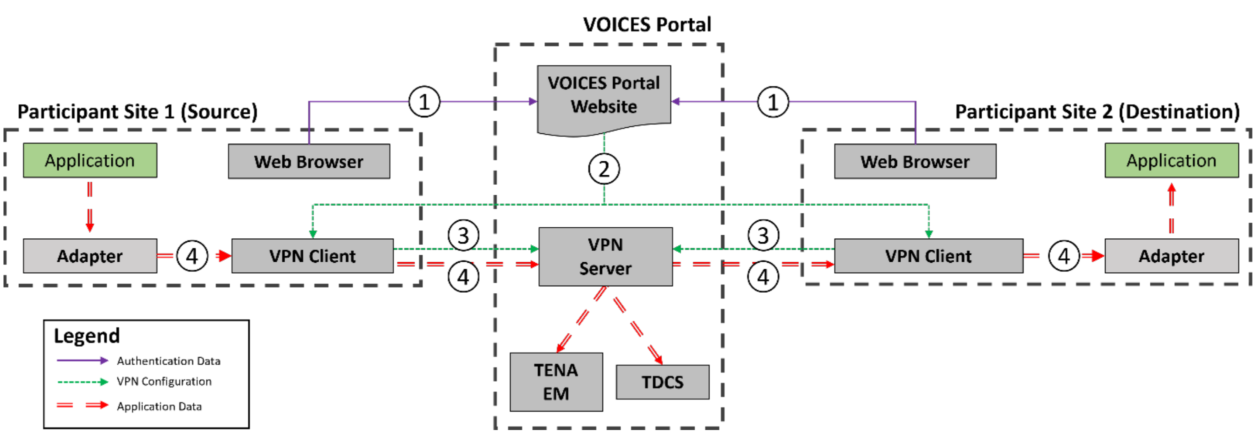
Event 2 used the VOICES portal, which provides a platform for teams to coordinate and connect to distributed testing events.[(2)] The platform allows users to create test spaces to coordinate and collaborate and test events to conduct testing. When a test event is started, an isolated instance of network and TENA infrastructure is dynamically generated on a cloud-based platform.[(12)]

The VOICES portal network is composed of the following main components:[(2)]

- VPN Server—A unique instance of this cloud-hosted service is generated for each event, which allows access to network resources by generating VPN tunnels through itself. The VPN server also hosts all configuration for user management and authorization, which is also uniquely generated for each event.
- VPN Client—An application, installed on user's devices, which allows users to provide authentication and receive authorization and connectivity to network resources via the VPN Server.

Users can download the VPN connection information from the VOICES portal and connect using the VPN Client.[(2)] Once connected to the network, users can connect TENA applications to the cloud-hosted EM with the provided Internet protocol (IP) address or host name.[(16)]

Figure 3 shows example network architecture for a distributed testing event using the VOICES portal: A one-way communication path exists for one adapter to communicate with another, but the VPN Clients can be used to communicate in both directions. The first step in the VOICES portal connection process is to log on to the VOICES portal, create an event, and invite the required participants. Once the event officially starts, the VOICES portal generates the required network and TENA architecture in the cloud. Participants can view the event and download the VPN configuration file. The network then loads the configuration file into the VPN client to establish a connection to the created network. Once this connection is established, the connected site can exchange data with all other sites connected to this event-specific network.[(2,12)]



Source: FHWA.

Figure 3. Diagram. Example VOICES portal network architecture.

TEST SITES

The next eight sections detail the test sites that participated in at least one event. These sites were distributed across the country and were connected using a VPN and the distributed testing platform.

FHWA

Location: McLean, VA.

Events: Event 0, Event 1, Event 2.

Participating assets: CARMA Platform automobile equipped with a cooperative automated driving system model leveraging CDASim.[6,17]

Summary: The FHWA site was hosted at the Turner Fairbank Highway Research Center in McLean, Virginia and was the designated test director for all events. This designation meant that FHWA hosted the TENA Scenario Publisher[12] and the EM when not hosted by the VOICES portal (figure 1).[2,6] For Event 0, FHWA manually controlled a simulated CARLA vehicle and drove the track with all other participants.[7] For Event 1, FHWA directed the test but did not bring any test assets to the event. For Event 2, FHWA participated with CARMA Platform and autonomously drove the Mcity loop as the fourth vehicle in the string.[6]

FHWA assessed the CARMA Platform simulation using parts of CDASim.[6,17] CDASim is FHWA's open-source software that integrates the four CARMA modules (CARMA Platform, CARMA Streets℠, CARMA Cloud℠, and CARMA Messenger℠) into a single cosimulation environment with central time management by an open-source cosimulation framework called Eclipse MOSAIC. (See reference 6 and references 18–21.) Central time management enables simulations to run faster or slower than realtime. However, this style of time management means that the more simulators exist, the slower the overall simulation runs the because the clock advances to the next soonest timestep of any of its simulators, no matter how small the step. This process is unlike distributed testing, which relies on realtime as its de facto central clock: The cosimulation will thus run no slower than realtime, but the cosimulation may not run faster than realtime. This constraint may be challenged in future test demonstrations. Pilot 2 used only the CARLA-CARMA Platform integration from CDASim and did not use any MOSAIC functionality.[17]

Mcity

Location: Ann Arbor, MI.

Events: Event 0, Event 1, Event 2.

Participating assets: Mcity digital twin, pedestrian detection data packaged into Sensor Data Sharing Message (SDSM), Signal Phase and Timing (SPaT) and MAP data from Mcity facility, simulated TeraSim™ vehicle model, physical teleoperated CAV, physical test track.[9,13,14]

Summary: Mcity is based at the University of Michigan in Ann Arbor, MI. Mcity provided a digital twin CARLA map of their test track facility (figure 4), which Mcity used for all three events in the Pilot 2 campaign.[7] For Event 0, Mcity manually controlled a simulated CARLA vehicle and drove the track with all other participants.[7] For Event 1, Mcity collected data from their pedestrian detection system, then repackaged and broadcasted the information as a standard SAE J3224 SDSM.[14] Mcity also generated J2735 MAP and SPaT data.[13] For Event 2, Mcity ran two TeraSim vehicles as background traffic while operating a live, teleoperated vehicle, which autonomously drove the physical Mcity loop as the fifth vehicle in the string.[9]

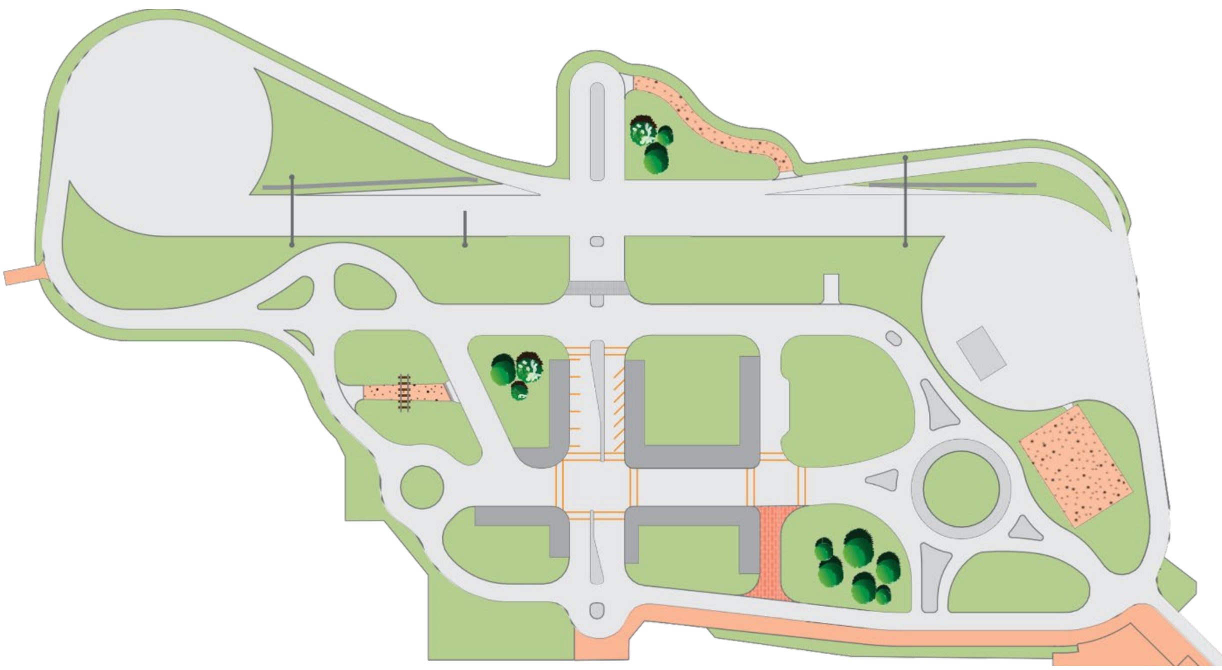


Figure 4. Diagram. Mcity test track.

ANL

Location: Lemont, IL.

Events: Event 0, Event 2.

Participating assets: simulated CAV model, live CAV on a chassis dynamometer test stand.

Summary: ANL hosted a site at their facility in Lemont, IL. For Event 0, ANL manually drove a simulated CARLA vehicle around the simulated track with all other participants. For Event 2, early integration testing, ANL used their in-house simulated basic CAV model. For the final Event 2, run-for-record, ANL operated a live electric vehicle on a chassis dynamometer test stand. Hardware-in-the-loop test instrumentation fed actual ground truth measurements of wheel speed back into the cosimulation, which updated to move a simulated CARLA vehicle.(7) In the cosimulation environment, the ANL vehicle autonomously drove the Mcity loop as the second vehicle in the string.

UCLA

Location: Los Angeles, CA.

Events: Event 0, Event 2.

Participating assets: simulated OpenCDA CAV model.(8)

Summary: UCLA hosted a site in Los Angeles, California. For Event 0, UCLA participated with an OpenCDA vehicle, which stopped at a signalized intersection, and a manually driven, simulated CARLA vehicle, which drove the track with all other participants.(7,8) For Event 2, UCLA participated with a simulated OpenCDA vehicle, which autonomously drove the Mcity loop as the lead vehicle in the string.(8)

ORNL

Location: Oak Ridge, TN.

Events: Event 0, Event 2.

Participating Assets: simulated CAV model.

Summary: ORNL hosted a site at their facility in Oak Ridge, TN. For Event 0, ORNL manually drove a simulated CARLA vehicle with all other participants. For Event 2, ORNL ran its own simulated CAV model autonomously on the Mcity loop and served as the third vehicle in the string. Researchers created the ORNL CAV model from scratch for Event 2, leveraging the CARLA application programming interface (API) and providing example control code. ORNL integrated an eco-approach algorithm, which ORNL developed in-house, into their CAV model to test this model against other vehicles in Event 2.

Econolite

Location: Toledo, OH.

Events: Event 0, Event 2.

Participating assets: virtual TSC.

Summary: These events were in Toledo, Ohio and consisted of a virtual TFC used in events 0 and 2. Econolite produced and distributed J2735 SPaT for use in CDA applications and also converted the SPaT into TrafficLight SDOs to update the appropriate signal heads in each participant's CARLA simulation.[7,13] For Event 0, Econolite also manually drove a simulated CARLA vehicle with all other participants in the collaborative test event.[7]

TTS

Location: Beaverton, OR (via Edinburgh, UK).

Events: Event 0, Event 1.

Participating assets: Personal Signal Assistant (PSA) tool.[22]

Summary: The computer system used by TTS was located in Beaverton, OR but was controlled remotely by a team member in Edinburgh, UK. For Event 0, TTS manually drove a simulated CARLA vehicle along with all other participants.[7] For Event 1, TTS participated with their PSA tool, which received J2735 SPaT and then displayed this information on a graphical map overlay and predicted future phase information.[13,22]

VW

Location: Belmont, CA.

Events: Event 0, Event 1.

Participating assets: Analysis of SAE J3224 sensor data-sharing message and European Telecommunications Standards Institute (ETSI) cooperative perception message standards.[14,23]

Summary: VW hosted a site in Belmont, CA. For Event 0, VW manually controlled a simulated CARLA vehicle and drove the track with all other participants.[7] For Event 1, VW received J2735 SDSM, SPaT, and MAP data, which can be used to compare different vulnerable road user standards.[13]

DISTRIBUTED TESTING SOFTWARE DEVELOPMENT

This section contains details about the new software developed for Pilot 2 to support the use cases for all three Pilot 2 test events. Chapter 5 discusses smaller enhancement, additions, and lessons learned during the integration and testing efforts of Pilot 2.

Defining Distributed Testing Layers

Based on an evolving understanding of how best to conduct distributed testing, testing best practices are subject to change. The most recent focus has been to emphasize that systems under testing be defined with separate physical and digital layers. The physical layer is defined as a description of the ground truth states of a system—for instance, the true kinematic states of a vehicle or the true status of a TFC. This definition applies for physical and simulated assets. The digital layer comprises information communicated from a system—for instance, the reported states of a vehicle or TFC, which may include noise or latency, based on how the states are collected and reported. If a system under test is simulated, a researcher may legitimately choose to treat data from the physical and digital layers as equivalent out of convenience, but the physical and digital layers are actually distinct. This concept must be applied to how data are classified and treated in a mixed-reality test.

During the previous test campaign, Pilot 1, the distributed testing system underwent a transition in its TENA OM classification to better adhere to the distinct physical and digital layers. During Pilot 2, participants were caught between two generations of TENA OMs, with the occasional use of the discontinued BSM OM. This OM went against current best practices because the OM took a J2735 BSM, a message from the digital layer, and decoded the contents for use in mapping and other localization (in the physical layer). These data should remain within the J2735 BSM message as reported vehicle states, to be used by other participants however they choose.(13)

To replace the BSM SDO, researchers created the TENA J2735 message during pilot 1 to simply pass on the reported data in native format. Future tests will even further simplify this exchange with the TENA vehicle-to-everything (V2X) message, combining all hex-encoded V2X messages, including the existing the TENA J2725 and TENA J3224 messages. Thus, if consulting this report to guide formulation of additional distributed testing, researchers should keep the general V2X message in mind when designing test architecture and dataflows.(13)

J3224 TENA Message and J3224 Adapter

One of the primary objectives of Pilot 2, Event 1 was to distribute SAE J3224™ SDSMs across the distributed testing network.(14) To achieve this goal, researchers had to create a new J3224 TENA Message and J3224 Adapter, which were nearly identical to the J2735 TENA Message and Adapter.(5) The J3224 TENA Message contained two fields: a message type string and an encoded hex payload. The J3224 Adapter was able to receive user datagram protocol (UDP) J3224 messages and convert them into J3224 TENA Messages and vice versa.(14,24) The J3224 TENA Message and Adapter were successfully used in Event 1 to send J3224 SDSMs from Mcity to all other participating sites.(14)

After developing the J3224 Adapter, researchers determined that both J2735 and J3224 adapters serve the same purpose and ideally could be combined. Originally, these adapters were separated so that each message could be identified and subscribed to. However, as message types were added to the capabilities of the system, an increasingly large number of adapters and messages would be required. To simplify this situation, researchers decided that a combined V2X adapter or V2X TENA message should be created to include all encoded V2X messages. Researchers created the V2X TENA Message and Adapter during Pilot 2 but did not incorporate these items into the system architecture. Rather, researchers will incorporate the new message and adapter for use in future distributed testing use cases involving V2X messaging.

Docker Deployment

One of the most difficult tasks in getting started with the early distributed testing system was personal computer environment setup. To run all the required software, computers had to be set up with the correct operating system, TENA Middleware, specific TENA OMs, additional TENA tools, and a long list of software packages.[12] To streamline this process, the distributed testing environment for Pilot 2 has been converted into two Docker® containers: one for the CARLA simulator and another for the TENA and other software components, including Eclipse® SUMO™. (See references 7, 12, 25, and 26.) These Docker containers each contained a virtualized operating system, which included all the software and dependencies for both TENA and CARLA.[7,12,25] With these two Docker containers, all a participant had to do is download the two Docker images, set up their configuration, and run a single script to use the system.[25] Multiple system configurations were created and packaged in Docker to allow users to decide if the users would like to run with or without CARLA.[7,25]

The research team further streamlined the startup process by developing a startup script, which checks for updates, validates required configuration parameters, provides user confirmation of the running configuration, and verifies network connectivity. This script increases testing efficiency by catching common user and system errors before startup.

CARLA Adapter Enhancement

During the integration testing of Event 0, researchers discovered a gradually increasing delay in the receipt of Vehicle SDO updates, which contain the updated true position of a vehicle. The observed behavior was an increasing delay from when a vehicle moved in its own simulation to when the vehicle was observed moving in another participant's simulation. This delay started at around a second but increased to over a minute as the testing went on. Through investigation, researchers determined that the Vehicle SDO updates were backing up at the receiving CARLA Adapter due to the volume of messages received (8 vehicles at 10 updates per second). The CARLA Adapter was processing and updating each vehicle individually for each SDO update received and was receiving updates faster than it could process them.

The solution that the team selected made two major improvements. First, the CARLA Adapter has been upgraded to process multiple updates in parallel. Previously, when the CARLA Adapter received an update, this adapter would latch on to the CARLA API and apply the updates one at a time.[7] With the Pilot 2 change, if multiple updates are received at the same time, the CARLA Adapter will cache those updates and apply them at the same time. In addition, the CARLA Adapter has been enhanced to only apply the most recent vehicle data. If a second update for a vehicle is received before the first update is applied, the adapter has been updated to replace stale updates with the most recent updates. This process mitigates messages from backing up by consolidating all updates received for a single vehicle into one single update—the latest. This approach technically loses data in the form of the intermediate updates, but the team decided that, in the case of vehicle SDO updates, having the latest information is more important than the in-between steps.

The CARLA Adapter updated also includes the ability to generate basic safety messages (BSMs) using vehicle data from CARLA.[7,12] This update allows the adapter to send BSMs to the rest of the network using the TENA J2735 message, even if the test asset is not normally equipped to send BSMs.[12] This capability enables these CDA applications to communicate, even when test assets are not yet equipped to generate CDA-relevant messages.

CHAPTER 3. ANALYSIS AND ASSESSMENT RESULTS

SUMMARY OF ASSESSMENT

The collective research team successfully achieved the primary test objectives for Pilot 2, which focused on both cosimulation and CDA applications, throughout the test campaign. For the cosimulation objectives, the team successfully connected multiple test sites via a cloud-based network, demonstrated the interoperability of distinct models and simulation platforms, and identified areas where simulation standards can potentially be beneficial.

Additionally, the team developed and collected digital test assets, including scenarios, maps, adapters, and analysis tools. The team also successfully advanced application objectives by testing algorithms from varied sources to identify scenarios for deconfliction and create safer interactions, thereby increasing the technical maturity of the constituent software packages. By using standard message formats, each team member is now equipped to more easily test and interact with other systems in the future.

EVENT 0

Research Question

Can multiple sites simultaneously cosimulate and interact with microscopic traffic simulators across a cloud-based network?

Event Summary

Event 0 successfully connected all sites and microscopic traffic simulators (traffic simulators that simulate the behavior of individual vehicles, as opposed to aggregate metrics, such as traffic flow and density), as well as validated the cloud-based network's capability with minimal added latency. The latency added by the distributed testing platform was typically within 5–10 ms, often falling within the network's jitter, as shown by the latency performance analysis in chapter 4. Additionally, an issue with vehicle update backlogs in the CARLA Adapter was identified and resolved, thus improving test system efficiency. Event 0 was a two-part event, which served as a regression test for existing functionality and a starting point for the Pilot 2 test campaign.

Event 0, Part 1 replicated a scenario very similar to pilot 1 that was located on a new digital twin. Two FHWA-controlled simulated SUMO vehicles and a UCLA simulated OpenCDA vehicle approached an intersection with an Econolite-controlled simulated traffic signal.[8,26] The SUMO vehicles started opposite each other and at perpendicular approaches to the OpenCDA vehicle.[8,26] When the signal turned green for the SUMO vehicles, all vehicles were engaged.[8,26] The SUMO vehicles passed through the intersection on green, and the UCLA vehicle stopped at the red light.[8,26] When the light turned green for the UCLA vehicle, the vehicle passed through the intersection. Once the UCLA vehicle was completely through the intersection, the test was complete. Figure 5 features a diagram of the starting positions for this event. This test verified that all prior cosimulation capability was retained, even with changes to the digital twin, the use of Docker, and the new CARLA Adapter.[25]

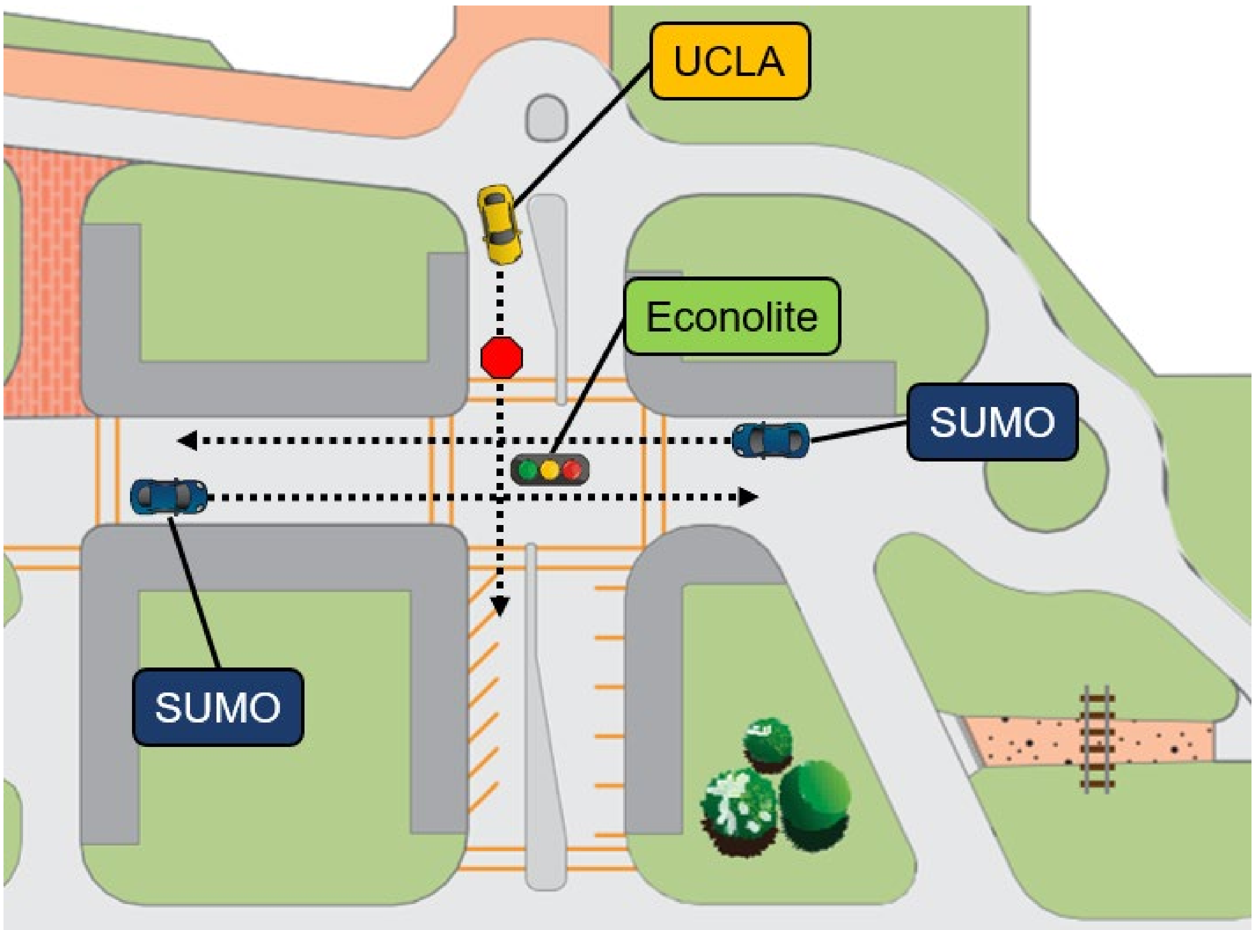


Figure 5. Diagram. Pilot 2, Event 0, Part 1 scenario.

In Event 0, Part 2, all participating sites connected with a manually controlled simulated CARLA vehicle while the Econolite traffic signal remained connected.(7) The participating manually controlled simulated vehicles for Event 0, Part 2 were Mcity, TTS, ANL, ORNL, UCLA, Econolite, FHWA, and VW. Figure 6 features a diagram containing the starting positions for each site. Researchers controlled the vehicles using a keyboard or gaming-style steering wheel and pedal box. Once all vehicles were connected, researchers drove the vehicles to a central location and then drove a full lap around an inner loop of the test track. The exact path of the vehicles was not important for this test, only that the vehicles were all connected and able to exchange data with one another.

