Cooperative Driving Automation—Highway Driving Simulator Architecture: High-Level Architecture Design

May 2025

Publication No. FHWA-HRT-25-047

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

Recent advances in automated vehicles (AV) and connected automated vehicles (CAV) have increased the need to understand how drivers will adjust to and perform with these new vehicle systems in surrounding traffic and within the vehicles themselves. The Federal Highway Administration's (FHWA) Highway Driving Simulator (HDS) is a high-fidelity tool that can address human factors (HF) research needs related to emerging roadway and vehicle automation technologies. FHWA's Cooperative Driving Automation (CDA) Program focuses on researching and testing these emerging technologies, where vehicle-to-everything communications enable cooperative driving behaviors. The CDA Program develops and maintains the CARMA℠ Ecosystem, a suite of open-source software that includes automation technologies that allow infrastructure and vehicles to communicate and negotiate future actions with each other (FHWA 2022a).

This document presents a high-level software architecture for integrating CARMA's CDA capabilities with the HDS system. The high-level architecture is designed to enable the inclusion of information from infrastructure via CDA functionality into the HDS simulation environment that will mimic cooperative AV alerts to the driver. Coupling the HDS system with CDA functionality will lead to enhanced human-in-the-loop and software-in-the-loop simulation capabilities to support HF research needs related to CDA that are key to increasing the safety, effectiveness, and acceptance of emerging AV, CAV, and CDA technologies.

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

Recommended citation: Federal Highway Administration, *Cooperative Driving Automation—Highway Driving Simulator Architecture: High-Level Architecture Design* (Washington, DC: 2025) https://doi.org/10.21949/1521547

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. FHWA-HRT-25-047	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Cooperative Driving Automation—Highway Driving Simulator Architecture: High-Level Architecture Design		5. Report Date May 2025 6. Performing Organization Code HRSO-30
7. Author(s) John Stark, Yingyan Lou (ORCID: 0000-0001-9913-572X), Cheng Yuan, Ethan Slattery, Collin Nangle, David Gayman, Eric Chen, Kyle Rush (ORCID: 0000-0001-9769-9273)		8. Performing Organization Report No.
9. Performing Organization Name and Address Leidos Inc. 1750 Presidents St. Reston, VA 20190		10. Work Unit No. (TRAIS) 11. Contract or Grant No. 693JJ321D000010N / 693JJ322F00192N
12. Sponsoring Agency Name and Address U.S. Department of Transportation Federal Highway Administration 1200 New Jersey Avenue, SE Washington, DC 20590		13. Type of Report and Period Covered Final Report; August 2022–September 2024 14. Sponsoring Agency Code ITS/JPO and HRSO-30
15. Supplementary Notes: The contracting officer's technical representative was Brian Philips (HRSO-30; ORCID 0000-0002-8426-0867). The HDS Team and Sudhakar Nallamothu also contributed significantly to the creation of this document by providing edits and content material.		
16. Abstract Recent advances in automated vehicles (AV) and connected automated vehicles (CAV) have increased the need to understand how drivers will adjust to and perform with these new vehicle systems, both in surrounding traffic and in the vehicles they operate. The Federal Highway Administration's (FHWA) Highway Driving Simulator (HDS) is a valuable tool for human factors (HF) research; the HDS's capabilities can be expanded to address HF research needs related to emerging roadway and vehicle technologies. In addition, FHWA has spearheaded the Cooperative Driving Automation (CDA) Program. The CDA Program has led to a suite of open-source software tools called the CARMA℠ Ecosystem, which includes automation technologies that allow infrastructure and vehicles to communicate and negotiate future actions with each other (FHWA 2022a). This report presents a high-level software architecture for integrating the CARMA CDA capabilities with FHWA's HDS system. The high-level software architecture is designed to enable a variety of CDA activities and behaviors to be integrated into the HDS simulation environment and even put CDA technology into the subject cab vehicle when desired. The resulting integrated CDA-HDS system will lead to enhanced human-in-the-loop and software-in-the-loop simulation capabilities to support HF research needs related to CDA that are key to the safety, effectiveness, and acceptance of emerging AV, CAV, and CDA technologies.		
17. Key Words Driving simulation, vehicle automation, AV, connected automated vehicle, CAV, cooperative driving automation, CDA		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 63	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