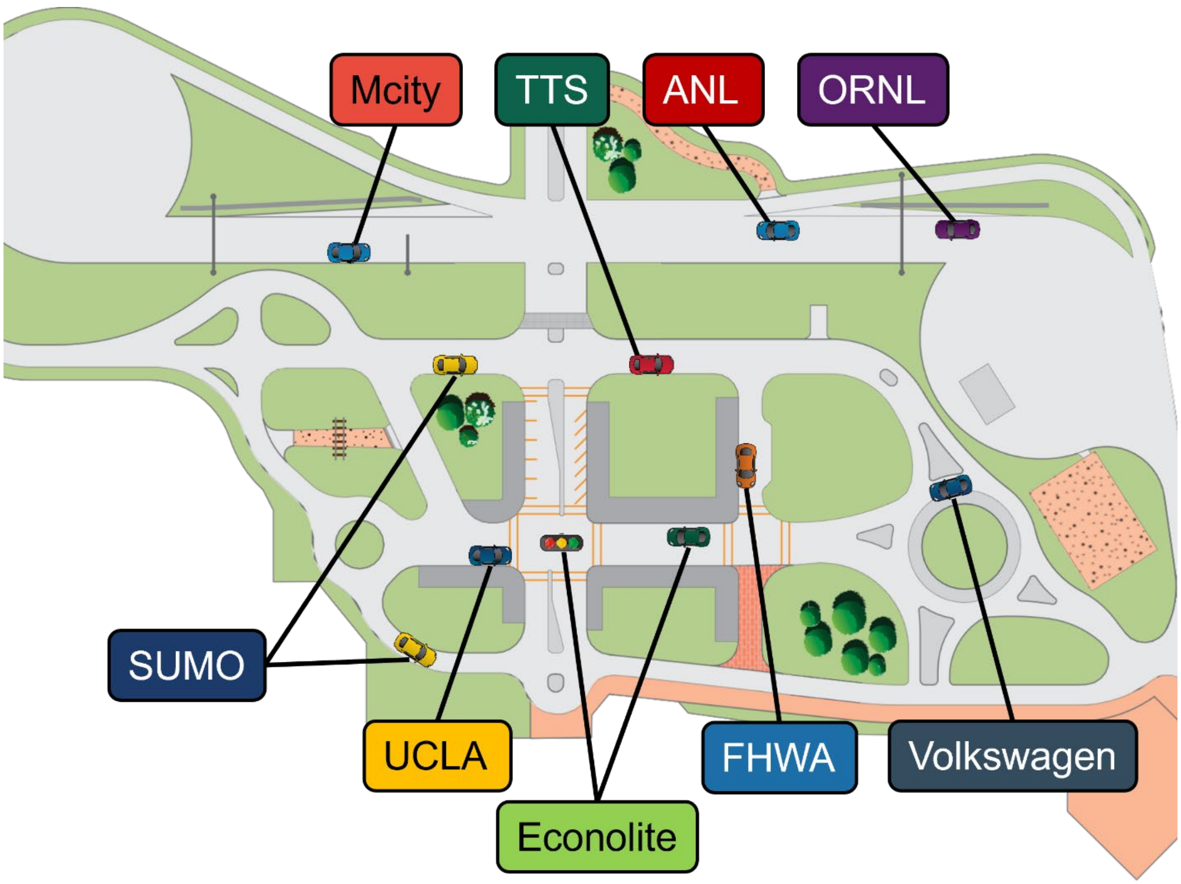


Figure 6. Diagram. Pilot 2 Event 0, Part 2 scenario.

Objective Results

Researchers completed all planned activities for Event 0, parts 1 and 2 successfully. This event also signifies the first time that multiple simulation platforms (CARLA and SUMO) successfully cosimulated using the distributed testing environment.[7,26] Given that all sites and microscopic traffic simulators were able to connect and complete their intended activities, researchers considered Event 0 successful.

In the process of achieving this result, researchers identified and implemented efficiency improvements to the CARLA Adapter. Specifically, an issue with backlogged vehicle updates was identified and resolved, which allowed the system to support the full number of participants without significant added latency, relative to the network latency (which comprises the majority of the end-to-end latency of a vehicle update; see chapter 4). More details on this enhancement are in the Platform Development section of chapter 2 in the CARLA Adapter Enhancement section.

EVENT 1

Research Question

How compatible are two different vulnerable road user-related standards?

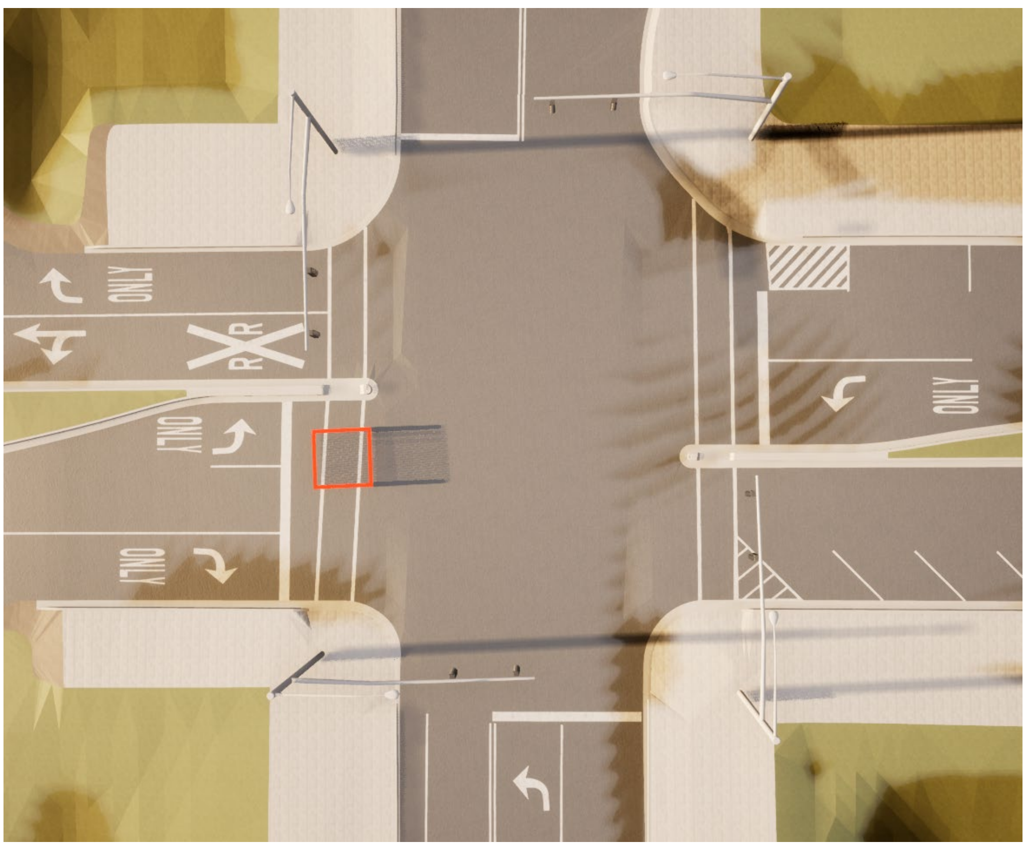
Event Summary

The primary objective for Event 1 was to exchange SAE J2735 MAP and SPaT messages as well as SAE J3224 SDSM.[13,14] Event 1 met all its planned objectives, in particular the broadcast and ingestion of SDSM and SPaT messages in realtime.[13] The development and implementation of the J3224 TENA Message and Adapter (as detailed in chapter 2, under J3224 TENA Message and J3224 Adapter) further proved the platform's capability to handle data exchange across multiple, very different systems.

During the event, Mcity replayed MAP, SPaT, and SDSM messages collected from a live intersection using real pedestrian detections.[13,14] The MAP message provides intersection and roadway lane geometry information from the digital Mcity intersection, including turning lane attributes; meanwhile, the SPaT message provides signal state and timing information.[13] The SPaT was ingested by the TTS PSA tool, which allows users to view traffic signal states and future state predictions on a local or regional level (figure 7). [13,22]

To prepare for Event 1, Mcity recorded data of a vulnerable road user crossing each crosswalk in both directions at the test intersection. These data were then packaged as SDSMs and replayed during the test event. The SDSM message provides location and classification information for vulnerable road users.[14] The SDSMs were processed by the CARLA Adapter and used to visualize the location of detected vulnerable road users on the CARLA map. [7,14] VW participated in this event as an observer and received all data for potential use and comparison to other messaging standards, such as the ETSI cooperative perception message standards.[23]

A. Screenshot. Pilot 2, Event 1 PSA tool.



B. Screenshot. Pilot 2, Event 1 SDSM visualization.

Figure 7. PSA Tool and SDSM visualization of the same test map.

Objective Results

Event 1 accomplished all planned activities, including the broadcast of SDSM for all eight of the pedestrian crossing data playbacks, the ingestion of SDSM and display of the vulnerable road user truth state data in realtime, the broadcast of SPaT messages for signal states, the ingestion of SPaT and display on the TTS PSA tool in realtime, and the data collection from VW.[13,14,22]

In addition to the receipt and display of SDSM, the TTS PSA tool was also able to display the broadcast SPaT messages in a dialogue box on the user interface (figure 7-A) in realtime.[13,14,22] This process demonstrates the capability of distributed testing to test and integrate different strategies used by traffic management centers. With the installation of a cellular modem and edge compute device to an existing TFC cabinet, traffic light information can potentially be broadcast to one or more locations in realtime for development or testing.

Finally, VW was able to receive all J2735 MAP and SPaT messages and the J3224 SDSM messages.[13,14] The company's primary focus for Event 1 was to collect SDSM messages for comparison against the ETSI Collective Perception Message, a similar European messaging standard.[14,23] Such analysis can potentially assist in harmonizing global standards and streamlining V2X industry development around the world.

EVENT 2

Research Question

Does surrounding AV traffic challenge the assumptions each vehicle makes when optimizing for ecodriving?

Event Summary

Ecodriving algorithms are sometimes developed and optimized based on individual vehicle behavior under very controlled conditions. The resulting calculated or even measured energy savings may therefore not be robust when the surrounding environment changes. Event 2 sought to understand this phenomenon in a distributed testing cosimulation environment. Event 2 focused on the energy usage of five autonomous vehicle platforms driving around the inner loop of the Mcity test track, as shown in figure 8. To assess the impact of surrounding traffic on each driving algorithm, researchers completed test runs three times with each vehicle individually, as well as five times with all five vehicles together. For all runs, vehicles started outside the loop and navigated toward the signalized intersection controlled by the Econolite virtual traffic controller. Each run was initiated so that the vehicle(s) would approach the traffic signal when in a red phase. The vehicle(s) would then observe the red light using either the SPaT generated by the Econolite traffic controller or the CARLA API states and stop at the light.[7,13]

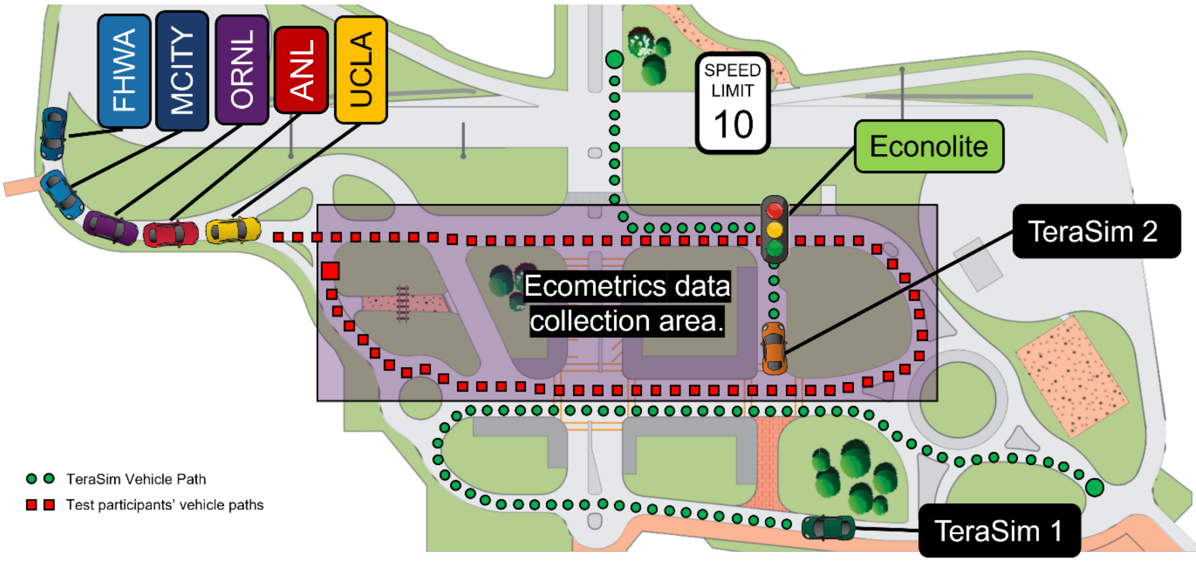


Figure 8. Diagram. Pilot 2, Event 2 scenario.

In the case of the combined runs, vehicles also used the slowing and stopping of other vehicles to stop at the intersection. The slowing and stopping of vehicles was determined from either the SAE J2735 BSMs broadcast by each vehicle, the CARLA API states, or CARLA simulated light detection and ranging.[7,13] When the light turned green, the vehicle(s) would continue around the test loop until reaching the end point (figure 8). Simultaneously, two TeraSim vehicles drove parallel paths near the test loop (figure 5).[9] These vehicles were controlled by the University of Michigan's naturalistic driving simulator, TeraSim, and were added as background traffic for the driving scenario.[9] Energy data were collected and analyzed for each participant vehicle for every run.

Each of the five participating autonomous vehicle platforms was unique and developed by a different organization. In the front, UCLA operated its OpenCDA driving platform.[8] Next, ANL operated a live vehicle on a chassis dynamometer. This system used other vehicle's BSM data to calculate and apply acceleration controls to the live vehicle and then used the resulting wheel speed to advance the vehicle along a set path.[13] Next, the ORNL team developed a driving system from scratch using the CARLA API and an in-house-developed, eco-approach algorithm.[7] Next came the Mcity vehicle, which was an Autoware-based automated driving platform that controlled a live vehicle on the Mcity test track via teleoperation.[10] The central driving system gave acceleration and steering commands to the live vehicle, and the live vehicle provided the resulting speed and location. More information on each organization's driving platform is in the organization's test site section in this chapter.

Objective Results

In Event 2, while energy consumption data were collected from participating AVs, system constraints and approximations made it difficult to draw concrete conclusions. However, the vehicles successfully completed their routes, and the Econolite traffic signal broadcast SPaT data across all sites.[13] These collective results demonstrate that the testing platform not only achieved its technical goals but also facilitated valuable insights for future CDA research.

While all prescribed activities and connections were completed for Event 2, making conclusions about energy-savings performance on the collected energy data were difficult, due to the low speed at which the test was executed, variations between runs, and a short test loop. This constraint was imposed by the live vehicle at Mcity, which is teleoperated and must operate at speeds of 10 mph or less for safety reasons. Under this constraint, all participating physical and simulated AVs were able to complete the designated route without collision.

In addition, the traffic signal was able to broadcast SPaT data to all sites, BSM and SPaT messages were received by all sites and were used for path planning and collision avoidance, vehicle and traffic light updates were shared and updated on all sites, and energy data were collected on all sites for solo and group runs.[13] The success of this test event— the first time FHWA sponsored a cooperative driving event with five unique automated driving platforms— was a notable step in CDA development. Through this testing, researchers made enhancements and fixes to nearly all vehicle models (described in more detail in chapter 5 under Lessons Learned). This type of distributed testing can enable academia, industry, and government to test CDA applications easier and earlier in the development process by enabling researchers to connect digitally and integrate with existing development platforms.

Even though the energy consumption analysis yielded inconclusive results due to the low speed of the test, Event 2 was still invaluable as a demonstration of how such testing might be conducted. Energy consumption data were collected from one live vehicle and four simulated vehicles for three individual runs per site and five group runs. ANL had a vehicle installed on a chassis dynamometer and collected real world energy consumption data from sensors on the vehicle (figure 9).[27]

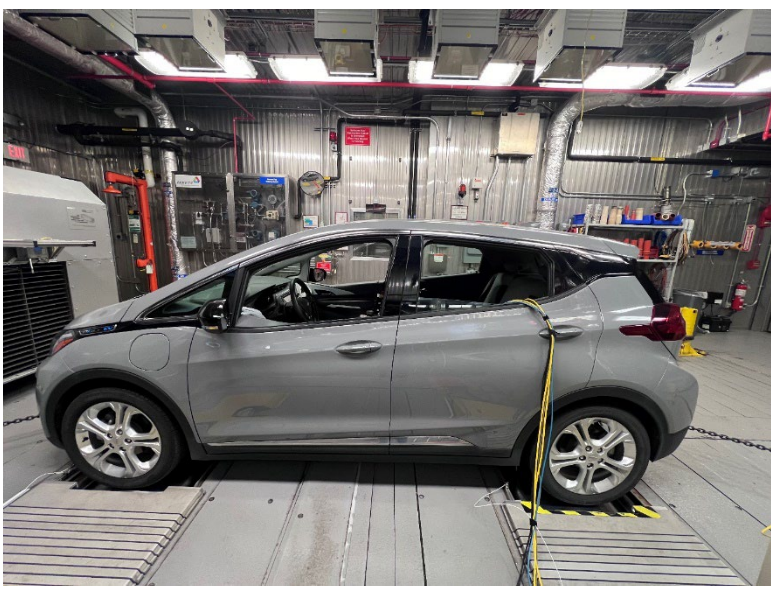
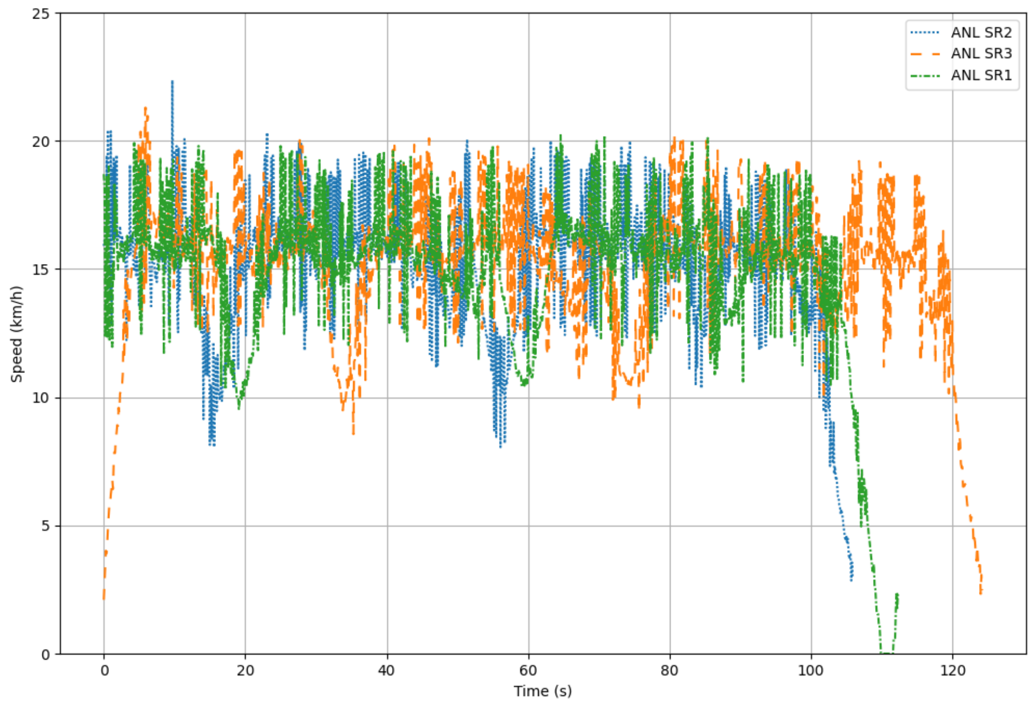


Figure 9. Photo. ANL dynamometer.

For the simulated vehicles run by FHWA, UCLA, ORNL, and Mcity, data were collected through the CARLA API, including vehicle type, location, speed, and road grade, to be fed into an energy consumption algorithm. Given the uniqueness of each of the automated platforms and how these platforms interacted with CARLA, researchers needed to make some approximations to collect comparable data, as follows:(7)

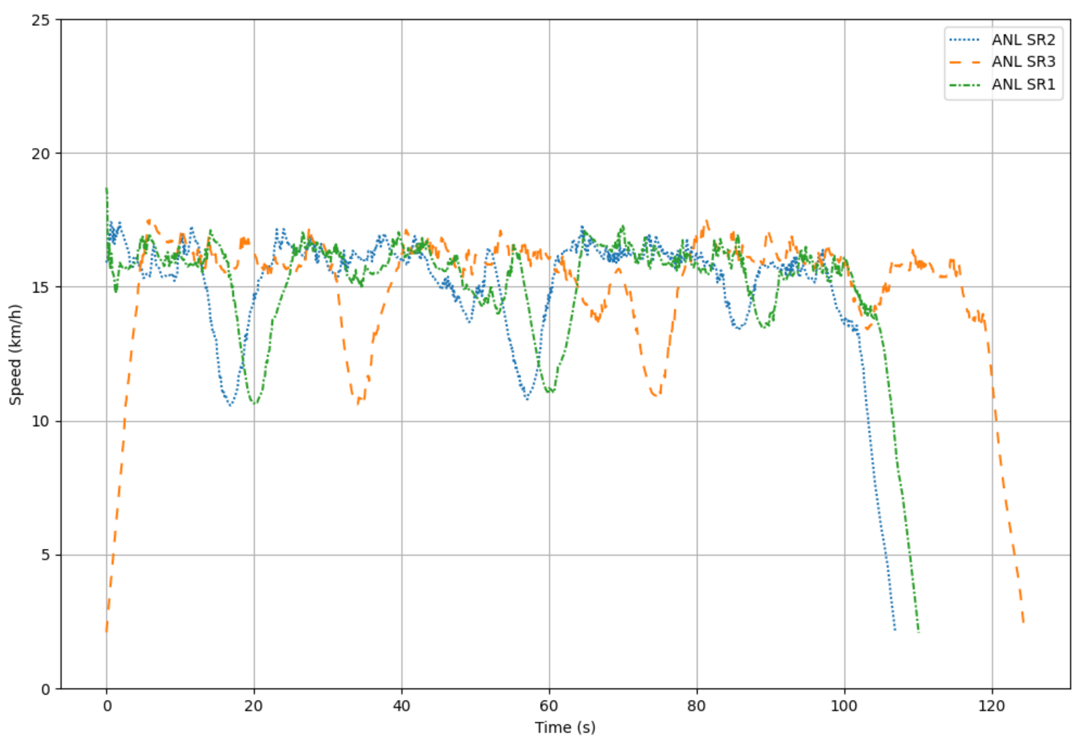
- During integration testing, some vehicles could not use CARLA's instantaneous velocity for the energy consumption analysis.(7) Additionally, the simulations for FHWA's CARMA Platform and UCLA's OpenCDA were running slightly faster than realtime, causing their reported instantaneous velocity to be faster than the observed 10 mph.(6,8) Also, the Mcity vehicle, using a SUMO-CARLA integration, did not provide an instantaneous velocity output from CARLA.(7) This lack of output meant that all vehicles except for ORNL were missing or not providing accurate speed values directly from CARLA.(7) For consistency and improved accuracy, researchers decided that the speed for all simulated vehicles should be calculated by differentiating position over time. While this method yielded more accurate speed data, the method also introduced significant noise, requiring minor smoothing to ensure usability. The smoothing calculation used was a simple exponentially weighted average with an alpha of 0.1. The resulting smoothed outputs were subsequently used for energy consumption calculations. Figure 10 and figure 11 illustrate an example of speed data before and after smoothing.

- Without a direct way to measure road grade, vehicle pitch over time is the most direct way to determine road grade for a vehicle, as the vehicle should pitch to the road grade while driving. This method worked well for the vehicles developed by FHWA and UCLA, which originated in CARLA and therefore leveraged its physics engine.(7) But since Mcity and ANL were both being updated via external sources that did not include vehicle pitch (SUMO and waypoints, respectively), their vehicles always appeared perfectly horizontal, regardless of the road grade.(26) To obtain the road grade, the nearest point on the CARLA roadway was found from every vehicle position.(7) From each of those positions, the next and previous road points were found, and the slope was calculated between these points.



Source: FHWA.

Figure 10. Graph. ANL vehicle speed without smoothing (solo runs).



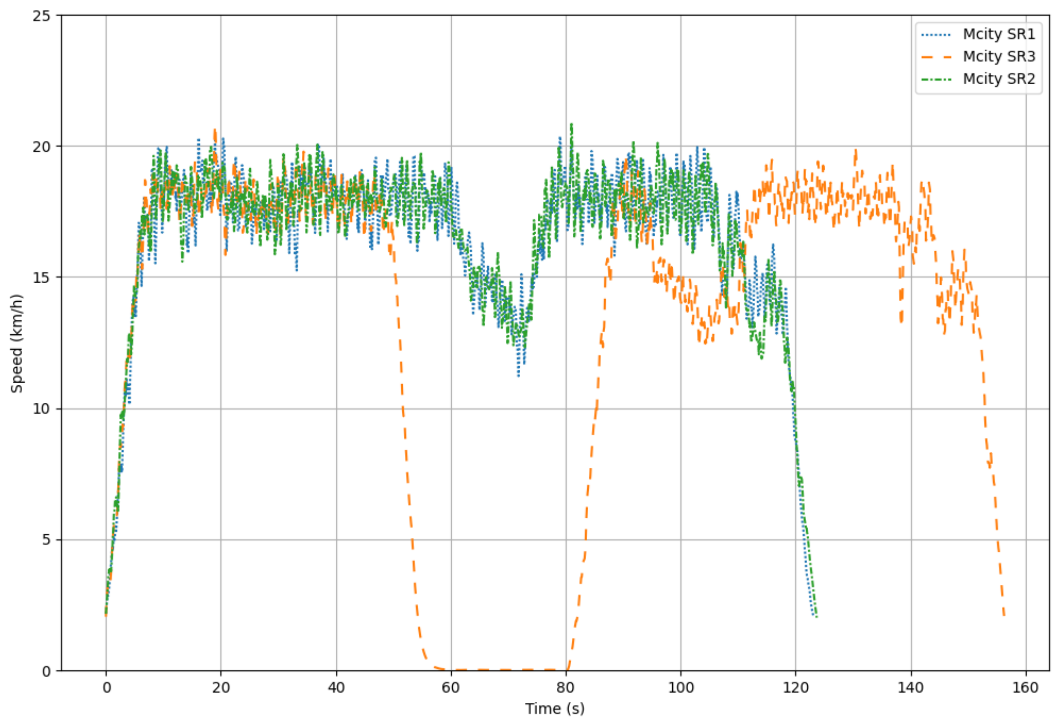
Source: FHWA.

Figure 11. Graph. ANL vehicle speed with smoothing (solo runs).

The factors that made eco-analysis for Event 2 difficult to calculate were as follows:

- Run duration was short due to the size of the Mcity track, as well as the desire to execute tests quickly. Driving the Mcity loop multiple times to collect more data was possible, but the team opted against this option due to budget and scheduling constraints for test assets, test facilities, and staff availability.
- The speed limit for the event was limited to 10 mph (the maximum safe operating speed of the live Mcity teleoperated vehicle). These low speeds meant that energy consumption was difficult to calculate for all the vehicles in the event. Higher speeds create opportunities for more or less aggressive acceleration and deceleration, producing more significant variations in energy data. With a slow speed, all vehicles quickly accelerate to the cruising speed and spend most of their time cruising. Performance differences due to the ecodriving algorithms were most likely lost in the measurement noise. For the live vehicle, low speeds meant that the energy consumption required for driving the engine was smaller, so slight energy consumption variations in other onboard electrical components, such as the air conditioning, would have a higher impact on the final energy consumption value.
- Slight variation in drive cycles between the solo and group runs was a feature of this test event, not a bug—notably. All vehicles were meant to follow approximately the same path and encounter a red light. However, because some vehicles either started at different times, relative to the signal phase or ran the red light, researchers observed some significant inconsistency of drive cycles within the solo runs (SRs). An example of this variance can be seen in a plot of the SR speeds for Mcity in figure 12. This plot shows that only SR 3 resulted in a full stop; meanwhile, traffic light timing during runs SR1 and SR2 resulted in the vehicle slowing but not stopping as the light turned green again.

Run-to-run variability may mask potential differences in energy performance of each ecodriving algorithm. Too few runs were conducted to allow for characterization of distinct distributions of performance for individual versus group runs. In the future, more runs could be conducted to sufficiently capture statistically significant distributions of performance for individual versus group runs. More runs will also allow runs with problematic deviations to be excluded while ensuring enough runs exist to produce conclusive metrics.



Source: FHWA.

Figure 12. Graph. Mcity vehicle speed (SRs).

The collected data were passed on to ANL's team for Autonomie, a vehicle system simulation tool, to process into energy consumption estimates.(28) Autonomie has aggregated vehicle dynamometer test data across dozens of powertrain configurations to estimate energy consumption, performance, and cost across different vehicle classes, powertrains, components, and control strategies. To simplify the analysis for Pilot 2, Event 2, all vehicles used the energy consumption profile of the model used on the ANL chassis dynamometer test stand.(27) The results for the energy consumption analysis for the SRs are in table 1; group runs are in table 2. Given the short duration of each test run (less than 5 min), no measurable energy consumption was recorded on the live ANL vehicle on the chassis dynamometer. Therefore, ANL energy consumption data were calculated using the CARLA API position, same as all the other vehicles.(7) The averages and standard deviations for each site for solo and group runs are in table 3 for comparison.

Table 1. Autonomie-estimated energy consumption for SRs (Wh/mi).

Site	Run 1	Run 2	Run 3
ANL	300.4	303.1	298.6
Mcity	487.1	479.6	463.8
UCLA	356.5	355.8	360.3
ORNL	318	318.2	314.1
FHWA	314.7	322.3	317.7

Table 2. Autonomie-estimated energy consumption for group runs (Wh/mi).

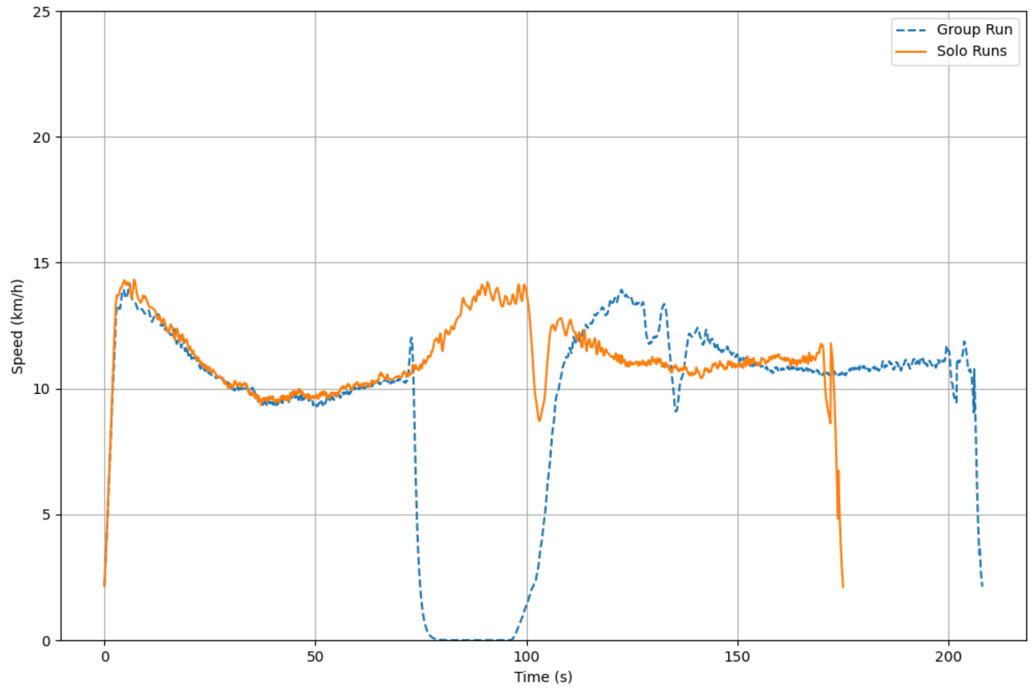
Site	Run 1	Run 2	Run 3	Run 4	Run 5
ANL	359.5	393.8	370.6	364.5	369.7
Mcity	432.8	467.7	438.8	421.8	444.7
UCLA	381.1	400	410.7	385.1	397.2
ORNL	335.3	348.4	332.4	346.1	341.7
FHWA	377.2	400.7	409.1	382.1	371

Table 3. Autonomie-estimated average energy consumption for solo and group runs.

Site	Solo Average (Wh/mi)	Solo Std. Dev (Wh/mi)	Group Average (Wh/mi)	Group Std. Dev (Wh/mi)
ANL	300.7	2.3	371.6	13.2
Mcity	476.8	11.9	441.1	17.1
UCLA	357.5	2.4	394.8	11.9
ORNL	316.7	2.3	340.7	6.8
FHWA	318.2	3.8	388.0	16.2

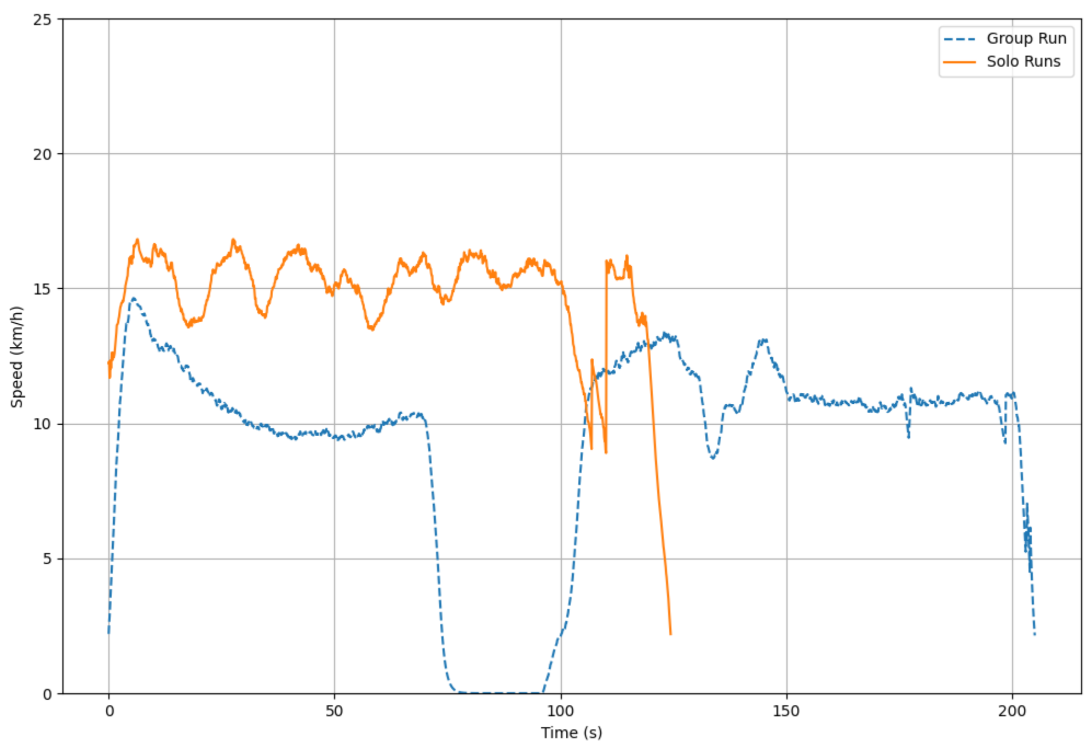
Vehicles appear to decrease in efficiency when driving as a group by an average of 11 percent. Only the Mcity vehicle increased in efficiency, by 7 percent. This finding affirms the hypothesis that five AVs that are driving together without sharing their intent will drive less efficiently than one of those vehicles driving by itself. The team assumed that surrounding traffic challenges the energy-optimized trajectories an individual vehicle may make, which appears to be supported by the data collected in Event 2. Looking at the plots of averaged solo and group run speeds over time (figure 13 through figure 17), fewer large speed changes in the SRs are seen, compared to the group runs. Additionally, the average SR speed appears to have been noticeably higher than the group speed. This finding is likely because in the SRs, no other cars were present to slow the solo vehicles down.

The Mcity vehicle also showed faster speeds in SRs but a higher energy consumption (figure 12). This finding is likely due to the fact that Mcity did not stop for two of the three SRs, only slowing down to approximately eight km/h as opposed to one km/h for group runs. The fact that Mcity was near the end of the vehicle string may have also impacted the Mcity group run efficiency. This fact, coupled with its cautious driving behavior, resulted in the Mcity vehicle rarely coming to a complete stop at the traffic light in group runs. With these many factors at play, determining what aspect had the largest impact on Mcity energy consumption is hard.



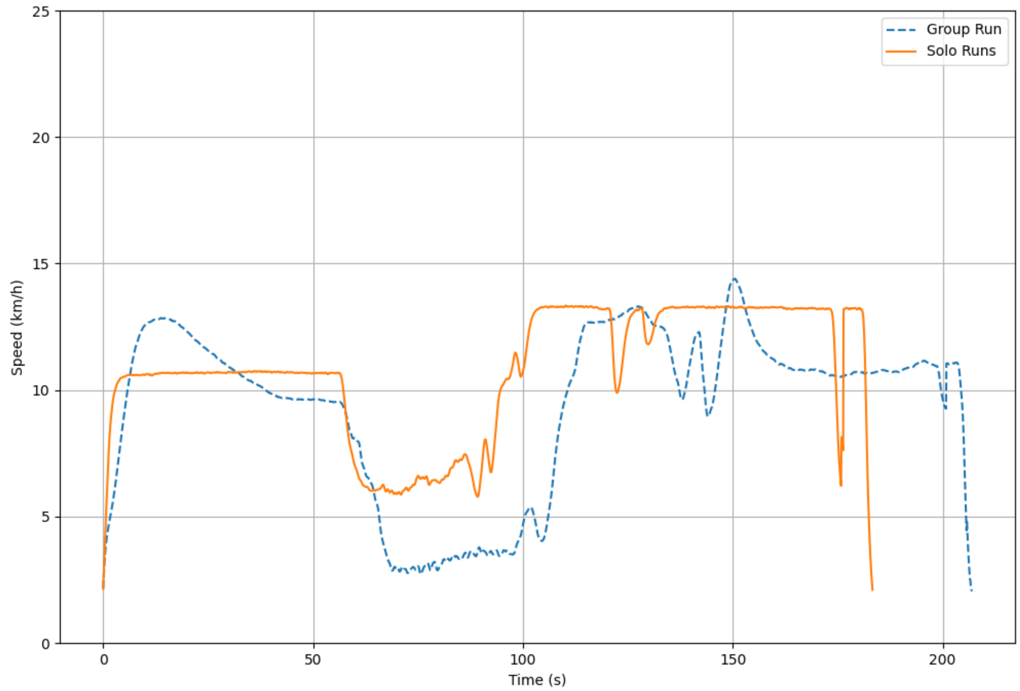
Source: FHWA.

Figure 13. Graph. Event 2 UCLA averaged vehicle speeds (group and solo).



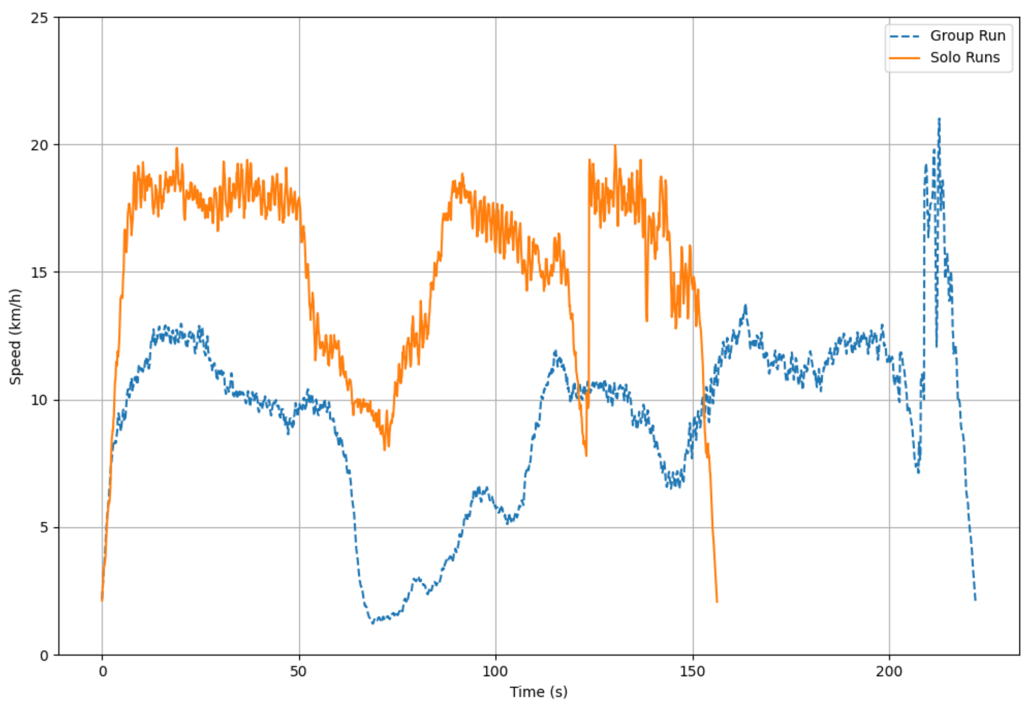
Source: FHWA.

Figure 14. Graph. Event 2 ANL averaged vehicle speeds (group and solo).



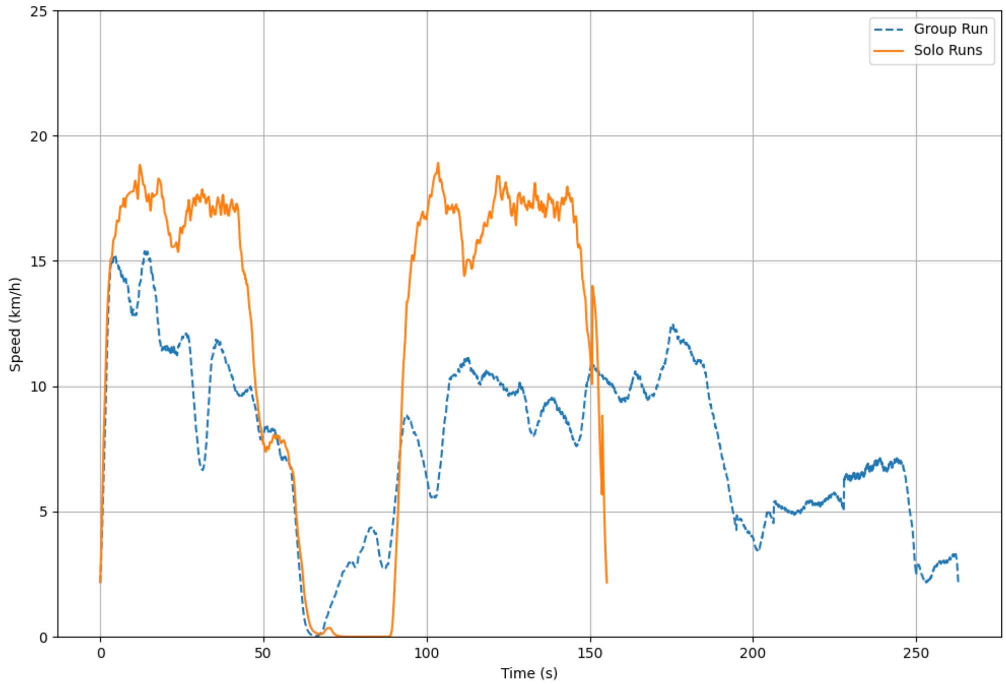
Source: FHWA.

Figure 15. Graph. Event 2 ORNL averaged vehicle speeds (group and solo).



Source: FHWA.

Figure 16. Graph. Event 2 Mcity averaged vehicle speeds (group and solo).



Source: FHWA.

Figure 17. Graph. Event 2 FHWA averaged vehicle speeds (group and solo).

Given the noted caveats and shortcomings of the collected energy data, all conclusions should be made with caveats. Regardless, Event 2 successfully collected energy consumption data that can be used to produce analysis results. The ability to compare energy consumption performance alone or in the presence of other vehicles can inform researchers about how valid surrounding traffic models are and provide a sense of how heterogeneous CAVs might perform together. With distributed testing, these studies can be made at a lower cost than live, multivehicle testing on a test track.

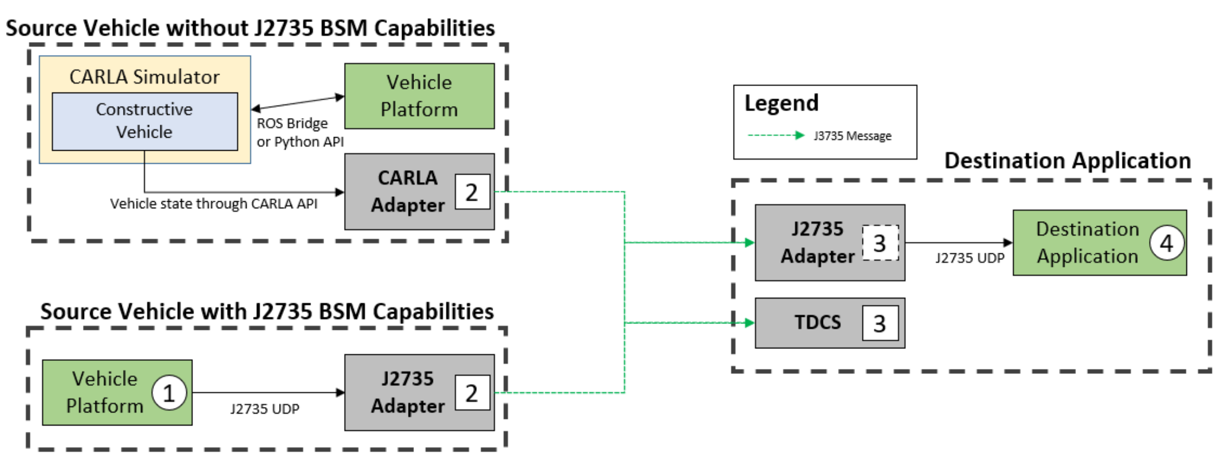
CHAPTER 4. NETWORK PERFORMANCE ANALYSIS RESULTS

This chapter details the network performance of the Pilot 2 events. The chapter documents what data were collected, how the data were collected, and how the data were analyzed; shares some example results; and draws some conclusions based on the results. While each event was unique, the data collection and analysis process stayed the same. Key differences between the events were the VPN solutions used, the sites participating, and the data exchanged.

DATA COLLECTION METHODOLOGY

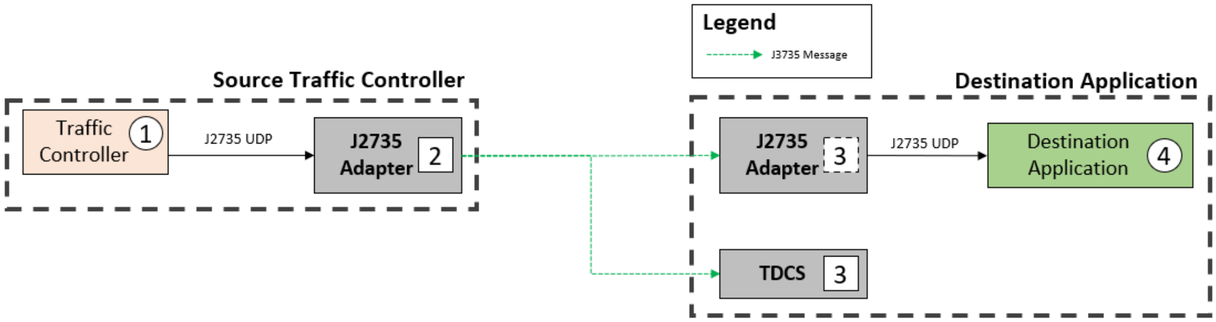
The data used for the network performance analysis were transmitted in two different formats, UDP and TENA.(12,24) Depending on the data type and the capabilities of the source site, data can either originate as a UDP message to be converted into TENA data by an adapter or be generated by a TENA adapter directly.(12,24) This section details the points where data were collected for each message flow for each type of message. Points where data were captured will be labeled with a number inside of a shape. If the shape is dashed, the data for that numbered step were collected using the TDCS.(12) This collection serves as an approximation of when the TENA data were received by the adapters on the destination side. A number inside a circle signifies that the data were UDP data collected using a packet capture, and a number inside a square signifies the data were TENA data collected using TDCS.(12,24)

Figure 18 illustrates an example of a site that generates UDP messages under the source vehicle with J2735 BSM capabilities.(13,24) First, the UDP message generated by the source application—in this case, a SAE J2735 BSM—is sent to the appropriate adapter via UDP.(13,24) The adapter receives the message, packages the encoded payload into the appropriate TENA message, and broadcasts that message to all subscribers. That TENA message will then be received by another site's adapter, which then converts the encoded payload back into a UDP message. That UDP message is then sent to the configured destination application. A similar process is seen for J2735 SPaT in figure 19 and J3224 SDSM in figure 20.(13,14,24)



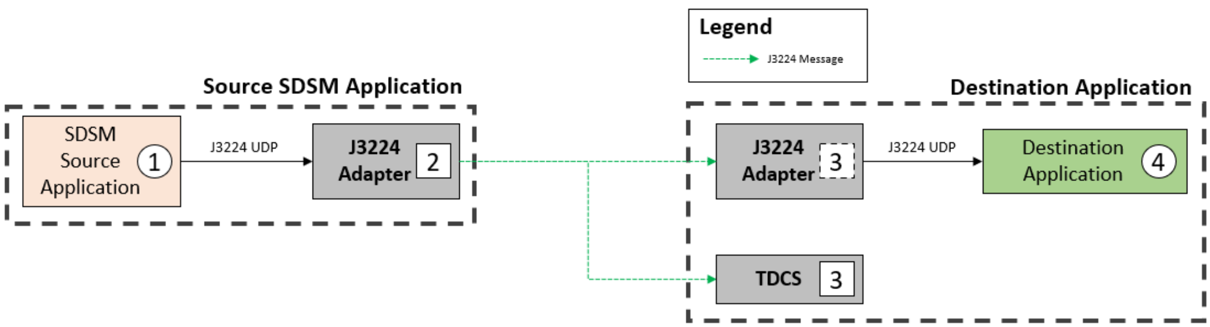
Source: FHWA.