3D	three dimensional
4K	resolution of 3,840 x 2,160 pixels
ADAS	advanced driver assistance system
ADS	automated driving system
AI	artificial intelligence
API	application programming interface
AR	augmented reality
ARCHER	Advanced Rendering Cluster for Highway Experimental Research
ASAM	Association for Standardization of Automation and Measuring Systems
AV	automated vehicle
CARLA	CAR Learning to Act
CAV	connected automated vehicle
CDA	cooperative driving automation
DDS™	Data Distribution Service
DoD	U.S. Department of Defense
FHWA	Federal Highway Administration
HDS	Highway Driving Simulator
HF	human factors
HIKER	Highly Immersive Kinematic Experimental Research
HLA	high-level architecture
HRSO	Office of Safety and Operations Research and Development
I/O	input/output
Hz	Hertz
IEEE	Institute of Electrical and Electronics Engineers
LiDAR	light detection and ranging
LVC	live, virtual, and constructive
MPI	message passing interface
NADS	National Advanced Driving Simulator
ns-3	Network Simulator 3
OEM	original equipment manufacturer
ORNL	Oak Ridge National Laboratory
ROS™	Robot Operating System™
SAE	SAE International
STOL	Saxton Transportation Operations Laboratory
SUMO™	Simulation of Urban MObility™
TENA	Test and Training Enabling Architecture
TFHRC	Turner-Fairbank Highway Research Center
TSMO	Transportation Systems Management and Operations
UoLDS	University of Leeds Driving Simulator
USDOT	U.S. Department of Transportation
USEP	unified simulation engineering platform
V2X	vehicle-to-everything

VOICES™	Virtual Open Innovation Collaborative Environment for Safety
VR	virtual reality
VRU	vulnerable road user
VTD™	Virtual Test Drive
XiL	everything-in-the-loop

CHAPTER 1. INTRODUCTION

DOCUMENT OVERVIEW

This document was developed under a Federal Highway Administration (FHWA) task order 693JJ322F00192N under contract number 693JJ321D000010 titled Cooperative Driving Automation (CDA)-Highway Driving Simulator (HDS) Architecture. Under the task order, the project team developed a system architecture that integrated the CARMA℠ Ecosystem[1] (FHWA 2022a)—a representative CDA implementation developed under FHWA's CDA Program—with the HDS ecosystem, software, and hardware, including the Advanced Rendering Cluster for Highway Experimental Research (ARCHER) (Williams et al. 2005).

Purpose

This report presents a high-level architecture for integrating the HDS with advanced driver assistance systems (ADAS), automated driving systems (ADS), and CDA technologies, as represented by the CARMA Ecosystem (FHWA 2022a). The researchers used the CARMA Ecosystem as the representative ADAS, ADS, and CDA implementation because the ecosystem is readily available and open source. Other mature ADS and CDA implementations are original equipment manufacturer (OEM) proprietary systems and finding information on them is difficult. The project team attempted to keep the interfaces general to support potential future integration with other CDA systems.

Audience

This report is intended for staff at FHWA, the National Highway Traffic Safety Administration, and the U.S. Department of Transportation (USDOT) who have an interest in the future of the HDS and how it may be used for human factors (HF) research, particularly related to CDA technologies associated with road safety and operations. This report documents existing HDS and CDA capabilities, describes the integrated CDA-HDS system requirements, and presents an approach for integrating HDS and CDA capabilities.

BACKGROUND AND MOTIVATION

A national perspective of transportation needs guides the work of FHWA's Office of Safety and Operations Research and Development (HRSO). The Saxton Transportation Operations Laboratory (STOL), established at the Turner-Fairbank Highway Research Center (TFHRC), conducts research into emerging technologies to help improve transportation mobility, efficiency, and safety. With recent advances in automated vehicles (AVs) and connected automated vehicles (CAVs), HRSO is responding to the increased need to understand how drivers will adjust to and perform with these new vehicle systems.

[1]See chapter 2 for more details.

Automated Driving and CDA

Technologies such as artificial intelligence (AI), computer vision, sensor fusion algorithms, vehicle motion planning, and automated actuation are cornerstones of automated driving. These self-contained, stand-alone technologies can enable various levels of assisted and automated driving with differing degrees of additional inputs from a human driver. CAV development by private industry in the United States has been focused on self-contained automated driving technologies. On the other hand, Government and research communities have advocated for CDA solutions where equipped vehicles work collaboratively and cooperatively with equipped infrastructure and other road entities to further the safety and mobility benefits of CAVs.

CDA allows equipped vehicles to communicate with each other, roadside equipment, and other road users via vehicle-to-everything (V2X) communication. SAE International® (SAE) has standardized how cooperation between vehicles is regarded. Similar to the SAE Levels of Driving Automation™ defined in SAE J3016™ (SAE 2021a), the new standard, SAE J3216™ (SAE 2021b), defines the classes of cooperation (Shi et al. 2020). Vehicles equipped with cooperative ADS can share their status and driving intent (classes A and B) and seek and enter cooperative driving agreements (classes C and D). Table 1 shows an overview of the relationship between cooperation classes and levels of vehicle automation.