Figure 18. Diagram. J2735 BSM message flow.



Source: FHWA.

Figure 19. Diagram. J2735 SPaT message flow.

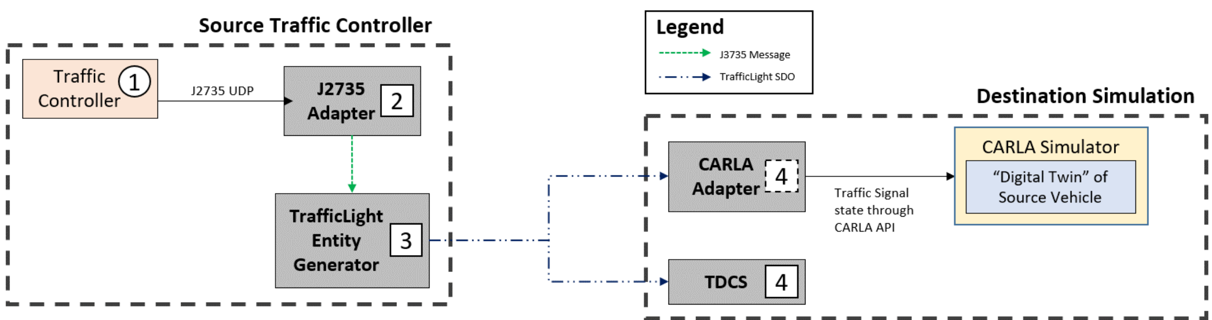


Source: FHWA.

Figure 20. Diagram. J3224 message flow.

If a vehicle platform is not equipped to generate J2735 BSM messages, the CARLA Adapter generates TENA J2735 messages for the vehicle (figure 18).[13] The process is nearly the same as for the source vehicle with J2735 BSM capabilities but skips the source UDP step, and BSMs, in the form of a TENA J2735 message, are generated directly by the CARLA adapter using vehicle information from the CARLA API.[7,13,24]

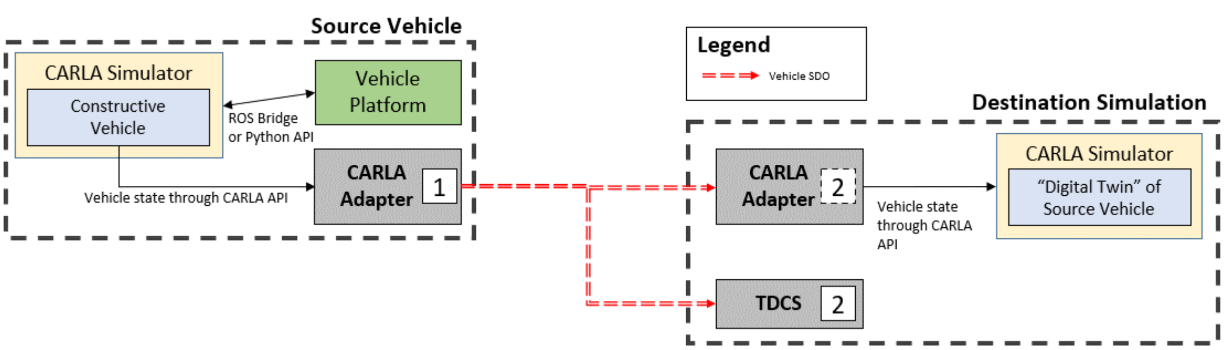
TrafficLight SDO generation requires an extra step, as compared to a standard UDP message flow.[13] Similar to the J2735 SPaT, the UDP message is generated by the traffic controller and is sent to the J2735 adapter via UDP (seen in figure 21 between points 1 and 2).[13,24] The adapter receives the message, packages the encoded payload into a TENA J2735 message, and broadcasts that message to all subscribers (figure 21). One of those subscribers, an adapter called the TrafficLight Entity Generator, receives the TENA J2735 message, decodes the payload, and packages the data into one or more TrafficLight SDO updates. That TrafficLight SDO update will then be received by another site's CARLA adapter, which then updates the CARLA traffic signal using the CARLA API.[7]



Source: FHWA.

Figure 21. Diagram. TrafficLight SDO message flow.

Vehicle SDOs are generated by the CARLA adapter by collecting vehicle information through the CARLA API.[7] The vehicle SDO update is then broadcast to its subscribers (figure 22). These vehicle SDO updates are then used to update the "digital twin" of the vehicle in destination simulations.



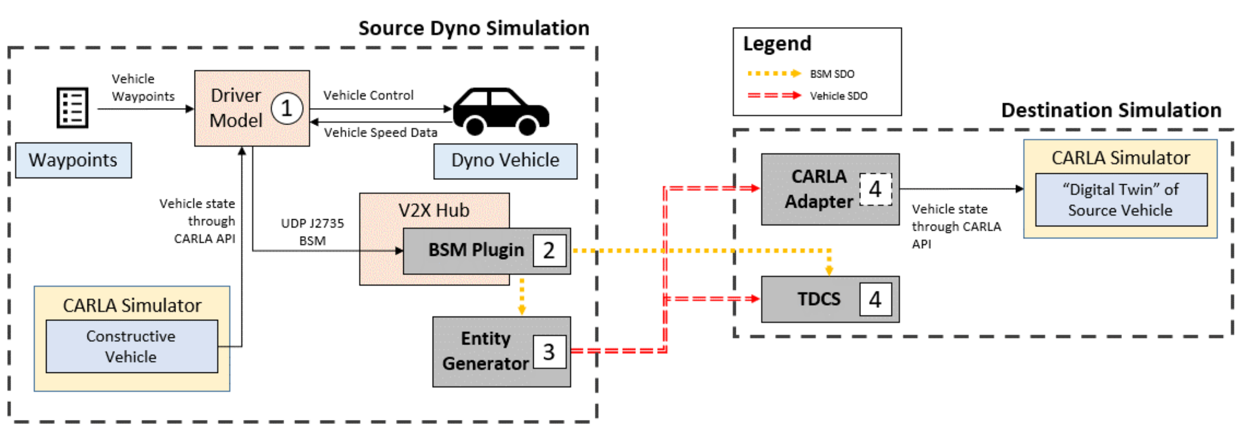
Source: FHWA.

Figure 22. Diagram. Vehicle SDO message flow.

For Event 2, ANL used a live vehicle on a chassis dynamometer and therefore could not generate vehicle SDOs from CARLA.[7] Instead, ANL updated their vehicle's position using an algorithm and architecture that packaged data into the BSM SDOs, which were converted into vehicle SDOs by the Entity Generator.[13] This process of generating vehicle SDO updates for distributed testing is not recommended by the research team, as it follows an old process, which was used for live vehicle integration for SIT-1.[4] The process is outdated due to the use of the BSM SDO rather than the J2735 TENA message.[13] After SIT-1, the distributed testing message architecture was reevaluated and simplified by replacing all message-specific J2735 TENA messages and SDOs (example: BSM, MAP, mobility) with a combined J2735 TENA message.[4,13] The new J2735 TENA message also no longer decoded the message payload and simply transmitted the hex bytes, decreasing adapter complexity and reducing processing time. The complex system that ANL developed to integrate their vehicle under testing was under development before this change was implemented and tested. Once integration testing began, researchers determined that using the old TENA messages and SDOs was more efficient than rearchitecting the entire ANL system, saving this time for additional testing. This architecture decision did not affect the ability for ANL to participate in Event 2, as the vehicle behind it in the string (ORNL) leveraged the CARLA API for vehicle perception and therefore did not need to receive TENA J2735 messages (containing BSM data) from ANL for perception.[7,13]

The ANL vehicle's speed commands were generated using a combination of dynamometer test stand data, CARLA API data, and J2735 BSMs from other vehicles.[7,13] These speed commands were then applied to the live vehicle on the dynamometer, which engaged the accelerator pedal command to spin the wheels. Measurements of this live wheel rotation data were integrated to emulate travel via a series of waypoints. As the vehicle traversed the waypoints, J2735 BSMs were generated based on the estimated truth states and sent to the BSM Plugin of the V2X Hub℠ (seen in figure 23 between points 1 and 2).[13,29] The BSM Plugin generated and broadcast BSM SDO updates, which were received by the Entity Generator (seen in figure 13 between points 2 and 3). The Entity Generator converted the BSM SDO data into Vehicle SDO updates, which were broadcast to other CARLA Adapters (seen in Figure 13 between points 3 and 4). These Vehicle SDO updates were then used to update the digital twin of the ANL vehicle in all participants' simulations.

Note that Mcity's live vehicle was essentially teleoperated open loop: The system closed the control loop using information from its simulated sensors and states, not from the physical sensors and estimated states thereof. (This use of simulated data is also why test speeds are restricted to 10 mph or less for safety reasons.) Therefore, Mcity used data flows analogous to the simulated vehicle models and not the ANL data flow, despite operating a physical test asset.



Source: FHWA.

Figure 23. Diagram. Vehicle SDO via BSM SDO message flow.

In addition to latency data, data throughput was also calculated for each site by message type. Measuring throughput is important because it shows how much bandwidth is required to complete the test. For the tests conducted in Pilot 2, the total throughput for a given site was well below 1 MB/s, but these values can be used to extrapolate the bandwidth required for larger scale test events. Throughput was captured by replaying subsets of collected data and measuring the network traffic over time in kilobytes per second (KB/s). These data could not be captured live because there is currently no way to measure incoming and outgoing data by message type. To achieve this measurement, researchers filtered subset data by message type and replayed these data multiple times using the TENA Playback tool to collect average throughput measurements for each type of data.[12]

DATA COLLECTED

Details about what types of data were exchanged for each event are in table 4 and table 5. If data were exchanged during an event, the data were intended to be collected. For different reasons, this intention was not always the outcome that happened. Explanations for where and why data were not collected for specific events are in the Data Collection Notes (DCNs) section.

Table 4. Pilot 2 UDP data collected.

Message Type	Event 0	Event 1	Event 2
J2735 BSM[13]	No	No	Yes
J2735 SPaT[13]	Yes	Yes	Yes
J3224 SDSM[14]	No	Yes	No

Table 5. Pilot 2 TENA data collected.

Message Type	Event 0	Event 1	Event 2
Scenario SDO	Yes	Yes	Yes
Vehicle SDO	Yes	No	Yes
TrafficLight SDO	Yes	Yes	Yes
J2735 Message (BSM)[13]	Yes	No	Yes
J2735 Message (SPaT)[13]	Yes	Yes	Yes
J3224 Message (SDSM)[14]	No	Yes	No
BSM SDO	No	No	Yes

DCNS

Data were collected for each site at the outbound UDP, outbound TENA, inbound TENA, and inbound UDP steps for all runs.[12,24] Some sites did not produce or receive specific data for certain runs. DCNs are numbered (example: DCN1, DCN2), so they can be referenced in tables and plots. These instances and other missing data are explained in table 6.

Table 6. DCNs.

DCN Identifier	DCN Text
DCN1	Econolite was unavailable to host the TSC for the run-for-record event, so the TSC data had to be replayed at the FHWA site. These data were replayed as TDCS data; therefore, no inbound UDP existed.(24)
DCN2	Only FHWA broadcast SPaT, so no SPaT messages were received by FHWA or sent by any other sites.(13)
DCN3	Due to a CARMA Platform bug that has since been resolved, FHWA did not generate BSMs for Runs 1, 3, and 4 and only generated BSMs for half of Run 5.(6,13)
DCN4	Only CARMA Platform generated its own UDP J2735 BSMs messages. Other sites had their J2735 BSMs Messages, generated through the CARLA Adapter as TENA J2735 Messages.(6,13,24)
DCN5	ANL did not generate TENA J2735 Messages, but instead BSM SDOs were used to generate Vehicle SDO updates. Therefore, their BSM latency calculations measured the BSM SDO latency instead of the TENA J2735 BSMs.(13)
DCN6	For an unknown reason, outbound UDP messages, including J2735 BSMs from ANL, were not collected.(12)
DCN7	For an unknown reason, some ping data collected from official test runs shows "unknown" as the IP address for some applications.(16) This lack of knowledge makes determining what sites the data are from difficult, often leading to missing ping data for some sites for some runs.
DCN8	For an unknown reason, ping data for Mcity was not recorded for Event 2, Run 5.

SAMPLE DATA

A sample set of data tables for end-to-end latency, ping versus end-to-end latency, and throughput for Event 2, Run 3 are shown in table 7, table 8, and table 3, respectively, as a representative dataset from Pilot 2. The end-to-end latency datasets for each site were concatenated to create a single dataset per site that contained data from all five Event 2 group runs. Histograms, cumulative distribution function (CDF), and cumulative histograms were created using these concatenated datasets. The results for all other data not seen in this section are in appendix A. Key findings from these plots are in the next section, Key Findings.

End-to-End latency tables show the average, minimum, maximum, and jitter for the time elapsed for data transmitted from original source to final destination (for example, table 7). In the case of a J2735 SPaT message, these data would be the time from when the UDP message is sent from the traffic controller to when the UDP message is received by the destination application (figure 19).(13,24) These metrics are valuable because the metrics show the actual latency performance of the system between every site. Each row on the table represents data originating from the site found in the left column. Each cell in that row provides latency information for data sent from that source site to the destination site shown in the intersecting column. Cells that have the same source and destination site are labeled not applicable, because sites did not send data to themselves. Cells may also be labeled for specific DCNs, which are in the SCN's self-titled sections.

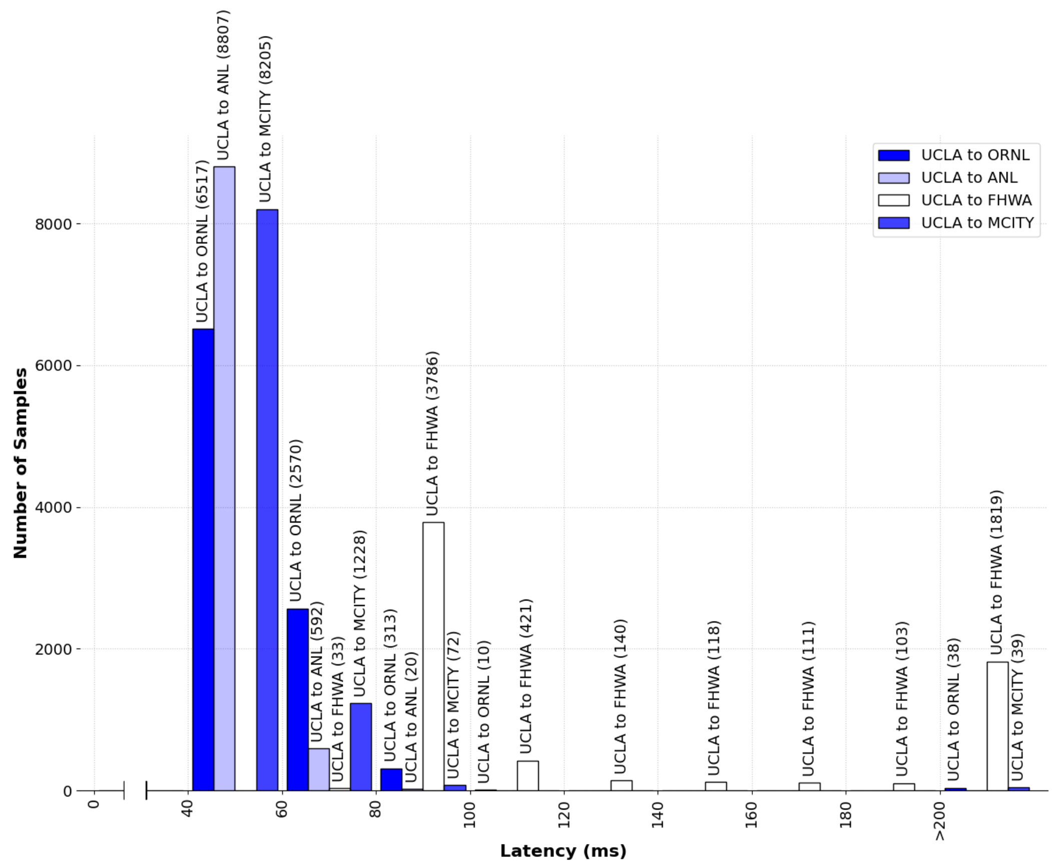
Due to the higher latencies at the FHWA test laboratory, which uses a cellular connection, researchers conducted some of the FHWA tests using a gigabit fiber connection at a staff member's home network. To differentiate, data collected by FHWA from the cellular connection has been labeled FHWA [CELL], and data collected by FHWA from the fiber connection has been labeled FHWA [FIBER].

Table 7. Event 2, Run 3, TENA J2735 message, end-to-end latency results (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Avg: 102.697 Min: 62.942 Max: 1129.831 Jitter: 129.109	Avg: 150.374 Min: 72.401 Max: 1855.508 Jitter: 252.596	Avg: 164.174 Min: 87.26 Max: 1947.381 Jitter: 253.74	Avg: 147.91 Min: 66.679 Max: 1974.41 Jitter: 269.307
SRC: ANL (IL)	Avg: 182.816 Min: 53.892 Max: 3177.025 Jitter: 41.705	N/A	Avg: 30.911 Min: 29.198 Max: 164.497 Jitter: 1.108	Avg: 46.008 Min: 44.16 Max: 179.078 Jitter: 1.656	Avg: 25.575 Min: 23.658 Max: 158.474 Jitter: 1.736
SRC: ORNL (TN)	Avg: 192.931 Min: 63.728 Max: 2586.202 Jitter: 317.697	Avg: 35.946 Min: 29.105 Max: 337.712 Jitter: 10.667	N/A	Avg: 61.522 Min: 53.48 Max: 361.599 Jitter: 12.545	Avg: 40.15 Min: 32.777 Max: 341.045 Jitter: 11.031
SRC: UCLA (CA)	Avg: 214.579 Min: 77.862 Max: 3118.556 Jitter: 342.692	Avg: 49.205 Min: 43.516 Max: 191.293 Jitter: 6.569	Avg: 60.056 Min: 52.631 Max: 115.753 Jitter: 8.339	N/A	Avg: 53.066 Min: 46.861 Max: 161.407 Jitter: 7.786
SRC: Mcity (MI)	Avg: 235.499 Min: 54.107 Max: 3492.023 Jitter: 432.59	Avg: 37.08 Min: 19.905 Max: 486.489 Jitter: 17.89	Avg: 46.714 Min: 29.242 Max: 495.357 Jitter: 18.233	Avg: 61.204 Min: 44.023 Max: 510.181 Jitter: 18.495	N/A

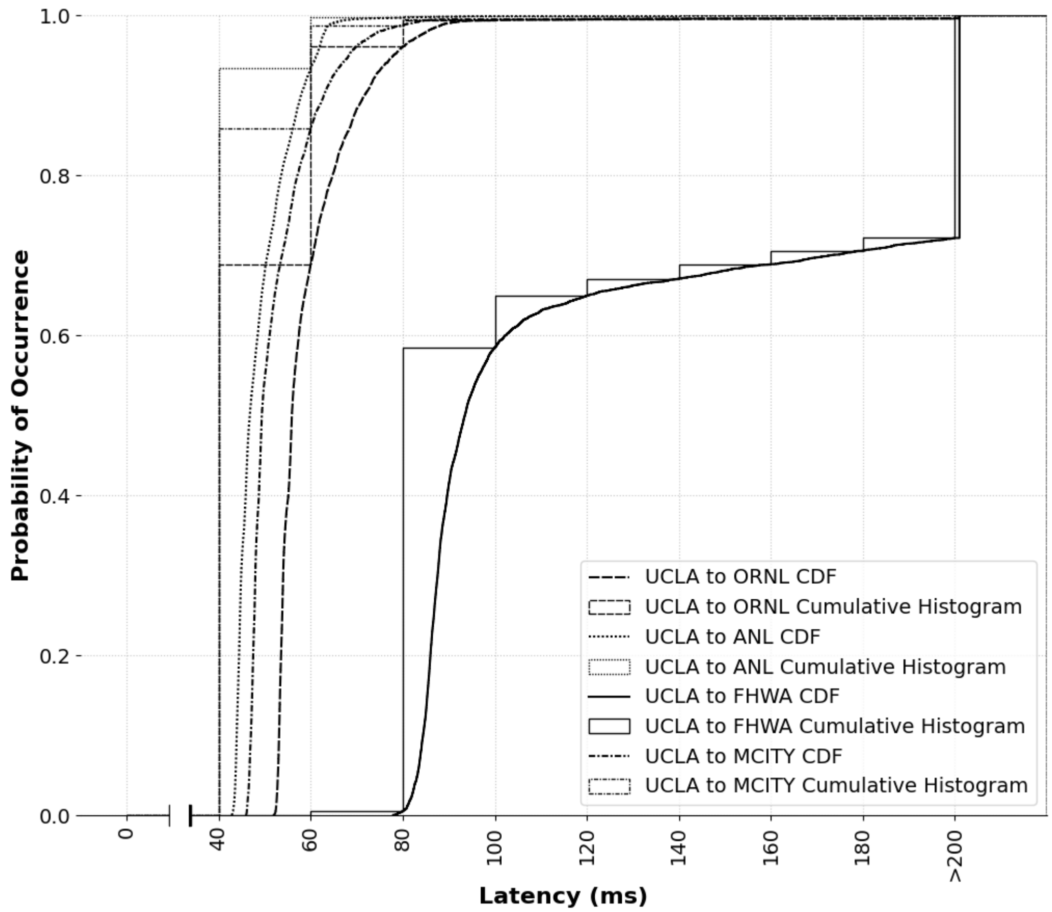
Avg. = average; DST = destination; min = minimum; max = maximum; N/A = not applicable; SCR = source.

The distribution of the latency data can be seen as histograms and CDFs in figure 24 through figure 32. Each histogram shows the latency from a single site to every other site during all the group runs. Each bar is labeled with the sending and receiving sites and the number of observed samples within that latency bin. Note that the plotted x-axes may be discontinuous (marked by gapping in the x-axis) to indicate no observations at those latency values. Also, the last bin of every histogram contains all values over the given value (marked by a > symbol). The CDF and cumulative histogram plots seen in figure 25, figure 27, figure 29, figure 31, and figure 33 show the cumulative probability that an observed latency from a dataset is below a given latency value.



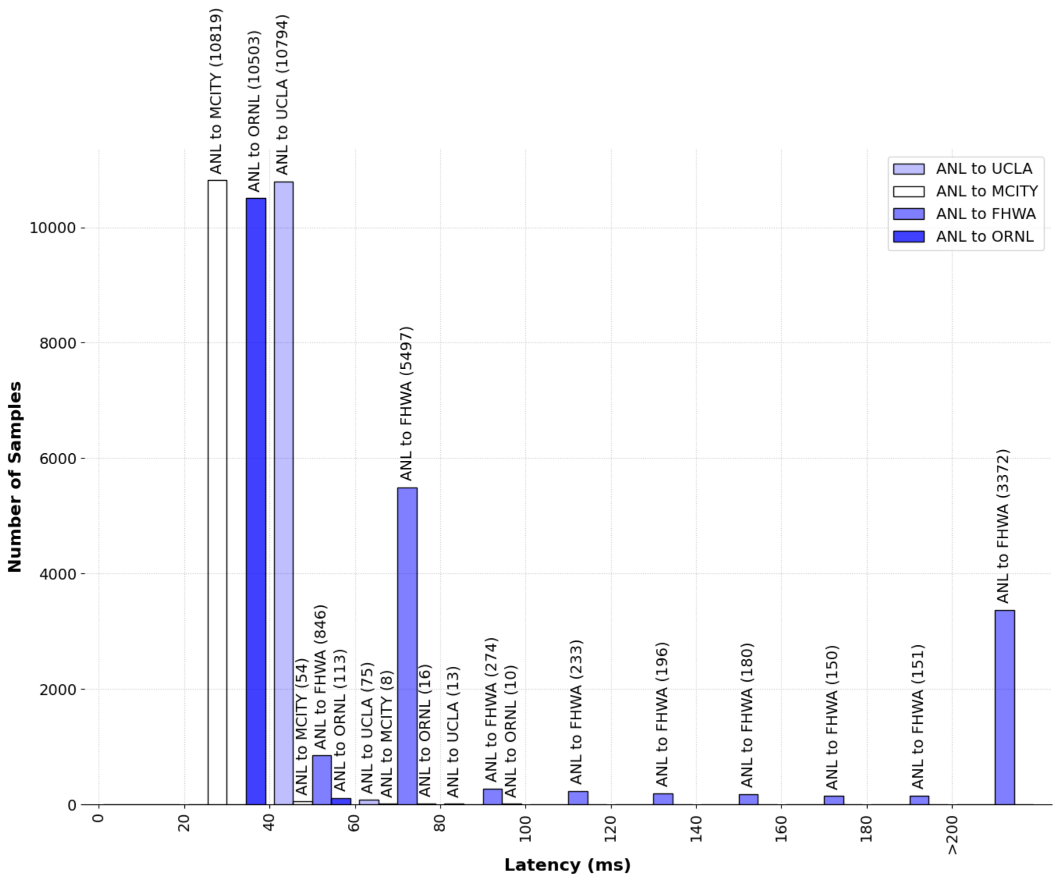
Source: FHWA.

Figure 24. Graph. Histogram of UCLA BSM latencies for all runs combined, 0 to >200 ms.



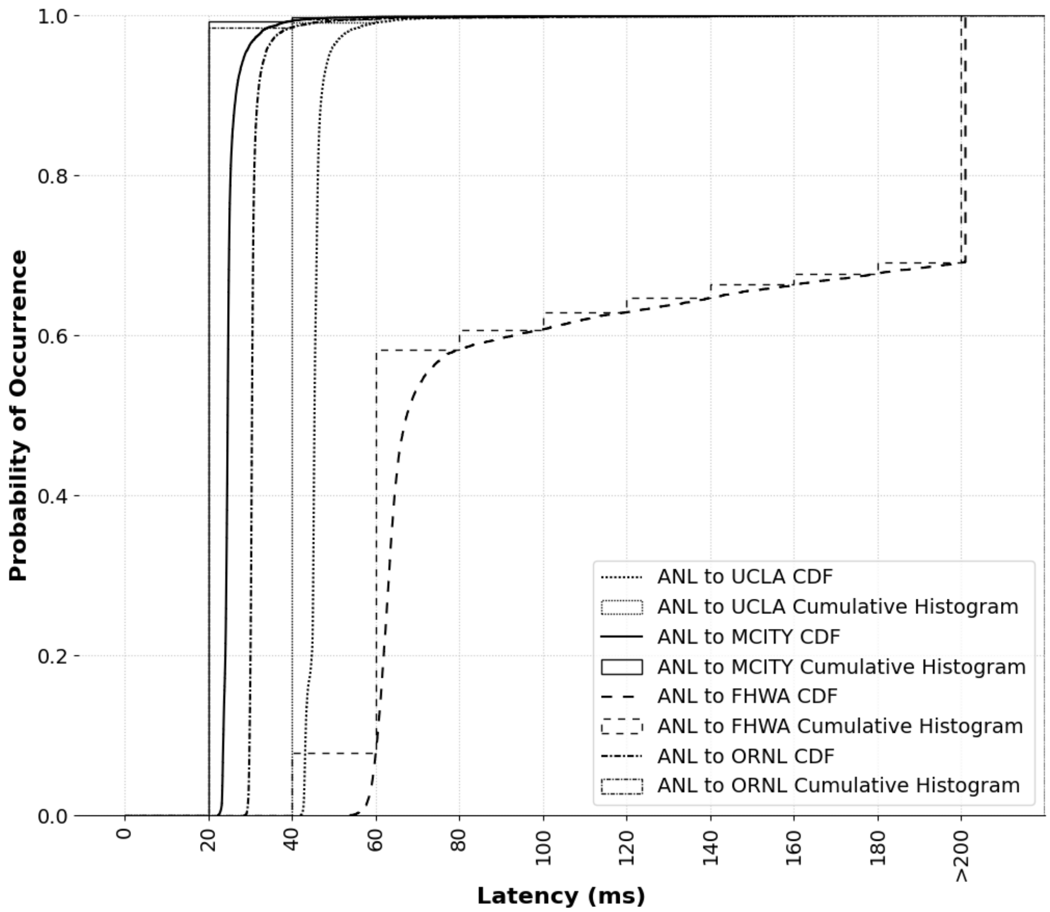
Source: FHWA.

Figure 25. Graph. CDF and cumulative histogram of UCLA BSM latencies for all runs combined, 0 to >200 ms.



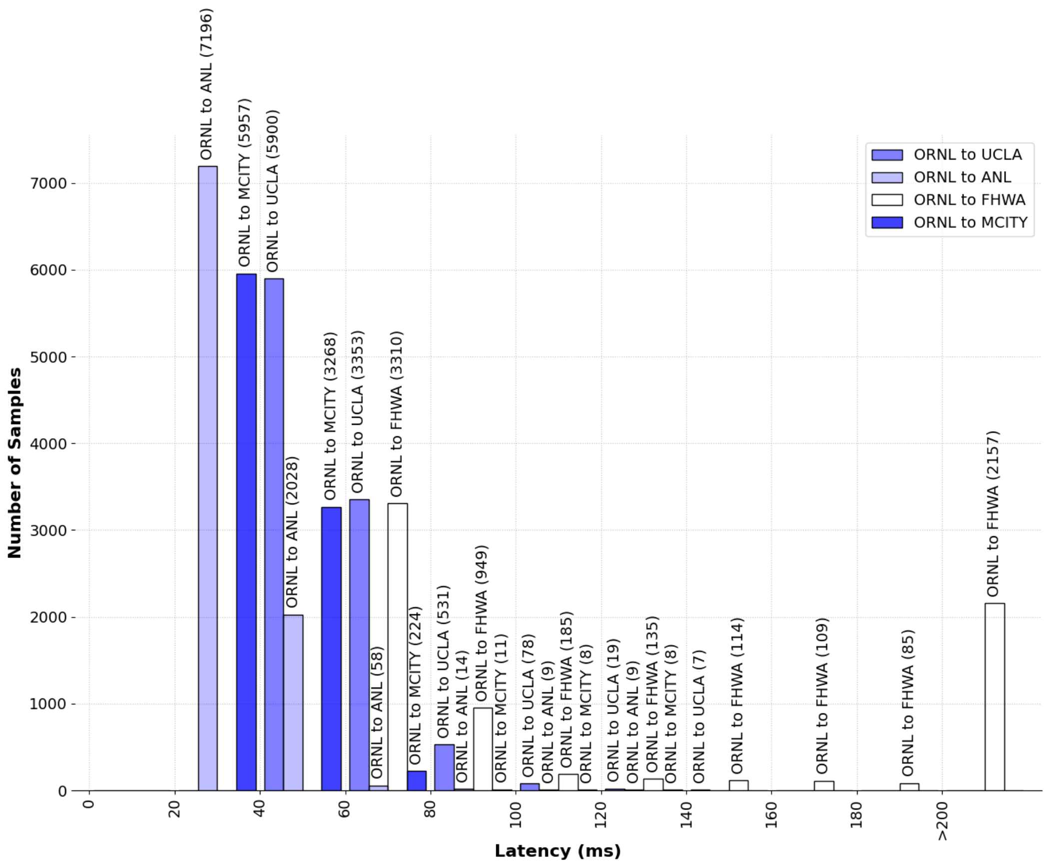
Source: FHWA.

Figure 26. Graph. Histogram of ANL BSM latencies for all runs combined, 0 to >200 ms.



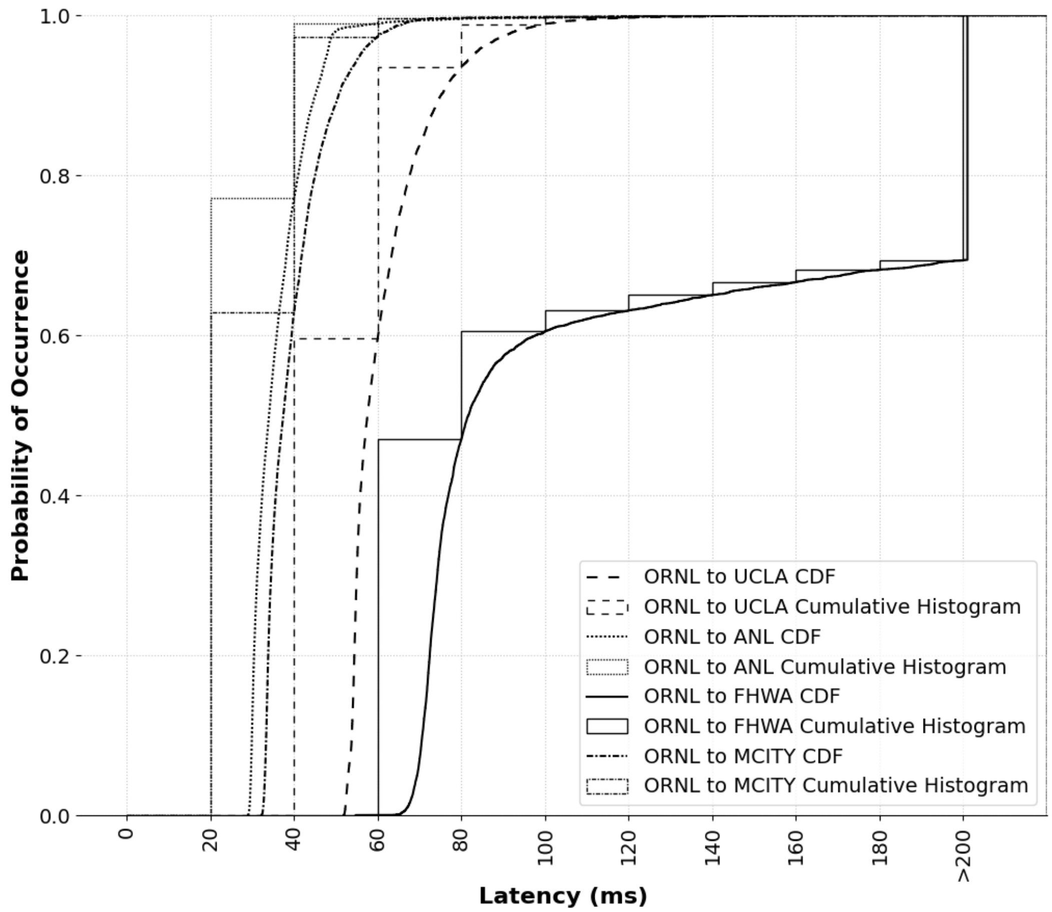
Source: FHWA.

Figure 27. Graph. CDF and cumulative histogram of ANL BSM latencies for all runs combined, 0 to >200 ms.



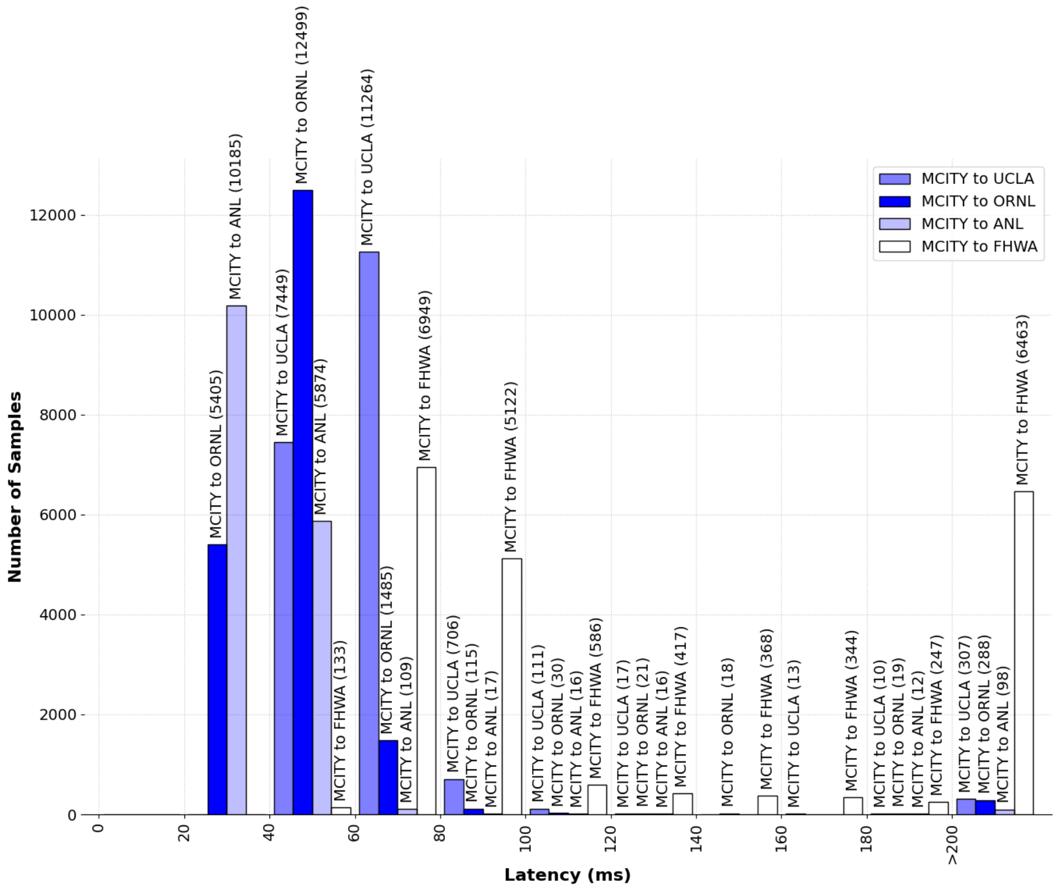
Source: FHWA.

Figure 28. Graph. Histogram of ORNL BSM latencies for all runs combined, 0 to >200 ms.



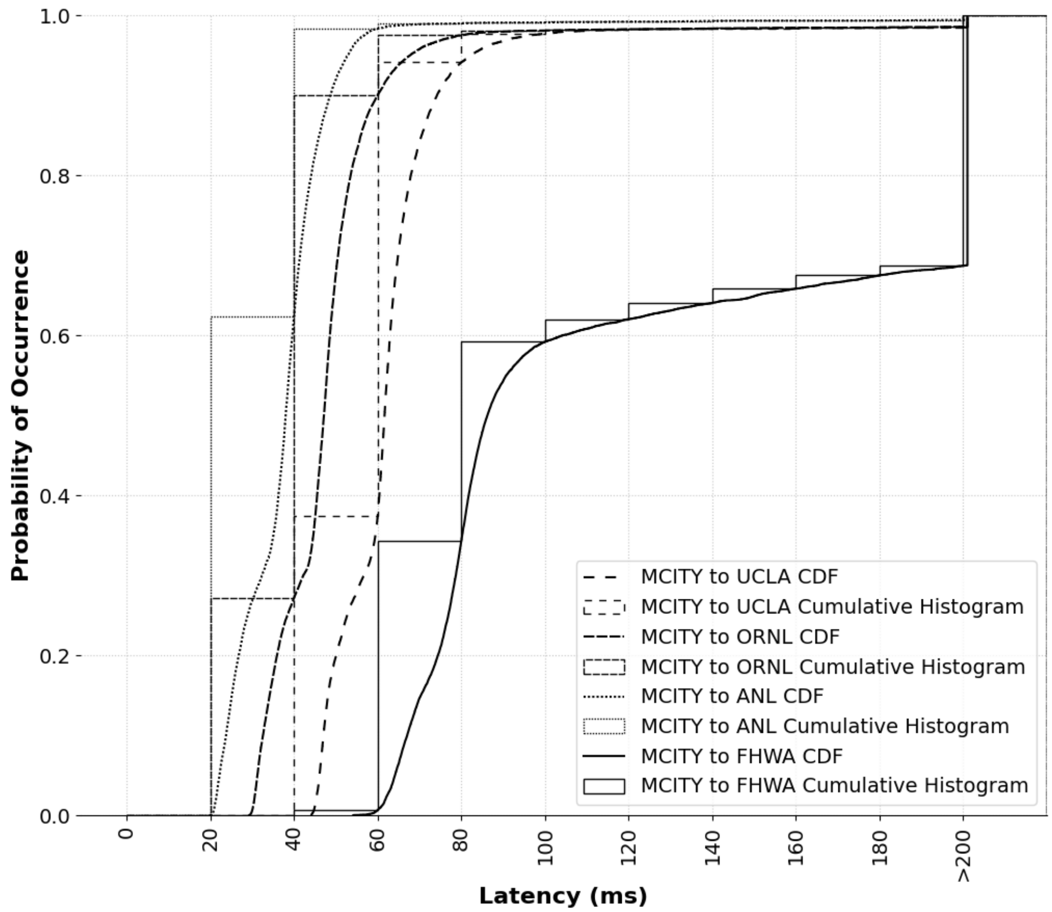
Source: FHWA.

Figure 29. Graph. CDF and cumulative histogram of ORNL BSM latencies for all runs combined, 0 to >200 ms.



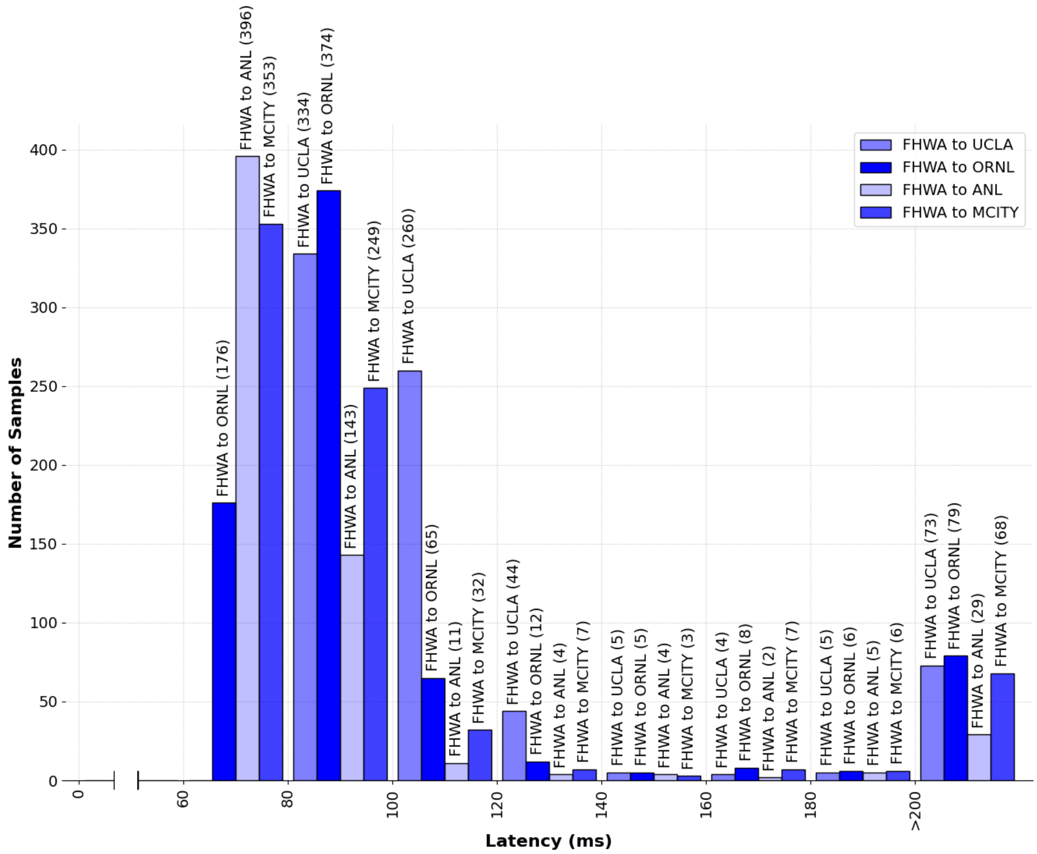
Source: FHWA.

Figure 30. Graph. Histogram of Mcity BSM latencies for all runs combined, 0 to >200 ms.



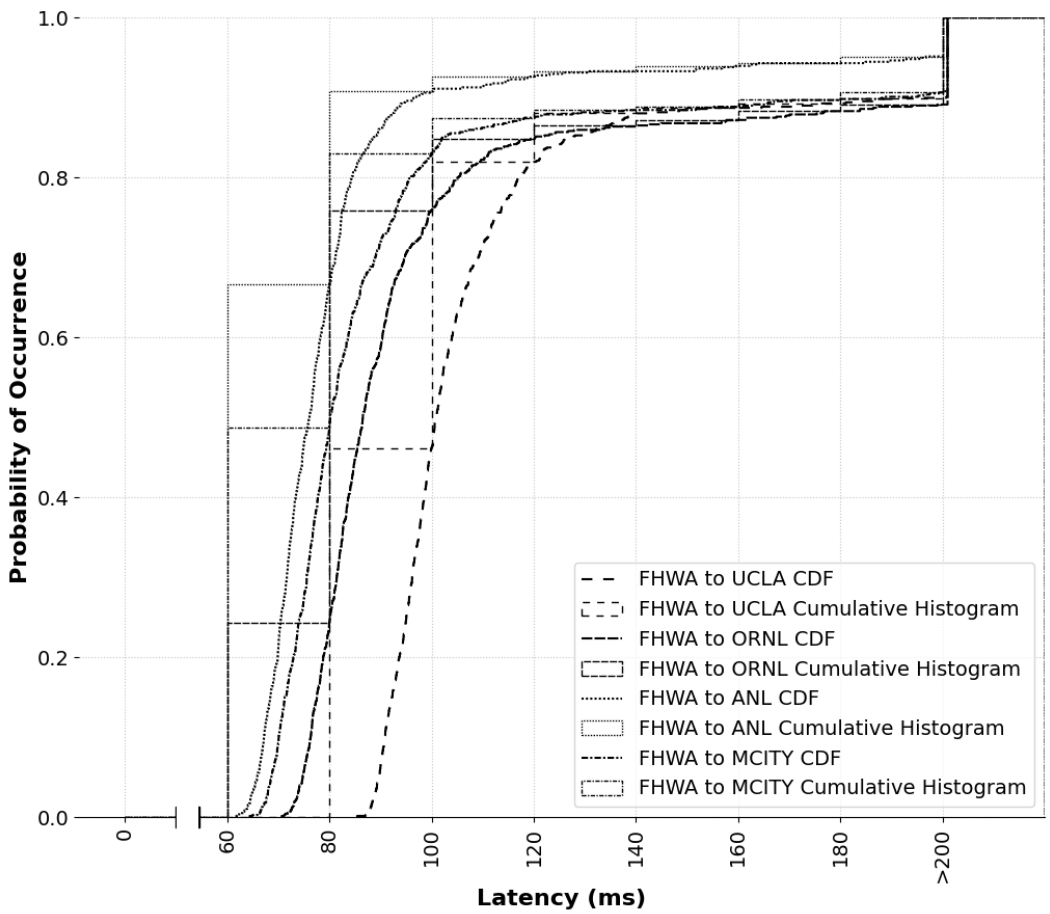
Source: FHWA.

Figure 31. Graph. CDF and cumulative histogram of Mcity BSM latencies for all runs combined, 0 to >200 ms.



Source: FHWA.