Table 1. Overview of the relationship between Classes of Cooperative Driving Automation J3216 and Levels of Automation J3016 (SAE 2021b; SAE 2021a).

			Partial Automation of Dynamic Traffic Task		Complete Automation of Dynamic Traffic Task		
		SAE® Level 0	SAE® Level 1	SAE® Level 2	SAE® Level 3	SAE® Level 4	SAE® Level 5
		No driving automation (human does all driving)	Driver assistance (longitudinal OR lateral vehicle motion control)	Partial driving automation (longitudinal AND lateral vehicle motion control)	Conditional driving automation	High driving automation	Full driving automation
NO COOPERATIVE AUTOMATION		E.g., Signage, Traffic control devices	Relies on driver to complete the dynamic traffic task and to supervise feature performance in real time		Relies on automated driving systems to perform complete dynamic traffic tasks under defines conditions (fallback condition performance varies between levels)		
CDA CLASSES							
SAE® CLASS A STATUS SHARING	Here I am and what I see	E.g., Brake lights, traffic signals	Potential for improved object and event detection[a]		Potential for improved object and event detection[b]		
SAE® CLASS B INTENT SHARING	This is what I plan to do	E.g., Turn signal, merge	Potential for improved object and event detection[a]		Potential for improved object and event detection[b]		
SAE® CLASS C AGREEMENT SEEKING	Let's do this together	E.g., Hand signal, merge	Not Applicable		Cooperative automated driving systems designed to attain mutual goals through coordinated actions		
SAE® CLASS D PRESCRIPTIVE	I will do as directed	E.g., Hand signals, lane assignments by officials			Cooperative automated driving systems designed to accept and adhere to a command		

(a) Improved object and event detection and prediction through CDA Class A and Class B status and intent sharing may not always be realized given that Level 1 and 2 driving automation features may be overridden by the driver at any time, and otherwise have limited sensing capabilities compared to Level 3, 4, and 5 ADS-operated vehicles. (b) Class A and B communications are one of the many inputs to an ADS's object and event detection and prediction capability, which may not be improved by the CDA message.

In this document, the term ADS refers to automated, but not connected, vehicles (i.e., AVs). The term CDA-enabled vehicles refers to cooperative AVs that must be connected to facilitate that cooperation. Such connected AVs could also be called CAVs; however, this report uses slightly different emphasis between CDA and CAV. Although a CAV is connected, its connectivity might be nothing more than its infotainment system. Whereas, for the purpose of researching how AVs can cooperate with each other and infrastructure, the focus is on connectivity that has a material impact on the performance of the automated driving task. This document uses the term CDA to refer to infrastructure and vehicle systems where cooperative driving behaviors are enabled by connectivity.

The CARMA Ecosystem (FHWA 2022a) is a collection of open-source software for CDA research developed by STOL, which includes roadway infrastructure and vehicle technologies.[1] A variety of proof-of-concept CDA use cases have been prototyped into the CARMA Ecosystem, including the following:

- Cooperative perception (FHWA 2023a).
- Work zone navigation (FHWA 2023b).
- Advanced traffic signal optimization with CDA vehicle trajectory smoothing (FHWA 2023c).
- Speed harmonization, cooperative merging, and platooning (FHWA 2023d).

Thus far, a total of 14 CDA use cases have been developed in the CARMA Ecosystem, with several more under development (Balse et al. 2024).

To enable high-fidelity CAV and CDA simulation with the CARMA Ecosystem software products in the loop (FHWA 2022a), STOL has developed a cosimulation tool, CDASim (FHWA n.d.a). CDASim integrates CARMA Ecosystem software with leading industry simulation tools to simulate realistic CDA, transportation traffic, and communications data traffic in one package. CDASim uses the Eclipse® MOSAIC™ (Eclipse 2024) cosimulation framework and incorporates open-source simulation tools such as CARLA® (CARLA 2022), Simulation of Urban Mobility™ (SUMO™) (Eclipse 2021), and Network Simulator 3 (ns-3) (nsnam 2021–2024). The CDASim tool supports the development, evaluation, and deployment of CDA technologies.