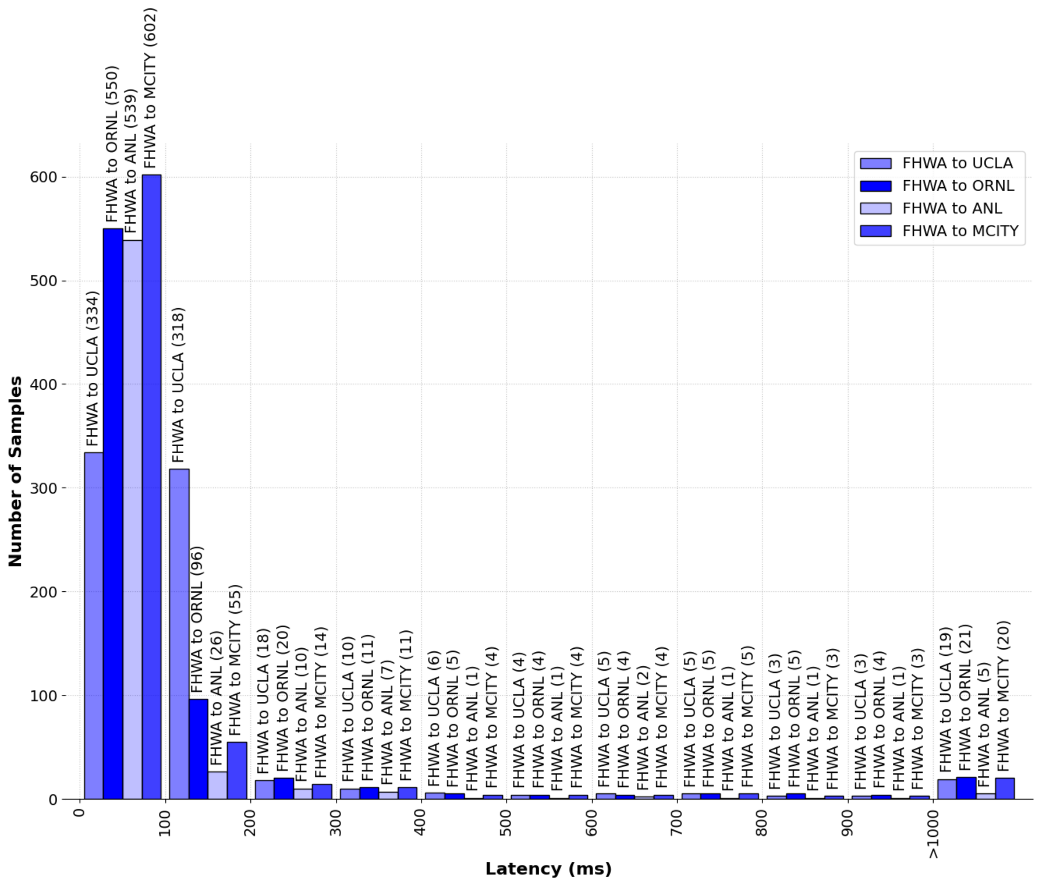
Figure 32. Graph. Histogram of FHWA BSM latencies for all runs combined, 0 to >200 ms.



Source: FHWA.

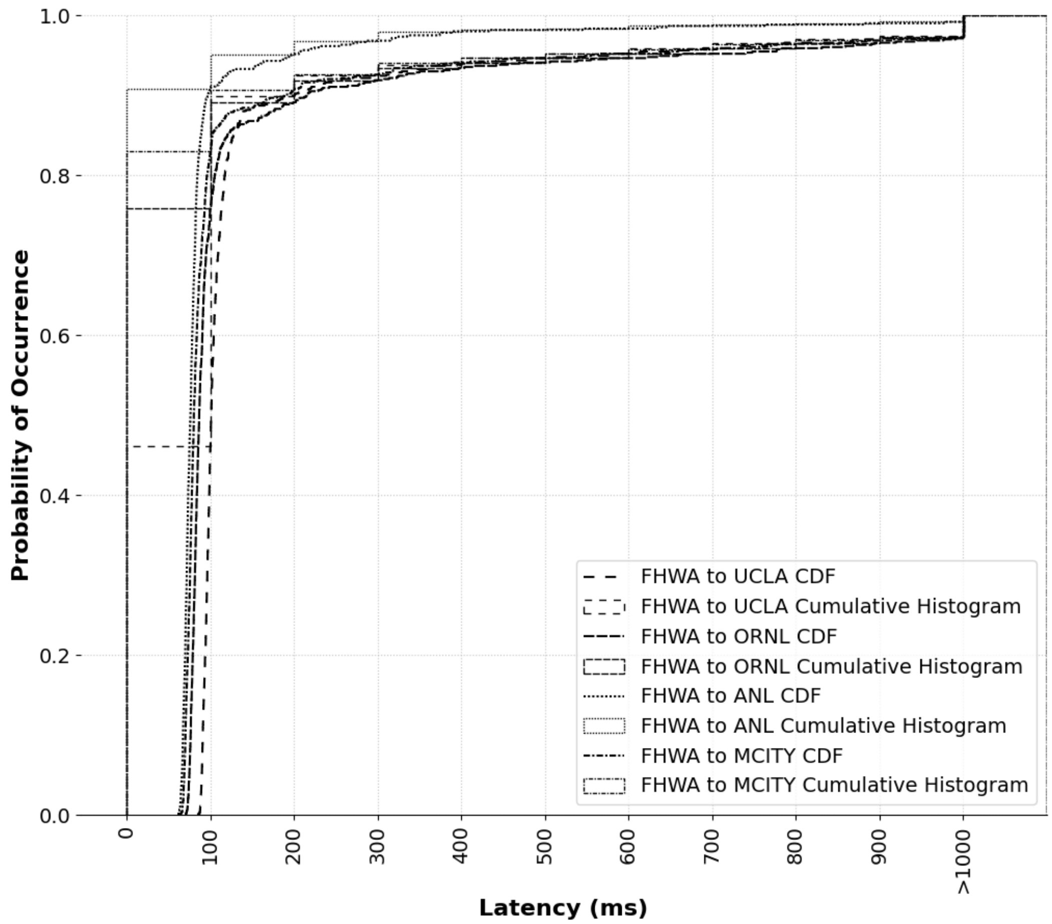
Figure 33. Graph. CDF and cumulative histogram of FHWA BSM latencies for all runs combined, 0 to >200 ms.

Observed across all the histograms but particularly noticeable in figure 32, a significant proportion of FHWA-observed latencies exceed 200 ms. This fact is due to the slow cellular connection used at the FHWA laboratory. Therefore, figure 34 shows the same data as figure 32 but extends the x-axis scale out to a maximum latency bin of >1,000 ms. The CDF and cumulative histogram plot for the same range can be seen in figure 35. More information about FHWA latencies is in the Key Findings section.



Source: FHWA.

Figure 34. Graph. Histogram of FHWA BSM latencies for all runs combined, 0 to >1,000 ms.



Source: FHWA.

Figure 35. Graph. CDF and cumulative histogram of FHWA BSM latencies for all runs combined, 0 to >1,000 ms.

The next group of tables featured in this report illustrate ping versus end-to-end latency (for example, table 8), which compares the end-to-end latency from the previous table to the ping latency between all sites. The ping latency between two sites is the time it takes for a simplified diagnostic packet to travel from one site to another and back. A common practice for ping testing is to use the Internet Control Message Protocol (ICMP), but TENA includes a ping testing capability with their TENA Console application, which leverages TENA formatted messages.(12,30) The use of TENA messages in the ping test more closely represents the transmission of test data, as IMCP messages do not operate on the same network transport layer as the UDP and Transmission Control Protocol (TCP) messages exchanged by TENA applications.(12,24,30,31) Ping latency is typically measured round trip (from source, to destination, and back), but the ping latency values on these tables were divided by two to match the end-to-end latency, which was measured one way. Each cell provides latency information for data sent from a source (SRC) site to a destination (DST) site. Each row contains data for a specific source site (indicated in the first column) and each column contains a destination site (indicated in the column header). Data from a particular source site to a particular destination site is shown in the intersecting row and column. For example, data from ORNL to UCLA can be found in the intersecting cell of the row SRC: ORNL (TN) and the column DST: UCLA (CA). Cells that have the same source and destination site are labeled not applicable, because sites did not send data to themselves. Cells may also be labeled for specific DCNs, which are detailed in the DCN section. Exactly why some Ping latencies are higher than their corresponding end-to-end latency is unclear. Researchers calculated these values using two different types of data, which may cause some variability.

Table 8. Event 2, Run 3 ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Ping: 111.931 E2E: 102.697	Ping: 128.106 E2E: 150.374	Ping: 148.849 E2E: 164.174	Ping: 126.698 E2E: 147.91
SRC: ANL (IL)	Ping: 131.216 E2E: 182.815	N/A	Ping: 35.400 E2E: 30.911	Ping: 48.855 E2E: 46.008	Ping: 32.306 E2E: 25.574
SRC: ORNL (TN)	Ping: 144.143 E2E: 192.931	Ping: 33.867 E2E: 35.946	N/A	Ping: 60.940 E2E: 61.522	Ping: 42.921 E2E: 40.15
SRC: UCLA (CA)	Ping: 163.495 E2E: 214.579	Ping: 47.952 E2E: 49.205	Ping: 59.760 E2E: 60.056	N/A	Ping: 55.023 E2E: 53.066
SRC: Mcity (MI)	Ping: 127.149 E2E: 235.499	Ping: 34.799 E2E: 37.08	Ping: 45.277 E2E: 46.714	Ping: 60.084 E2E: 61.204	N/A

E2E = end-to end.

Throughput results tables show the average network traffic for a given message type across a specific test run (for example, table 9). Each row of the tables contains the throughput data for a single site. Each cell in that row provides throughput data for the message type shown in the intersecting column. Data within the cell are shown in the form of upload and download rates, where the upload includes data sent to a single subscriber and download represents data received from all publishing sites.

Table 9. Event 2, Run 3 average throughput results to a single site.

Site	Vehicle SDO (KB/s)	TrafficLight SDO (KB/s)	J2735 (KB/s)	BSM SDO (KB/s)	Total (KB/s)
FHWA [CELL] (VA)	UP: 5.859 DOWN: 27.229	UP: 12.349 DOWN: 0	UP: 9.084 DOWN: 11.422	UP: 0 DOWN: 3.986	UP: 27.292 DOWN: 42.643
ANL (IL)	UP: 5.013 DOWN: 26.516	UP: 0 DOWN: 11.493	UP: 0 DOWN: 21.205	UP: 4.514 DOWN: 0	UP: 9.528 DOWN: 59.214
ORNL (TN)	UP: 4.892 DOWN: 26.816	UP: 0 DOWN: 10.395	UP: 2.909 DOWN: 18.111	UP: 0 DOWN: 5.613	UP: 7.801 DOWN: 61.516
UCLA (CA)	UP: 5.002 DOWN: 25.622	UP: 0 DOWN: 10.976	UP: 2.347 DOWN: 16.254	UP: 0 DOWN: 4.599	UP: 7.349 DOWN: 56.775
Mcity (MI)	UP: 11.989 DOWN: 15.232	UP: 0 DOWN: 10.301	UP: 6.355 DOWN: 12.535	UP: 0 DOWN: 5.134	UP: 18.345 DOWN: 43.296

DOWN = download rate; UP = upload rate.

The breakdown provided in table 9 enables the throughput estimation of individual components of a distributed test. For example, each site except for Mcity hosted a single vehicle, with Mcity hosting a total of three vehicles, but only two vehicles were running at a single time. This assertion can be validated in the throughput data, as FHWA, ANL, ORNL, and UCLA each showed an upload rate of approximately 5 to 6 KB/s, with Mcity at approximately 12 KB/s. With this information, researchers estimate that a single vehicle generates approximately 5 to 6 KB/s of data over the course of a distributed test event. In the same way, by looking at the upload rate of the TrafficLight SDOs at the FHWA site, researchers can confirm that FHWA hosted the traffic signal for the Event 2 data collection event and estimated that a single, four-phase intersection produces 10 to 12 KB/s. Researchers used interpretation of the throughput data for all Pilot 2 events to create throughput estimates for single instances of all used data types (table 10). These estimates can be used in future event planning to extrapolate bandwidth usage for a given test scenario.

Table 10. Throughput estimates for single instances of distributed testing data types.

Data Type	Throughput for Single Instance (KB/s)
Vehicle SDO (single vehicle)	5–6
TrafficLight SDO (single 4 phase intersection)	10–12
J2735 (single vehicle BSM)(12)	2–3
J2735 (single 4 phase intersection SPaT)(12)	6–7
J3224 (single pedestrian)	1
BSM SDO(12)	4–6

KEY FINDINGS

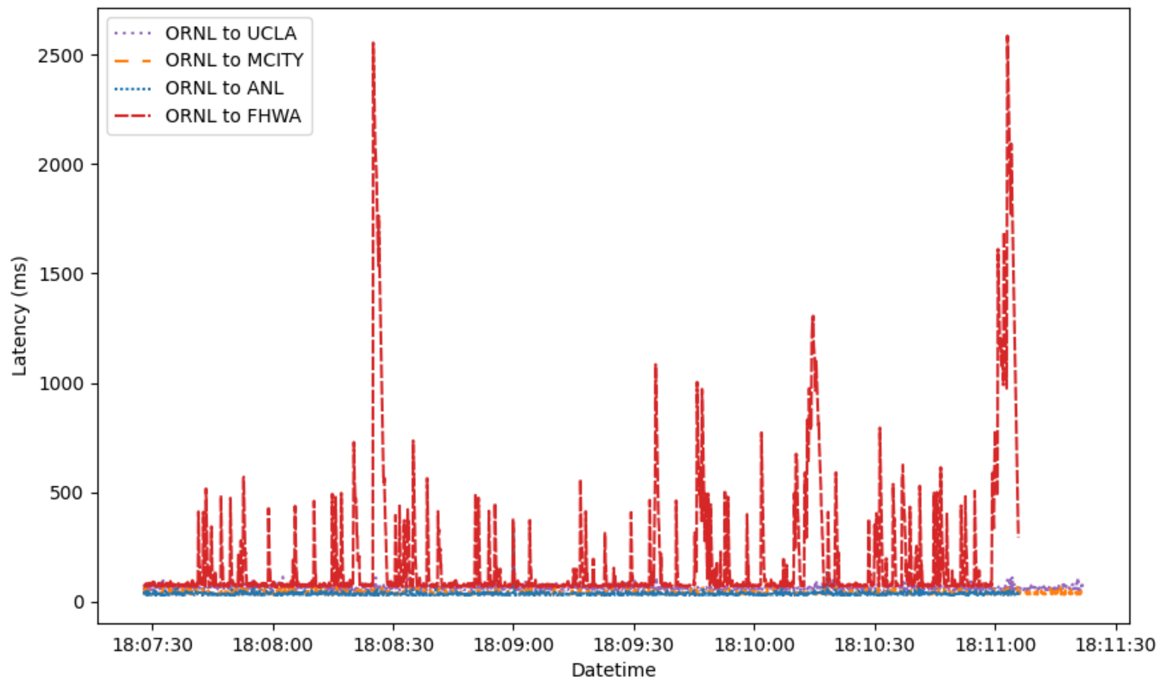
This section contains key findings of the Pilot 2 network performance results. Additional plots or tables are provided to support specific findings.

FHWA Network Latencies

The results from data originating at FHWA and arriving at FHWA shows a 2- to 4-times higher latency and up to 20-times higher jitter. This finding is almost certainly due to the cellular Internet connection used by the FHWA team. This connection is made up of four cellular modems with high-gain antennas bonded together. This type of connection is subject to the network performance of the nearby cellular towers and therefore suffers fluctuating bandwidth and latency. Depending on the day, the nominal network bandwidth has been measured between 5-25 Mb/s for download and 30 to 80 Mb/s for upload, and the network latency to common public services can range from 40 to 200 ms and jitter values of over 100 ms. This measurement aligns with the data from Pilot 2, Event 2 both in approximate end-to-end latency values as well as the high jitter. The distributed testing platform used in Pilot 2 can only perform as well as the network connection available to it, so therefore the results for the FHWA site are considered a result of their network connection, not of the performance of the system.

Latency Spikes

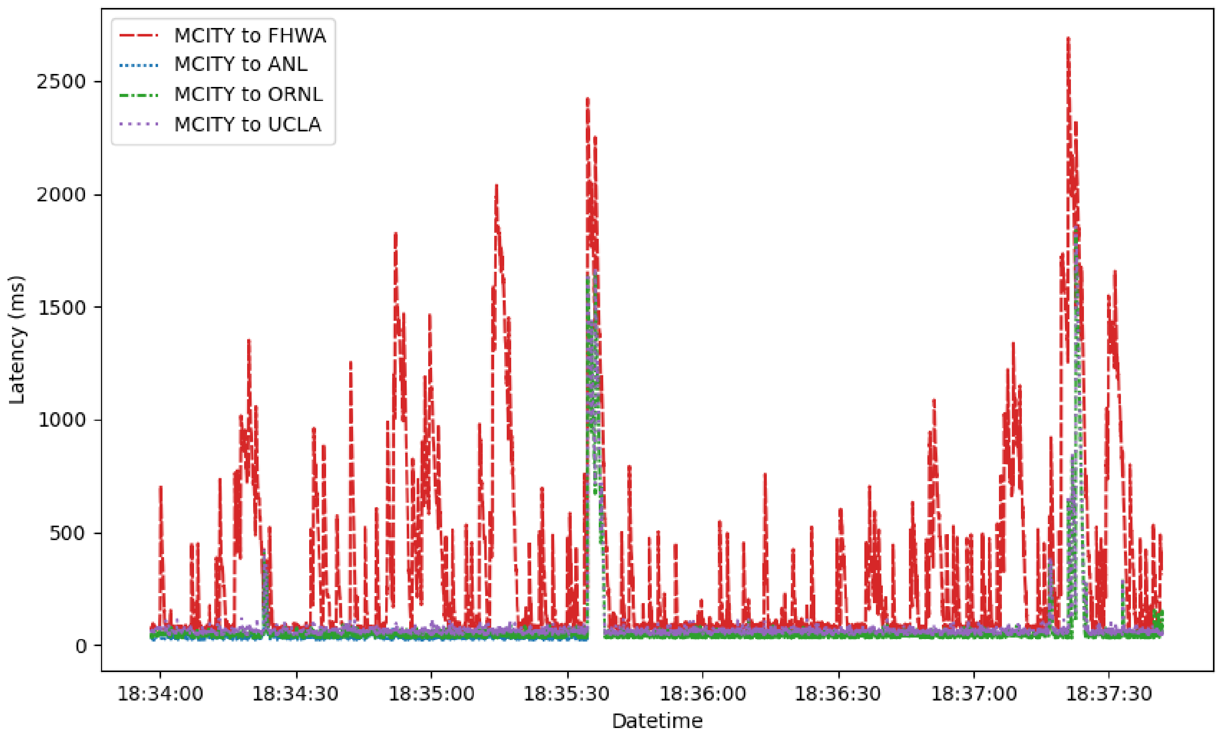
During testing, multiple occasions occurred where sites reported moments when participants in simulations would freeze, then quickly catch up or jump rapidly to their realtime locations. Most often, this occurrence would only happen on a single site, hinting toward a network issue for that specific site. One example of this occurrence can be seen in the spikes of latency values shown in the ORNL to FHWA data in figure 36. These spikes are likely caused by an unstable network connection or an overload of computer resources on the specific remote site.



Source: FHWA.
Note: FHWA in this diagram is FHWA [CELL].

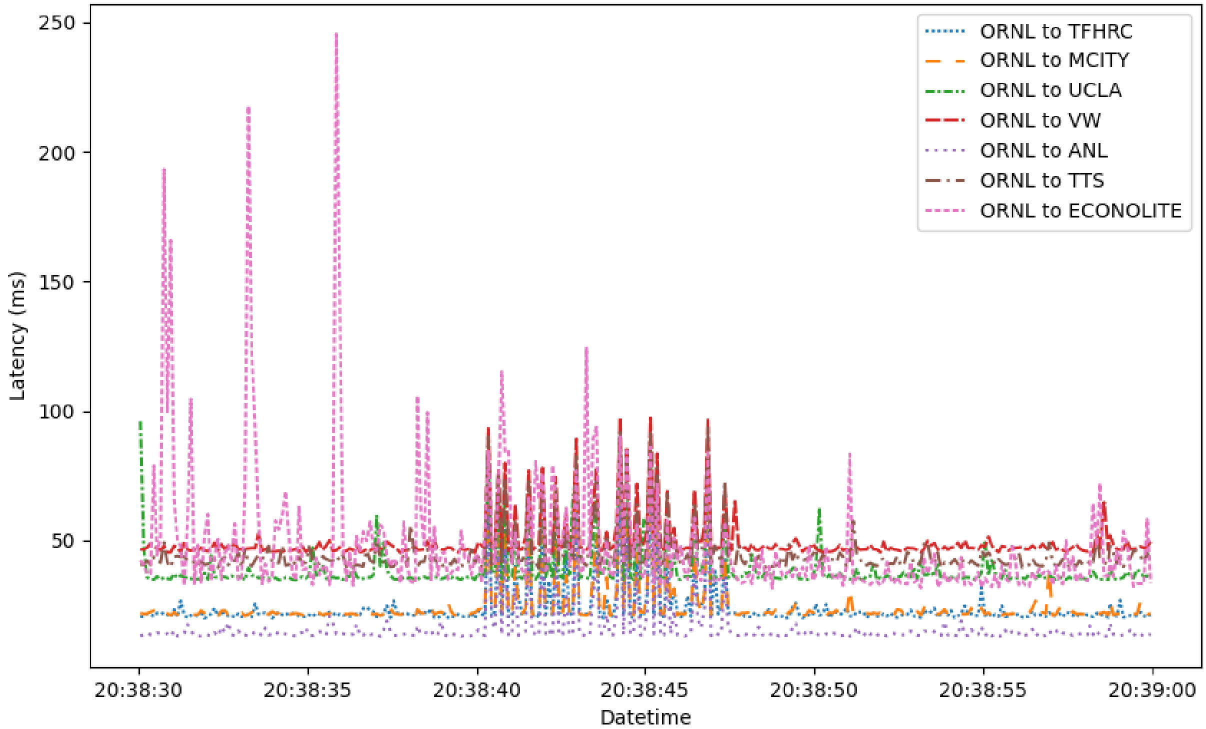
Figure 36. Graph. J2735 BSM end-to-end latency from ORNL for Event 2, Run 3.

In other plots, all destination sites showed a large spike at the same time, hinting toward an issue at the source site. This trend can be observed as a synchronized sharp decrease and increase in the latency of all sites' latency when the sites are overlaid on a single plot. This trend can be seen in the plot for Event 2, Run 5 in figure 37 at approximately 18:35:40 and 18:37:25. A short burst of increased latency can also indicate a network issue, as seen in figure 38 between 20:38:40 and 20:38:48. These spikes are likely caused by an unstable network connection or an overload of computer resources at the source site.



Source: FHWA.
*FHWA in this diagram is FHWA [CELL].

Figure 37. Graph. J2735 BSM end-to-end latency from Mcity for Event 2, Run 5*.



Source: FHWA.
*FHWA in this diagram is FHWA [FIBER].

Figure 38. Graph. J2735 BSM end-to-end latency from ORNL for Event 0*.

A key finding is the importance of a strong and stable network connection during distributed testing events. Since testing occurs in realtime, when a network spike occurs, external participants are not updated, but the local application continues. This circumstance temporarily leaves the local simulation with stale data, potentially leading to undesirable outcomes.

Network Latency Versus System Latency

Two types of latency make up an end-to-end latency in this distributed testing system. Network latency is the time it takes for data to travel from one site to another, and system latency is the delay overhead that the distributed testing system adds to the system in operations like converting back and forth to TENA message formats.(12)

The network latencies to and from each site are in the ping versus end-to-end latency charts (such as table 8) under ping latency, and the system latency can be found by subtracting the ping latency from the end-to-end latency. Except for data exchanges involving FHWA (likely due to the noted FHWA network issues), all average system latencies were below 10 ms, with most below 5 ms. Therefore, the research team concluded that end-to-end latencies are dominated by network latencies (30 to 60 ms), rather than overhead system latencies.

NETWORK SOLUTION COMPARISON

The events for Pilot 2 used two different VPN solutions: a cloud-based network for Events 0 and 1 and the VOICES portal for Event 2.(2,15) The cloud-based network and the VOICES portal have fundamentally different network architectures. The network is based on zero-trust connections to allow for direct-to-peer communication, while the VOICES portal is a more traditional hub and spoke solution, meaning data must flow through a central server before being routed to their final destination.

The participants and data exchanged varied for each event, so directly comparing one solution to the other is difficult, but some overlap exists that can be investigated. Specifically, TENA J2735 messages with BSM payloads were exchanged in events 0 and 2, with shared participants, including FHWA, ANL, UCLA, Mcity, and ORNL.(12) Data excerpts from this dataset are in table 11. FHWA is excluded from this table, as FHWA conducted events 0 and 2 in two different locations with two different Internet connections.

Table 11. J2735 BSM latency comparison between Pilot 2 events (ms).

Source to Destination	Event 0 (Network)	Event 2 (VOICES Portal)
ANL (IL) to UCLA (CA)	Avg: 30.084 Min: 29.457 Max: 37.359 Jitter: 0.512	Avg: 46.008 Min: 44.16 Max: 179.078 Jitter: 1.656
UCLA (CA) to ANL (IL)	Avg: 26.519 Min: 25.695 Max: 36.847 Jitter: 0.535	Avg: 49.205 Min: 43.516 Max: 191.293 Jitter: 6.569
ORNL (TN) to Mcity (MI)	Avg: 24.411 Min: 20.768 Max: 74.381 Jitter: 5.194	Avg: 40.15 Min: 32.777 Max: 341.045 Jitter: 11.031
Mcity (MI) to ORNL (TN)	Avg: 26.286 Min: 22.933 Max: 75.456 Jitter: 2.375	Avg: 46.714 Min: 29.242 Max: 495.357 Jitter: 18.233

COMPARISON TO PREVIOUS DISTRIBUTED TESTING EVENTS

Comparing the network performance of Pilot 2 with previous distributed testing events, such as SIT-1 and pilot 1, is difficult due to the differences in participants and network technologies used. SIT-1 participants included FHWA, MITRE, and Scientific Research Corporation and used direct site-to-site IP Security (IPsec) tunnels to connect all sites.(4) Pilot 1 participants included FHWA, UCLA, Nissan, and Econolite and leveraged the zero-trust, commercial-off-the-shelf (COTS) cloud-based network from Twingate, to connect all sites.(5,15) Pilot 2 participants included FHWA, UCLA, ANL, ORNL, Mcity, Econolite, TTS, and VW and leveraged both the COTS network and the VOICES Portal for different events.(2,15) The J2735 message format also changed from SIT-1 to Pilot 1, but this change resulted in a change of less than 1 ms in TENA message generation.(5)

Sample results from SIT-1 are in table 12, and sample data from Pilot 1 are in table 13.(4,5) Given the variables stated above and the high variability in data, even within a single event, few significant conclusions can be made when comparing the data between testing events. Even with variability, the end-to-end latency for most of the test events (excluding known network or time synchronization issues) remains within the same general range of 30 to 80 ms.

Note: At this time, the reason the latency is larger between MITRE and FHWA than between Scientific Research Corporation and FHWA is unclear, even though the former are geographically closer.

Table 12. SIT-1 Sample latency performance results using direct site-to-site IPsec tunnels.

Message Type, Source to Destination	Latency Results (ms)
BSM, FHWA* (VA) to MITRE (VA)	Avg: 55.376 Min: 35.809 Max: 149.720 Jitter: 12.330
BSM, MITRE (VA) to Scientific Research Corporation (GA)	Avg: 39.414 Min: 8.430 Max: 2181.260 Jitter: 7.873
BSM, Scientific Research Corporation (GA) to FHWA* (VA)	Avg: 16.273 Min: 10.88 Max: 75.500 Jitter: 6.616

*FHWA used the FHWA development network for SIT-1, which has a gigabit fiber connection to the Internet.

Table 13. Pilot 1 sample latency performance results using COTS cloud-based network.

Message Type, Source to Destination	Latency Results (ms)
MAP, Econolite (OH) to UCLA (CA)	Avg: 64.945 Min: 51.800 Max: 123.850 Jitter: 12.436
BSM, UCLA (CA) to Nissan (MI)	Avg: 46.769 Min: 44.32 Max: 50.407 Jitter: 0.439
BSM, Nissan (MI) to Econolite (OH)	Avg: 65.731 Min: 58.26 Max: 101.933 Jitter: 5.919

CHAPTER 5. LESSONS LEARNED AND CONCLUSIONS

Pilot 2 offered a unique opportunity to collaboratively test with others in a cosimulation environment, allowing participants to discover how their applications interacted with other applications. This chapter discusses the key findings from this testing and the lessons learned that can be applied to future development. The conclusion of this report outlines accomplishments and findings and reflects on the implications of performing this pilot test.

LESSONS LEARNED

Some lessons learned of Pilot 2 include:

- The biggest challenge of distributed testing was logistical: Coordinating with multiple organizations to find a common time when test assets were available had the potential to be complicated.
- Distributed testing helped quickly identify and resolve potential conflicting behaviors between different agents. Almost every organization tweaked their simulated and physical test assets (often by retuning controller gains) to avoid crashing in the cosimulation environment. Such fast, iterative development in simulation can lead to a more robust and efficient development process.
- Crashes between the simulated vehicles showed how distributed testing could cost-effectively improve the safety of systems before proceeding to live testing.
- A process flow to collect energy consumption data among multiple physical and simulated vehicles was successfully demonstrated. Although the low speed of the Pilot 2, Event 2 test event meant that the results of the Event 2 energy consumption analysis were inconclusive, the team gained a better understanding of what would be required to collect more representative data in the future. For example, higher vehicle speeds and longer test durations may be needed to provide enough statistically representative data. These changes might be achieved by either driving on a larger track or driving the same loop multiple times.

KEY FINDINGS

Pilot 2 provided organizations with the opportunity to collaboratively test on a cosimulation platform, enabling assessment of CDA applications and how multiple organizations' applications interact. Researchers learned many lessons through this pilot testing—lessons that can be applied to future testing and development of the CDA applications.

For CARMA Platform

Testing CARMA Platform highlighted several areas for improvement.(6) First, the platform's vehicle controls clearly tuning required enhancements to better interact with other simulated vehicles. CARMA Platform is typically tested in isolation or with other CARMA Platform instances, so integration with different vehicle controllers provided valuable insights and training opportunities to refine the platform.[6] Additionally, the testing revealed a need for improved processing of the J2735 MAP to more accurately generate world models.[13] Without this pilot test, the platform's MAP accuracy would have remained suboptimal, hindering its ability to geolocate itself and other CDA devices.[13] Finally, an issue with the enhanced yield functionality was identified, as this functionality failed to account for static objects on the road, leading to potential collisions. Addressing these learned lessons will significantly improve the CARMA Platform's overall performance and safety.[6]

For CDASim

Throughout Pilot 2, CARMA Platform used the CARMA-CARLA integration within the CDASim toolkit. [6,17]

During testing, researchers discovered issues with the CARMA-CARLA integration. The events of Pilot 2 involved testing this integration outside of the usual CDASim operating environment and in realtime, uncovering some dependencies on the rest of CDASim.[17] These dependencies meant that the CARMA–CARLA integration would not be able to operate without the rest of the CDASim components, which limited its usability as a standalone integration.[17] Therefore, researchers created a separate distributed testing variant of CDASim to address these dependencies and allow for the tool's operation within a distributed test event.

Additionally, the simultaneous development of CDASim and distributed testing software, both utilizing similar technologies, created a valuable opportunity for cross-team collaboration.[17] This collaboration facilitated the exchange of ideas and co-project troubleshooting, benefiting both projects. The pilot test also uncovered an unresolved issue with Carla-sensor-lib, or the CARLA sensor library, which contains functions, such as the filtering of occluded objects, where moving vehicles were incorrectly classified as being obstructed while using CARLA's simulated Light Detection and Ranging (LiDAR).[7] The CARLA sensor library was mistakenly filtering out moving vehicles, even when the vehicles were within the sensor's line of sight. This mistake prevented the CARMA Platform vehicle from seeing other simulated vehicles using the CARLA-simulated LiDAR. These discoveries provide critical insights for further refining the simulation tools.[6,7]

For VOICES Portal

By using the VOICES portal for testing, portal developers were able to obtain direct feedback from test participants for improvements or areas to focus future development on.[2] The research team suggested and implemented the addition of a TENA EM domain name system entry, streamlining connections to test events.[12] In addition, researchers added the Pilot 2 TENA OMs to event logs for use in Pilot 2 test events. Another implemented enhancement involved providing the option to select between UDP or TCP for VPN data transport.[24,31] This enhancement helped to avoid an issue called TCP meltdown, which is where data are sent via TCP and encounter exponentially increasing rebroadcast rates as multiple TCP layers attempt to rebroadcast the data at the same time.[31] These enhancements and discoveries contributed to the overall refinement of the system and were valuable lessons learned from the pilot test.

For ANL

Lessons learned from Pilot 2 highlighted some advancement opportunities for ANL's simulation capabilities. By participating in Pilot 2, ANL was able to use test infrastructure and a new configuration to advance the simulation capabilities of their dynamometer test system. ANL was also able to enhance their systems to receive and decode standard J2735 BSM messages to gain an awareness of other vehicles in the simulation.(13) The systems were also enhanced to receive and decode standard J2735 SPaT messages, allowing their simulated vehicle to stop at a simulated traffic signal during testing.

For Easier Distribution and Configuration Management of Software

By using Docker containers, the team was able to quickly and easily share and update the distributed testing software platform for a collaborative surface transportation use case. The integration of Docker streamlined the deployment and management of software components, ensuring a more efficient and scalable setup for the collaborative testing environment.(25)

For UCLA OpenCDA

Over the course of this testing, UCLA was able to improve its platform by adding the capability to receive and decode standard J2735 SPaT messages, allowing their OpenCDA vehicle to stop at simulated traffic signals.(8,13)

For ORNL

Through Pilot 2 testing, ORNL was able to develop their driving platform for integration with CARLA.(7) This development offered a new platform for testing existing ecodriving algorithms and should enable opportunities for future collaborative testing with other CARLA users.

For Mcity Digital Twin

Pilot 2 gave Mcity an opportunity and audience to collaboratively hone their map and build their digital twin. Mcity was able to enhance its digital map to better match real world geometries for more accurate and realistic simulated vehicle movements. CARLA map improvements were also applied to resolve issues with traffic light integration, enabling Mcity to create a more realistic simulation environment.(7) The Mcity digital twin capability can now host mixed-reality test events, such as those events from distributed testing. This goal is major focus as part of their National Science Foundation-funded Mcity 2.0 project.(32) The Pilot 2 test campaign also provided an opportunity to get real-world experience with typical data and workflows and to do such alongside established and experienced members of industry.

CONCLUSION

The Pilot 2 test campaign successfully accomplished the objective of conducting a geographically distributed test, with multiple test sites, using a variety of simulated models and physical platforms over a cloud-based network. Pilot 2 was able to demonstrate end-to-end latencies and distributed testing overhead that are comparable to previous successful test events: 30−60 ms and 5−10 ms, respectively. The importance of a strong and stable network connection was also confirmed during analysis of the latency and jitter results. Throughput results across all data types for all events not only provided insights into how much bandwidth was used but also enabled the extrapolation of these data to provide estimates for future test events. All of this testing was accomplished on two different secure network solutions.

Pilot 2 demonstrated how such testing might be used to inform, harmonize, or advance adoption of a standards development effort such as the SAE J3224 Sensor Data-Sharing Message or the ETSI Cooperative Perception Message.(14,23) By streamlining how researchers can work together and using standard message formats, Pilot 2 illustrated how distributed testing can improve the technical maturity of different applications and help ensure that algorithms and software from a variety of sources can work together safely to answer research questions that are important to the government, industry, and academia.

APPENDIX A. DETAILED NETWORK PERFORMANCE RESULTS

This appendix describes detailed network performance results for all test sites.

DETAILED LATENCY RESULTS

Table 14 describes the end-to-end latency results for J2735 BSMs sent during Event 0 from the source sites to the destination sites. Table 15 compares the ping (one-way) and end-to-end latency results for J2735 BSMs sent during Event 0. Table 16 describes the end-to-end latency results for J3224 SDSMs sent during Event 1, and table 17 compares the ping (one-way) and end-to-end latency results for J3224 SDSMs sent during Event 1. Table 18 describes the end-to-end latency results for J2735 SPaT messages sent during Event 1, and table 19 compares the ping (one-way) and end-to-end latency results for J2735 SPaT messages sent during Event 1.

Table 20 describes the end-to-end latency results for J2735 BSMs sent during Event 2, Run 1 and table 21 compares the ping (one-way) to end-to-end latency results for J2735 BSMs sent during Event 2, Run 1. Table 22 describes end-to-end latency results for J2735 SPaT messages sent during Event 2 Run 1 and table 23 compares ping (one-way) to end-to-end latency results for J2735 SPaT messages sent during event, 2 Run 1. Table 24 describes the end-to-end latency results for J2735 BSMs sent during Event 2, Run 2 and table 25 compares ping (one-way) to end-to-end latency for J2735 BSMs sent during event 2, Run 2. Table 26 describes the end-to-end latency results for J2735 SPaT messages sent during Event 2, Run 2. Table 27 compares the one-way ping to end-to-end latency results for J2735 SPaT messages sent during Event 2, Run 2.

Table 28 and table 29 both describe results from FHWA sending SPaT messages to all the other test sites during Event 2, Run 3. Table 28 describes the end-to-end latency for this run, and table 29 compares this run's one-way ping to the end-to-end latency. Table 30 describes the end-to-end latency results for J2735 BSMs sent during Event 2, Run 4, and table 31 compares one-way ping to end-to-end latency results for J2735 BSMs sent during Event 2, Run 4. Table 32 and table 33 describe results from Event 2, Run 4 where FHWA [cell] sent J2735 SPaT messages to the rest of the test sites. Table 32 describes the end-to-end latency results and table 33 compares them to the one-way ping. Table 34 describes the latency results for J2735 BSMS sent during Event 2, Run 5 and table 35 compares those results to the one-way ping results. Table 36 describes the end-to-end latency results for J2735 SPaT messages sent from FHWA [cell] to all other test sites during Event 2, Run 5. Table 37 compares those results to the one-way ping results.

Table 14. Event 0—J2735 BSM—end-to-end latency results (ms).

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)	SRC: Econolite (OH)	DST: VW (CA)	DST: TTS (OR)
SRC: FHWA [FIBER] (VA):	N/A	Avg: 18.667 Min: 8.425 Max: 46.947 Jitter: 2.269	Avg: 31.947 Min: 19.426 Max: 90.903 Jitter: 2.887	Avg: 39.061 Min: 28.986 Max: 74.528 Jitter: 2.194	Avg: 26.434 Min: 15.540 Max: 57.246 Jitter: 2.265	Avg: 43.470 Min: 15.530 Max: 223.973 Jitter: 5.903	Avg: 56.339 Min: 42.496 Max: 120.871 Jitter: 3.205	Avg: 50.601 Min: 38.632 Max: 82.723 Jitter: 2.705
SRC: ANL (IL)	Avg: 17.379 Min: 16.739 Max: 20.701 Jitter: 0.473	N/A	Avg: 29.348 Min: 26.608 Max: 76.395 Jitter: 3.226	Avg: 30.084 Min: 29.457 Max: 37.359 Jitter: 0.512	Avg: 10.703 Min: 9.923 Max: 23.856 Jitter: 0.823	Avg: 39.005 Min: 30.424 Max: 129.353 Jitter: 7.287	Avg: 47.687 Min: 42.864 Max: 150.639 Jitter: 4.885	Avg: 40.145 Min: 37.352 Max: 83.453 Jitter: 2.428
SRC: ORNL (TN)	Avg: 23.963 Min: 19.83 Max: 71.812 Jitter: 5.721	Avg: 16.35 Min: 12.761 Max: 64.099 Jitter: 5.266	N/A	Avg: 39.104 Min: 34.696 Max: 95.956 Jitter: 6.021	Avg: 24.411 Min: 20.768 Max: 74.381 Jitter: 5.194	Avg: 48.015 Min: 31.51 Max: 245.447 Jitter: 15.726	Avg: 49.42 Min: 45.103 Max: 97.422 Jitter: 5.474	Avg: 44.753 Min: 39.262 Max: 94.815 Jitter: 6.502
SRC: UCLA (CA)	Avg: 32.419 Min: 31.939 Max: 34.488 Jitter: 0.248	Avg: 26.519 Min: 25.695 Max: 36.847 Jitter: 0.535	Avg: 51.072 Min: 45.712 Max: 131.13 Jitter: 2.162	N/A	Avg: 26.71 Min: 26.105 Max: 37.425 Jitter: 0.382	Avg: 72.187 Min: 49.9 Max: 361.628 Jitter: 7.391	Avg: 14.465 Min: 11.783 Max: 44.114 Jitter: 1.195	Avg: 19.319 Min: 16.087 Max: 38.817 Jitter: 1.699

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)	SRC: Econolite (OH)	DST: VW (CA)	DST: TTS (OR)
SRC: Mcity (MI)	Avg: 18.16 Min: 17.53 Max: 25.36 Jitter: 0.295	Avg: 8.362 Min: 6.983 Max: 13.857 Jitter: 0.885	Avg: 26.286 Min: 22.933 Max: 75.456 Jitter: 2.375	Avg: 29.811 Min: 28.883 Max: 42.4 Jitter: 0.522	N/A	Avg: 37.782 Min: 27.038 Max: 159.851 Jitter: 7.721	Avg: 43.765 Min: 39.499 Max: 138.588 Jitter: 4.251	Avg: 37.328 Min: 33.45 Max: 84.215 Jitter: 2.684
SRC: Econolite (OH)	Avg: 19138.471 Min: 9222.114 Max: 30690.482 Jitter: 132.445	Avg: 8015.686 Min: 49.197 Max: 17581.234 Jitter: 150.802	Avg: 14426.967 Min: 1456.689 Max: 27752.129 Jitter: 113.696	Avg: 7226.36 Min: 75.544 Max: 14783.099 Jitter: 158.130	Avg: 10050.191 Min: 60.717 Max: 22750.804 Jitter: 120.798	N/A	Avg: 8615.936 Min: 100.27 Max: 20332.152 Jitter: 165.982	Avg: 9951.274 Min: 483.163 Max: 18952.895 Jitter: 130.183
SRC: VW (CA)	Avg: 39.163 Min: 37.646 Max: 43.308 Jitter: 0.949	Avg: 32.163 Min: 30.7 Max: 41.349 Jitter: 1.124	Avg: 50.909 Min: 46.493 Max: 97.123 Jitter: 3.189	Avg: 8.315 Min: 7.144 Max: 14.099 Jitter: 0.759	Avg: 40.13 Min: 38.751 Max: 56.061 Jitter: 1.048	Avg: 59.983 Min: 50.465 Max: 227.124 Jitter: 8.314	N/A	Avg: 24.377 Min: 20.425 Max: 49.087 Jitter: 2.734
SRC: TTS (OR)	Avg: 52.092 Min: 40.812 Max: 109.823 Jitter: 5.872	Avg: 36.281 Min: 25.7 Max: 91.436 Jitter: 5.763	Avg: 62.654 Min: 48.2 Max: 116.182 Jitter: 7.897	Avg: 28.193 Min: 18.249 Max: 83.65 Jitter: 5.442	Avg: 44.486 Min: 33.912 Max: 100.144 Jitter: 5.686	Avg: 73.348 Min: 52.512 Max: 318.321 Jitter: 13.729	Avg: 45.145 Min: 34.039 Max: 153.999 Jitter: 7.216	N/A

CA = California; IL = Illinois; MI = Michigan; OH = Ohio; OR = Oregon; TN = Tennessee; VA = Virginia.

Table 15. Event 0—J2735 BSM—ping (one-way) versus end-to-end latency.

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)	SRC: Econolite (OH)	DST: VW (CA)	DST: TTS (OR)
SRC: FHWA [FIBER] (VA)	N/A	Ping: 13.660 E2E: 18.667	Ping: 29.113 E2E: 31.947	Ping: 31.607 E2E: 39.061	Ping: 17.744 E2E: 26.434	Ping: No data (DCN7) E2E: 43.470	Ping: No data (DCN7) E2E: 56.339	Ping: No data (DCN7) E2E: 50.601
SRC: ANL (IL)	Ping: 13.843 E2E: 17.379	N/A	Ping: 25.302 E2E: 29.348	Ping: 27.625 E2E: 30.084	Ping: 8.843 E2E: 10.703	Ping: No data (DCN7) E2E: 39.005	Ping: No data (DCN7) E2E: 47.687	Ping: No data (DCN7) E2E: 40.145
SRC: ORNL (TN)	Ping: 24.319 E2E: 23.963	Ping: 22.854 E2E: 16.35	N/A	Ping: 42.020 E2E: 39.104	Ping: 26.177 E2E: 24.411	Ping: No data (DCN7) E2E: 48.015	Ping: No data (DCN7) E2E: 49.420	Ping: No data (DCN7) E2E: 44.753
SRC: UCLA (CA)	Ping: 31.219 E2E: 32.419	Ping: 27.544 E2E: 26.519	Ping: 46.985 E2E: 51.072	N/A	Ping: 27.845 E2E: 26.710	Ping: No data (DCN7) E2E: 72.187	Ping: No data (DCN7) E2E: 14.465	Ping: No data (DCN7) E2E: 19.319
SRC: Mcity (MI)	Ping: 17.789 E2E: 18.16	Ping: 8.677 E2E: 8.362	Ping: 29.635 E2E: 26.286	Ping: 27.812 E2E: 29.811	N/A	Ping: No data (DCN7) E2E: 37.782	Ping: No data (DCN7) E2E: 43.765	Ping: No data (DCN7) E2E: 37.328

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)	SRC: Econolite (OH)	DST: VW (CA)	DST: TTS (OR)
SRC: Econolite (OH)	Ping: No data (DCN7) E2E: 19138.471	Ping: No data (DCN7) E2E: 8015.686	Ping: No data (DCN7) E2E: 14426.967	Ping: No data (DCN7) E2E: 7226.360	Ping: No data (DCN7) E2E: 10050.191	N/A	Ping: No data (DCN7) E2E: 8615.936	Ping: No data (DCN7) E2E: 9951.274
SRC: VW (CA)	Ping: No data (DCN7) E2E: 39.163	Ping: No Data (DCN7) E2E: 32.163	Ping: No Data (DCN7) E2E: 50.909	Ping: No Data (DCN7) E2E: 8.315	Ping: No Data (DCN7) E2E: 40.130	Ping: No Data (DCN7) E2E: 59.983	N/A	Ping: No data (DCN7) E2E: 24.377
SRC: TTS (OR)	Ping: No data (DCN7) E2E: 52.092	Ping: No data (DCN7) E2E: 36.281	Ping: No data (DCN7) E2E: 62.654	Ping: No data (DCN7) E2E: 28.193	Ping: No data (DCN7) E2E: 44.486	Ping: No data (DCN7) E2E: 73.348	Ping: No data (DCN7) E2E: 45.145	N/A

Table 16. Event 1—J3224 SDSM—end-to-end latency results.

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: TTS (OR)	DST: VW (CA)
SRC: Mcity (MI)	Avg: 16.49 Min: 16.014 Max: 19.348 Jitter: 0.131	Avg: 61.495 Min: 46.236 Max: 204.973 Jitter: 11.157	Avg: 42.848 Min: 41.238 Max: 95.145 Jitter: 1.369

Table 17. Event 1—J3224 SDSM—ping (one-way) versus end-to-end latency.

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: TTS (OR)	DST: VW (CA)
SRC: Mcity (MI)	Ping: 17.573 E2E: 16.49	Ping: 48.701 E2E: 61.495	Ping: 42.146 E2E: 42.848

Table 18. Event 1—J2735 SPaT—end-to-end latency results.

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: TTS (OR)	DST: VW (CA)
SRC: Mcity (MI)	Avg: 16.897 Min: 16.215 Max: 28.062 Jitter: 0.169	Avg: 60.392 Min: 46.351 Max: 268.08 Jitter: 19.339	Avg: 43.265 Min: 41.603 Max: 163.232 Jitter: 1.268

Table 19. Event 1—J2735 SPaT—ping (one-way) versus end-to-end latency.