Integrated CDA-HDS System to Enable Research

The HDS at TFHRC is a high-fidelity driving simulation tool for HF and other research. The HDS's capabilities can be expanded to address HF research needs related to emerging roadway and vehicle technologies. The HDS was redesigned in the early 2000s (Williams et al. 2005). The existing HDS systems include the cab vehicle, motion base, rendering and display equipment, commercial simulation software for representing the dynamic motion of the cab

[1]See chapter 2 for more details.

vehicle, and ARCHER software that manages all aspects of the simulated driver experience (Williams et al. 2005).

For the past two decades, the HDS has been used to examine a wide range of research topics in transportation safety and operations. These studies span from testing novel intersection and sign designs to examining the HF impacts of increased vehicle automation (Weaver, Chao, and Philips 2022). For the past 8 yr, researchers have used the HDS to conduct HF research studies with AV and CAV with SAE Level 1 and Level 2 automation, such as cooperative adaptive cruise control, lane-keeping assistance systems, and emergency braking assist systems (Weaver Chao, and Philips 2022). Approximating AV and CAV behaviors in these previous studies has been difficult and time consuming because sufficient information on system behavior and accurately generating such behavior has been unavailable.

Incorporating realistic ADS and CDA behaviors into the HDS would expand research capabilities. However, proprietary OEM ADS and CDA algorithms present hurdles for how to accurately simulate ADS and CDA behaviors in the HDS, as is required for meaningful research. Because the CARMA Ecosystem (FHWA 2022a) is open source, it can be integrated with HDS more easily than a system that is not open source. While the CARMA Ecosystem is a research-grade prototype implementation of ADS and CDA, and not an alternative to OEM ADS and CDA systems, integrating the CARMA Ecosystem with HDS will directly incorporate prototype ADS and CDA algorithms in the HDS simulation loop. This integration will allow more realistic ADS and CDA behaviors to be represented in HDS. The integrated CDA-HDS system will also enable easier verification testing and more rapid development cycles of CDA algorithms and systems. An integrated CDA-HDS system will lead to enhanced human-in-the-loop and software-in-the-loop simulation capabilities to support HF research needs related to CDA that are key to the safety, effectiveness, and acceptance of emerging AV, CAV, and CDA technologies.

Task Order Objective

The objective of this task order was to develop an architecture that defines a system structure and any common interfaces needed to integrate the HDS and the CARMA Ecosystem (FHWA 2022a). The overall goal was to develop an architecture for an integrated CDA-HDS system and to recommend updates to the HDS that will expand research capabilities to enable ADS and CDA software-in-the-loop and human-in-the-loop simulations.

The long-term goal for CDA-HDS integration is to incorporate CARMA Ecosystem software (FHWA 2022a) into the HDS and connect HDS to FHWA's Distributed Testing framework[1] (Loughran et al. 2024) to provide a tool that can evaluate the benefits and advance the development of CDA and CAV technologies.

[1]See chapter 2 for more details.

The focus of this project was on developing an architecture to enable the following capabilities:

- Simulate ADS vehicles as background traffic in a virtual world in the HDS.
- Simulate two or more CDA-enabled vehicles in a virtual world interacting with a human driver operating the HDS cab vehicle.
- Simulate CDA-enabled roadway infrastructure with one or more CDA-enabled vehicles in a virtual world interacting with a human driver operating the HDS cab vehicle.
- Simulate one or more CDA-enabled vehicles and CDA-enabled roadway infrastructure in a virtual world interacting with a CDA-enabled HDS cab vehicle.

OVERVIEW OF THE RECOMMENDED CDA-HDS HIGH-LEVEL ARCHITECTURE

A Unified Simulation Engineering Platform (USEP) Approach

To integrate ADS and CDA software into FHWA's HDS, a three-dimensional (3D) simulation platform that supports the development and validation of ADS and CDA systems is necessary. Key capabilities of such a simulation platform include, but are not limited to, the following:

- World representation consistent with ADS and CDA world models.
- Flexible application programming interfaces (APIs) that allow ADS and CDA software to control the behaviors of simulated vehicles.
- Simulation of ADS driving sensor suites.
- A relatively realistic physics simulation of the environment and vehicle dynamics.

The current version of ARCHER (Williams et al. 2005) has used simplified sensor models, scripting of SAE Levels 1–3 (SAE 2021a), and scripted automated behavior models. In STOL's cosimulation tool, CDASim (FHWA n.d.a), the CARLA open-source software (CARLA 2022) serves the role of such a simulation platform. In addition to CARLA, NVIDIA® DRIVE Sim™ (NVIDIA n.d.a) and rFpro (rFpro n.d.) are also well-known examples of simulation platforms for ADS development in the research community and in the industry (Malik, Khan, and El-Sayed 2022).