SRC or DST Sites	DST: FHWA [FIBER] (VA)	DST: TTS (OR)	DST: VW (CA)
SRC: Mcity (MI)	Ping: 17.573 E2E: 16.897	Ping: 48.701 E2E: 60.392	Ping: 42.146 E2E: 43.265

Table 20. Event 2, Run 1—J2735 BSM—end-to-end latency results (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	None (DCN3)	None (DCN3)	None (DCN3)	None (DCN3)
SRC: ANL (IL)	Avg: 410.525 Min: 54.851 Max: 3432.653 Std. Dev: 68.176	N/A	Avg: 30.815 Min: 28.584 Max: 145.424 Std. Dev: 1.782	Avg: 44.025 Min: 42.07 Max: 158.824 Std. Dev: 1.759	Avg: 24.360 Min: 22.279 Max: 139.577 Std. Dev: 1.671
SRC: ORNL (TN)	Min: 65.658 Max: 4062.256 Avg: 696.815 Jitter: 896.947	Min: 28.977 Max: 194.218 Avg: 36.131 Jitter: 9.424	N/A	Min: 51.836 Max: 153.27 Avg: 60.37 Jitter: 10.825	Min: 32.168 Max: 133.902 Avg: 40.013 Jitter: 8.959
SRC: UCLA (CA)	Min: 82.991 Max: 4691.933 Avg: 852.277 Jitter: 1282.837	Min: 45.291 Max: 238.747 Avg: 51.276 Jitter: 8.187	Min: 54.844 Max: 248.736 Avg: 61.836 Jitter: 10.269	N/A	Min: 48.294 Max: 242.737 Avg: 54.424 Jitter: 8.333
SRC: Mcity (MI)	Min: 61.446 Max: 4406.845 Avg: 395.418 Jitter: 673.507	Min: 22.96 Max: 188.55 Avg: 38.65 Jitter: 9.838	Min: 32.578 Max: 197.665 Avg: 48.715 Jitter: 11.247	Min: −108.746 Max: 133.338 Avg: 61.552 Jitter: 10.566	N/A

Table 21. Event 2, Run 1—J2735 BSM—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Ping: 68.220 E2E: None (DCN3)	Ping: 78.833 E2E: None (DCN3)	Ping: 95.866 E2E: None (DCN3)	Ping: 77.833 E2E: None (DCN3)
SRC: ANL (IL)	Ping: 69.440 E2E: 410.525	N/A	Ping: 31.466 E2E: 30.815	Ping: 48.600 E2E: 44.025	Ping: 32.866 E2E: 24.360
SRC: ORNL (TN)	Ping: 77.967 E2E: 696.815	Ping: 31.6 E2E: 36.131	N/A	Ping: 57.278 E2E: 60.37	Ping: 42.556 E2E: 40.013
SRC: UCLA (CA)	Ping: 93.033 E2E: 852.277	Ping: 49 E2E: 51.276	Ping: 63 E2E: 61.836	N/A	Ping: 58.889 E2E: 54.424
SRC: Mcity (MI)	Ping: 75.2 E2E: 395.418	Ping: 30.033 E2E: 38.65	Ping: 40.5 E2E: 48.715	Ping: 54.167 E2E: 61.552	N/A

Table 22. Event 2, Run 1—J2735 SPaT—end-to-end latency results (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Avg: 535.794 Min: 58.927 Max: 5360.98 Jitter: 937.275	Avg: 568.024 Min: 68.186 Max: 5120.244 Jitter: 1018.973	Avg: 609.752 Min: 81.252 Max: 4858.693 Jitter: 1009.075	Avg: 600.893 Min: 61.524 Max: 6618.974 Jitter: 1110.07

Table 23. Event 2, Run 1—J2735 SPaT—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Ping: 68.220 E2E: 535.794	Ping: 78.833 E2E: 568.024	Ping: 95.866 E2E: 609.752	Ping: 77.833 E2E: 600.893

Table 24. Event 2, Run 2—J2735 BSM—end-to-end latency results (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	None (DCN3)	None (DCN3)	None (DCN3)	None (DCN3)
SRC: ANL (IL)	Avg: 733.414 Min: 54.736 Max: 7341.227 Std. Dev: 79.351	N/A	Avg: 31.499 Min: 29.566 Max: 154.749 Std. Dev: 1.786	Avg: 46.212 Min: 44.169 Max: 169.123 Std. Dev: 1.918	Avg: 25.658 Min: 23.447 Max: 148.933 Std. Dev: 1.840
SRC: ORNL (TN)	Min: 65.33 Max: 9428.318 Avg: 1461.13 Jitter: 2626.78	Min: 29.198 Max: 80.601 Avg: 37.378 Jitter: 7.67	N/A	Min: 53.438 Max: 121.802 Avg: 63.734 Jitter: 12.573	Min: 32.497 Max: 73.181 Avg: 42.238 Jitter: 8.914
SRC: UCLA (CA)	Min: −3.024 Max: 9540.73 Avg: 291.817 Jitter: 651.027	Min: 44.07 Max: 83.451 Avg: 49.969 Jitter: 6.577	Min: 53.414 Max: 3746.711 Avg: 208.542 Jitter: 595.003	N/A	Min: 47.423 Max: 3737.036 Avg: 202.226 Jitter: 595.575
SRC: Mcity (MI)	Min: 60.11 Max: 11034.915 Avg: 1326.271 Jitter: 2827.833	Min: 22.776 Max: 65.906 Avg: 37.86 Jitter: 8.38	Min: 31.902 Max: 88.115 Avg: 47.456 Jitter: 9.212	Min: 46.731 Max: 108.323 Avg: 62.267 Jitter: 9.896	N/A

Table 25. Event 2, Run 2—J2735 BSM—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Ping: 69.773 E2E: None (DCN3)	Ping: 78.622 E2E: None (DCN3)	Ping: 94.614 E2E: None (DCN3)	Ping: 77.351 E2E: None (DCN3)
SRC: ANL (IL)	Ping: 85.410 E2E: 733.414	N/A	Ping: 33.764 E2E: 31.499	Ping: 48.815 E2E: 46.212	Ping: 31.807 E2E: 25.658
SRC: ORNL (TN)	Ping: 85.792 E2E: 1461.13	Ping: 34.162 E2E: 37.378	N/A	Ping: 61.244 E2E: 63.734	Ping: 43.526 E2E: 42.238
SRC: UCLA (CA)	Ping: 106.455 E2E: 291.817	Ping: 27.5 E2E: 49.969	Ping: 60.452 E2E: 208.542	N/A	Ping: 57.939 E2E: 202.226
SRC: Mcity (MI)	Ping: 123.955 E2E: 1326.271	Ping: 32.991 E2E: 37.86	Ping: 45.919 E2E: 47.456	Ping: 59.394 E2E: 62.267	N/A

Table 26. Event 2, Run 1—J2735 SPaT—end-to-end latency results (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Avg: 425.854 Min: 57.676 Max: 4818.238 Jitter: 891.916	Avg: 480.058 Min: 67.3 Max: 4828.3 Jitter: 933.964	Avg: 484.553 Min: 81.366 Max: 5129.878 Jitter: 959.565	Avg: 447.067 Min: 62.184 Max: 4845.702 Jitter: 923.973

Table 27. Event 2, Run 2—J2735 SPaT—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Ping: 47.5 E2E: 425.854	Ping: 78.623 E2E: 480.058	Ping: 94.614 E2E: 484.553	Ping: 77.351 E2E: 447.067

Table 28. Event 2, Run 3—J2735 SPaT—end-to-end latency results (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Avg: 259.758 Min: 61.702 Max: 6666.356 Jitter: 725.043	Avg: 163.038 Min: 71.448 Max: 3339.671 Jitter: 330.425	Avg: 209.312 Min: 85.772 Max: 3740.188 Jitter: 421.157	Avg: 169.056 Min: 65.363 Max: 3333.624 Jitter: 351.907

Table 29. Event 2, Run 3—J2735 SPaT—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Ping: 111.931 E2E: 259.758	Ping: 128.106 E2E: 163.038	Ping: 148.849 E2E: 209.312	Ping: 126.698 E2E: 169.056

Table 30. Event 2, Run 4—J2735 BSM—end-to-end latency results (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	None (DCN3)	None (DCN3)	None (DCN3)	None (DCN3)
SRC: ANL (IL)	Avg: 229.515 Min: 55.385 Max: 5367.922 Jitter: 47.622	N/A	Avg: 31.531 Min: 29.152 Max: 170.604 Jitter: 2.330	Avg: 46.217 Min: 44.229 Max: 185.383 Jitter: 1.740	Avg: 25.648 Min: 23.377 Max: 164.694 Jitter: 2.127
SRC: ORNL (TN)	Avg: 236.703 Min: 54.71 Max: 5378.796 Jitter: 599.942	Avg: 36.605 Min: 29.232 Max: 139.542 Jitter: 9.027	N/A	Avg: 61.774 Min: 53.487 Max: 161.228 Jitter: 10.839	Avg: 40.437 Min: 32.734 Max: 160.156 Jitter: 9.735
SRC: UCLA (CA)	Avg: 155.335 Min: 79.359 Max: 1405.302 Jitter: 157.477	Avg: 47.947 Min: −45.581 Max: 195.797 Jitter: 7.08	Avg: 57.523 Min: −35.959 Max: 194.812 Jitter: 8.326	N/A	Avg: 51.167 Min: −41.845 Max: 216.843 Jitter: 7.957
SRC: Mcity (MI)	Avg: 253.092 Min: 56.528 Max: 5573.988 Jitter: 524.163	Avg: 45.495 Min: 20.157 Max: 1394.987 Jitter: 81.161	Avg: 55.172 Min: 29.291 Max: 1401.633 Jitter: 81.715	Avg: 70.41 Min: 44.173 Max: 1418.851 Jitter: 83.105	N/A

Table 31. Event 2, Run 4—J2735 BSM—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Ping: 75.507 E2E: None (DCN3)	Ping: 82.786 E2E: None (DCN3)	Ping: 93.643 E2E: None (DCN3)	Ping: 82.369 E2E: None (DCN3)
SRC: ANL (IL)	Ping: 131.216 E2E: 229.515	N/A	Ping: 35.400 E2E: 31.531	Ping: 48.855 E2E: 46.217	Ping: 32.306 E2E: 25.648
SRC: ORNL (TN)	Ping: 152.95 E2E: 236.703	Ping: 33 E2E: 36.605	N/A	Ping: 58.021 E2E: 61.774	Ping: 39.25 E2E: 40.437
SRC: UCLA (CALIFORNIA)	Ping: 94.513 E2E: 155.335	Ping: 47.438 E2E: 47.947	Ping: 57.896 E2E: 57.523	N/A	Ping: 60.333 E2E: 51.167
SRC: Mcity (MI)	Ping: 121.27 E2E: 253.092	Ping: 30.82 E2E: 45.495	Ping: 40.467 E2E: 55.172	Ping: 56.367 E2E: 70.41	N/A

Table 32. Event 2, Run 4—J2735 SPaT—end-to-end latency results (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Avg: 230.906 Min: 57.036 Max: 6,226.223 Jitter: 648.659	Avg: 252.463 Min: 65.585 Max: 6,235.353 Jitter: 682.238	Avg: 264.408 Min: 84.382 Max: 6,250.639 Jitter: 690.53	Avg: 236.551 Min: 63.711 Max: 6,229.675 Jitter: 660.816

Table 33. Event 2, Run 4—J2735 SPaT—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CALIFORNIA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Ping: 75.507 E2E: 230.906	Ping: 82.786 E2E: 252.463	Ping: 93.643 E2E: 264.408	Ping: 82.369 E2E: 236.551

Table 34. Event 2, Run 5—J2735 BSM—end-to-end latency results (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Avg: 88.875 Min: 61.811 Max: 388.316 Jitter: 58.495	Avg: 174.808 Min: 70.677 Max: 1,208.633 Jitter: 222.228	Avg: 163.641 Min: 85.459 Max: 1,223.602 Jitter: 183.741	Avg: 142.494 Min: 64.416 Max: 1,201.971 Jitter: 188.773
SRC: ANL (IL)	Avg: 292.886 Min: 55.713 Max: 2,765.839 Jitter: 55.577	N/A	Avg: 31.293 Min: 29.133 Max: 191.815 Jitter: 1.816	Avg: 46.938 Min: 44.928 Max: 207.451 Jitter: 1.796	Avg: 25.225 Min: 23.226 Max: 185.838 Jitter: 1.740
SRC: ORNL (TN)	Avg: 303.332 Min: 65.555 Max: 2,018.462 Jitter: 371.557	Avg: 35.917 Min: 29.154 Max: 182.598 Jitter: 8.105	N/A	Avg: 62.796 Min: 54.238 Max: 147.786 Jitter: 10.635	Avg: 39.362 Min: 32.286 Max: 90.926 Jitter: 7.089
SRC: UCLA (CA)	Avg: 323.056 Min: 77.937 Max: 2,626.33 Jitter: 392.271	Avg: 48.072 Min: 42.777 Max: 107.955 Jitter: 5.959	Avg: 58.106 Min: 51.922 Max: 104.122 Jitter: 7.976	N/A	Avg: 51.391 Min: 45.89 Max: 114.855 Jitter: 7.25
SRC: Mcity (MI)	Avg: 356.407 Min: 55.297 Max: 2,690.542 Jitter: 465.076	Avg: 37.518 Min: 20.172 Max: 388.128 Jitter: 18.476	Avg: 76.567 Min: 29.548 Max: 1,838.623 Jitter: 178.757	Avg: 92.548 Min: 45.169 Max: 1,853.901 Jitter: 180.827	N/A

Table 35. Event 2, Run 5—J2735 BSM—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: FHWA [CELL] (VA)	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	N/A	Ping: 91.987 E2E: 88.875	Ping: 103.442 E2E: 174.808	Ping: 116.152 E2E: 163.641	Ping: No Data (DCN8) E2E: 142.494
SRC: ANL (IL)	Ping: 84.433 E2E: 292.886	N/A	Ping: 33.154 E2E: 31.293	Ping: 48.630 E2E: 46.938	Ping: No Data (DCN8) E2E: 25.225
SRC: ORNL (TN)	Ping: 88.742 E2E: 303.332	Ping: 33.275 E2E: 35.917	N/A	Ping: 58.375 E2E: 62.796	Ping: No Data (DCN8) E2E: 39.362
SRC: UCLA (CA)	Ping: 142.479 E2E: 323.056	Ping: 47.544 E2E: 48.072	Ping: 59.611 E2E: 58.106	N/A	Ping: No Data (DCN8) E2E: 51.391
SRC: Mcity (MI)	Ping: No Data (DCN8) E2E: 356.407	Ping: No Data (DCN8) E2E: 37.518	Ping: No Data (DCN8) E2E: 76.567	Ping: No Data (DCN8) E2E: 92.548	N/A

Table 36. Event 2, Run 5—J2735 SPaT—end-to-end latency results (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CALIFORNIA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Avg: 306.318 Min: 59.704 Max: 3,444.194 Jitter: 572.229	Avg: 315.042 Min: 65.853 Max: 3,555.322 Jitter: 580.735	Avg: 345.45 Min: 81.217 Max: 3,687.742 Jitter: 603.194	Avg: 278.709 Min: 62.782 Max: 3,545.505 Jitter: 531.72

Table 37. Event 2, Run 5—J2735 SPaT—ping (one-way) versus end-to-end latency (ms).

SRC or DST Sites	DST: ANL (IL)	DST: ORNL (TN)	DST: UCLA (CA)	DST: Mcity (MI)
SRC: FHWA [CELL] (VA)	Ping: 91.987 E2E: 306.318	Ping: 103.442 E2E: 315.042	Ping: 116.152 E2E: 345.45	Ping: E2E: 278.709

E2E = end-to end.

DETAILED THROUGHPUT RESULTS

This section describes the detailed throughput results for vehicle SDOs, TL SDOs, and J2735 and the total throughput. Table 38 shows the average results for Event 0, table 39 shows the average results for Event 1, and table 40 shows the average results for Event 2.

Table 38. Event 0 average throughput results.

Site	Vehicle SDO (KB/s)	TL SDO (KB/s)	J2735 (KB/s)	Total (KB/s)
FHWA [FIBER] (VA)	UP: 8.878 DOWN: 23.540	UP: 0 DOWN: 12.610	UP: 7.552 DOWN: 24.146	UP: 16.430 DOWN: 60.297
ANL (IL)	UP: 0.003 DOWN: 37.912	UP: 0 DOWN: 17.948	UP: 2.305 DOWN: 32.762	UP: 2.308 DOWN: 88.622
ORNL (TN)	UP: 3.074 DOWN: 22.144	UP: 0 DOWN: 15.780	UP: 5.441 DOWN: 30.622	UP: 8.515 DOWN: 68.547
UCLA (CA)	UP: 7.873 DOWN: 38.817	UP: 0 DOWN: 8.697	UP: 4.005 DOWN: 27.339	UP: 11.878 DOWN: 74.853
Mcity (MI)	UP: 1.752 DOWN: 42.096	UP: 0 DOWN: 11.742	UP: 1.835 DOWN: 18.239	UP: 3.586 DOWN: 72.077
TTS (OR)	UP: 5.687 DOWN: 36.271	UP: 0 DOWN: 11.913	UP: 2.992 DOWN: 22.309	UP: 8.679 DOWN: 70.492
VW (CA)	UP: 3.704 DOWN: 51.879	UP: 0 DOWN: 3.871	UP: 0.677 DOWN: 28.595	UP: 4.381 DOWN: 84.344
Econolite (OH)	UP: 3.126 DOWN: 42.425	UP: 11.847 DOWN: 0	UP: 5.799 DOWN: 27.267	UP: 18.455 DOWN: 69.691

Table 39. Event 1 average throughput results.

Site	J3224 (KB/s)	TL SDO (KB/s)	J2735 (KB/s)	Total (KB/s)
FHWA [FIBER] (VA)	UP: 0 DOWN: 0.614	UP: 15.779 DOWN:	UP: 0 DOWN: 9.107	UP: 0 DOWN: 0.614
Mcity (MI)	UP: 0.649 DOWN: 0	UP: 0 DOWN: 15.730	UP: 9.078 DOWN:	UP: 0.649 DOWN: 0
TTS (OR)	UP: 0 DOWN: 0.638	UP: 0 DOWN: 15.588	UP: 0 DOWN: 9.009	UP: 0 DOWN: 0.638
VW (CA)	UP: 0 DOWN: 0.638	UP: 0 DOWN: 15.867	UP: 0 DOWN: 9.060	UP: 0 DOWN: 0.638

Table 40. Event 2 average throughput results (KB/s).

Site	Run 1	Run 2	Run 3	Run 4	Run 5
FHWA [CELL] (VA)	UP: 23.807 DOWN: 40.171	UP: 20.228 DOWN: 29.476	UP: 27.292 DOWN: 42.643	UP: 23.817 DOWN: 45.074	UP: 23.429 DOWN: 40.747
ANL (IL)	UP: 11.440 DOWN: 50.338	UP: 10.915 DOWN: 52.955	UP: 9.528 DOWN: 59.214	UP: 10.582 DOWN: 39.883	UP: 9.031 DOWN: 52.257
ORNL (TN)	UP: 9.922 DOWN: 59.408	UP: 8.005 DOWN: 50.874	UP: 7.801 DOWN: 61.516	UP: 6.203 DOWN: 26.756	UP: 6.449 DOWN: 22.221
UCLA (CA)	UP: 6.990 DOWN: 46.393	UP: 8.329 DOWN: 54.999	UP: 7.349 DOWN: 56.775	UP: 7.463 DOWN: 56.065	UP: 8.204 DOWN: 50.341
Mcity (MI)	UP: 9.922 DOWN: 40.812	UP: 18.996 DOWN: 40.312	UP: 18.345 DOWN: 43.296	UP: 17.263 DOWN: 45.968	UP: 16.919 DOWN: 38.954

COMMENTS

Figure 4 through figure 8 are based on maps provided by Mcity. FHWA modified figure 4 through figure 8 to show the placements of vehicles during test setup.

REFERENCES

1. U.S. Department of Transportation. n.d. “Virtual Open Innovation Collaborative Environment for Safety (VOICES) Proof of Concept (PoC)” (web page). https://www.transportation.gov/hasscoe/voices, last accessed February 24, 2026.

2. MITRE. n.d. “VOICES User Portal” (web page). https://portal.voices.mitre.org/, last accessed November 7, 2024.

3. FHWA. 2024. “Saxton Transportation Operations Laboratory” (web page). https://highways.dot.gov/research/laboratories/saxton-transportation-operations-laboratory/saxton-transportation-operations-laboratory-overview, last accessed February 24, 2026.

4. FHWA. 2024. *VOICES Cooperative Driving Automation (CDA) Proof-of-Concept Systems Integration Test 1*. Report No. FHWA-HRT-24-072.pdf. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-24-072.pdf, last accessed February 24, 2026.

5. FHWA. 2024. *Virtual Open Innovation Collaborative Environment for Safety (VOICES) Distributed Testing Pilot Test 1*. Report No. FHWA-HRT-24-159. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-24-159.pdf, last accessed February 24, 2026.

6. FHWA. 2024. *CARMA Platform* (software). Version 4.5.0

7. CARLA. n.d. “CARLA Open-Source Simulator for Autonomous Driving Research” (web page). http://carla.org/, last accessed November 7, 2024.

8. FHWA. 2022. “OpenCDA v0.1.2 Release” (software and configuration files in GitHub repository). https://github.com/ucla-mobility/OpenCDA/releases/tag/v0.1.2, last accessed November 7, 2024.

9. Mcity. 2023. *TeraSim* (software). Version 1.

10. Autoware Foundation. 2024. “Welcome to the Autoware Foundation” (web page). https://autoware.org/, last accessed November 7, 2024

11. Mcity. 2024. “Virtual Open Innovation Collaborative Environment for Safety (VOICES) Proof of Concept Presentation” (YouTube video). https://www.youtube.com/watch?v=KHRquc5HFYI, last accessed November 7, 2024.

12. Department of Defense. “Test Resource Management Center” (web page). https://www.trmc.osd.mil/, last accessed November 7, 2024.

13. SAE International. 2022. *V2X Communications Message Set Dictionary*, J2735_202211. 2022-11-14 revision. Warrendale, PA: SAE International. https://www.sae.org/standards/content/j2735_202211/, last accessed November 7, 2024.

14. SAE International. 2020. V2X *Sensor-Sharing for Cooperative and Automated Driving*, SAE J3224_202208. Warrendale, PA: SAE International. https://www.sae.org/standards/content/j3224_202208/, last accessed November 7, 2024.

15. Twingate. 2024. "Twingate" (web page). https://www.twingate.com/, last accessed November 7, 2024

16. National Institute of Standards and Technology. n.d. "Internet Protocol" (web site). https://csrc.nist.gov/glossary/term/internet_protocol, last accessed November 7, 2024.

17. FHWA. 2024. *CDASim* (software). Version 4.6.0.

18. FHWA. 2024. *CARMA Streets* (software). Version 4.5.1.

19. FHWA. 2024. *CARMA Cloud* (software). Version 4.6.1.

20. FHWA. 2024. *CARMA Messenger* (software). Version 4.8.0.

21. Eclipse MOSAIC. n.d. *Eclipse MOSAIC: A Multi-Domain and Multi-Scale Simulation Framework for Connected and Automated Mobility* (software). https://www.eclipse.dev/mosaic/, last accessed November 7, 2024.

22. TTS. n.d. "Personal Signal Assistant®" (web page). https://www.traffictechservices.com/personalsignalassistant.html, last accessed November 7, 2024.

23. ETSI. 2020. *ETSI TR 103 562 V2.1.1 (2019-12): ETSI ITS Intelligent Transport System (ITS); Vehicular Communications. Basic Set of Applications; Analysis of the Collective Perception Service (CPS); Release 2*. Sophia Antipolis, France: European Telecommunications Standards Institute. https://www.etsi.org/deliver/etsi_tr/103500_103599/103562/02.01.01_60/tr_103562v020101p.pdf, last accessed November 7, 2024.

24. Postel, J. 1980. "User Datagram Protocol" (web page). https://www.ietf.org/rfc/rfc768.txt, last accessed November 7, 2024.

25. Docker. 2024. "Docker: Accelerated Container Application Development" (web page). https://www.docker.com/, last accessed September 13, 2024.

26. Eclipse. 2023. *SUMO* (software). Version 1.17.0.

27. Chevy. 2021. "Chevrolet Grows EV Lineup With 2022 Bolt EUV and Bolt EV" (web page). https://news.chevrolet.com/public/us/en/chevrolet/newsroom.detail.html/Pages/ne

ws/us/en/2021/feb/0214-boltev-bolteuv-specifications.html, last accessed November 7, 2024.

28. ANL. n.d. ”Autonomie Vehicle System Simulation Tool” (web page). https://www.anl.gov/taps/autonomie-vehicle-system-simulation-tool, last accessed November 7, 2024

29. FHWA. 2024. *An Open-Source Tool for Interoperable Connectivity*. Publication No. FHWA-HRT-24-121. Washington, DC: Federal Highway Administration. https://highways.dot.gov/media/52571, last accessed February 24, 2026.

30. Postel, J. 1981. “Internet Control Message Protocol RFC 792 Part of STD 5” (web page). https://datatracker.ietf.org/doc/rfc792/, last accessed November 7, 2024.

31. National Institute of Standards and Technology. n.d. “Transmission Control Protocol” (web page). https://csrc.nist.gov/glossary/term/transmission_control_protocol, last accessed November 7, 2024.

32. Mcity. 2024. “Mcity 2.0” (web page). https://mcity.umich.edu/mcity-2-0/, last accessed November 7, 2024.

Recycled
Recyclable

Recommended citation: Federal Highway Administration, *Virtual Open Innovation Collaborative Environment for Safety (VOICES™) Distributed Testing Pilot Test 2* (Washington, DC: 2026) https://doi.org/10.21949/1521554

HRTM-30/03-26(WEB)E