The project team initially considered two CDA-HDS integration approaches. Approach 1 considered integrating CARMA Ecosystem infrastructure and vehicle software into ARCHER (FHWA 2022a; Williams et al. 2005); and approach 2 considered integrating ARCHER components into CDASim (FHWA n.d.a).

The current version of ARCHER features a 3D simulation environment designed to simulate the driving experience from the perspective of a human driver or operator in the HDS cab vehicle (Williams et al. 2005). However, direct integration of the CDA infrastructure and vehicle software of the CARMA Ecosystem (FHWA 2022a) into ARCHER (i.e., approach 1) could require some rework of the ARCHER world model representation, physics engine, and dynamic objects simulation. Enhanced functionalities such as APIs for ADS and CDA software to control vehicle behaviors, ADS sensor simulation, and V2X communications would need to be developed.

Alternatively, relevant components of the ARCHER system (Williams et al. 2005) could be integrated into CDASim (FHWA n.d.a) (i.e., approach 2). More specifically, components such as the HDS cab vehicle control and dynamics modules could be integrated into CARLA (CARLA 2022) to control a simulated vehicle in the CARLA world representation. In this approach, the CARMA Ecosystem CDA software (FHWA 2022a) and the relevant ARCHER modules are both external software to CARLA, controlling relevant simulated dynamic entities in the CARLA world representation. One key challenge in directly integrating relevant ARCHER modules into CDASim is how to reconcile different world representations between ARCHER and CDASim. This challenge can be addressed by introducing a single master world model that holds the ground truth dynamic-state data for the roadway and all entities in the simulation (vehicles, pedestrians, signage, traffic signals, etc.). Another challenge in approach 2 is whether CARLA's support for 3D rendering is sufficient for simulating driving experiences in HDS. If CARLA's 3D-rendering support is not sufficient, other industry solutions (e.g., NVIDIA Drive Sim and rFpro may be explored (NVIDIA n.d.a; rFpro n.d.).

The project team recommends approach 2 for several reasons:

- Leveraging existing 3D simulation platforms designed to support ADS and CDA development and validation (e.g., the open-source CARLA (CARLA 2022)). instead of recreating such a simulation platform, could lead to more affordable and more rapid development than approach 1. If CARLA is deemed adequate for the 3D-rendering needs of HDS, then the existing integration of CARMA Ecosystem CDA software (FHWA 2022a) with CARLA could also be leveraged for the integrated CDA-HDS system, potentially further reducing the integration cost and schedule.

- Leveraging 3D ADS and CDA simulation platforms that are widely adopted in the industry would support the longer term goal of connecting HDS to FHWA's Distributed Testing framework (Loughran et al. 2024).

- Adopting a simulation engineering platform to host physics simulations would allow a delineation of subsystem responsibilities between vehicle behavior (motion and physical interactions) and the human experience (visual, motion, sound, and haptic experience). This delineation would allow subsystems to specialize in the aspects in which they are designed to excel. The delineation could also reduce development time for a variety of HF studies related to ADS and CDA since the need to design and develop custom models in ARCHER (Williams et al. 2005) to mimic desired ADS and CDA behaviors would be reduced. Such custom models will either be replaced by ADS and CDA software in the loop (if available) or developed more easily within the underlying 3D simulation engineering system using well-defined and documented interfaces. This delineation will also further modularize the HDS and allow specialized teams to focus on more streamlined subsystems.

The project team further recommends adopting a single master world model in approach 2 to reconcile different world simulations. More specifically, with the introduction of a 3D simulation platform designed to support ADS and CDA development and validation (e.g., the open-source CARLA (CARLA 2022)), the ARCHER software (Williams et al. 2005) could share the responsibility of simulating dynamic entities other than the HDS cab vehicle in the simulated world. Moreover, ARCHER's world representation is likely still needed for specific HF studies, such as rendering, data collection, and scenario control based on human subjects' actions. A single master world model that holds the ground truth dynamic-state data for all entities in the simulation would eliminate inconsistency in data across different representations. The 3D ADS and CDA simulation engineering platform of choice is the sensible choice to host this single master world model instead of hosting the master world model in the ARCHER modules that control the HDS cab vehicle or the CARMA Ecosystem CDA software (FHWA 2022a).

Since approach 2 features an existing 3D simulation platform for ADS development as the host for the world simulation and the host for the master world model—a single unified representation of the simulation world—approach 2 is referred to as the USEP approach. USEP is not meant to perform all computations needed to simulate the world. USEP serves as a host simulation for relevant ARCHER modules (Williams et al. 2005) that control the HDS cab vehicle, the CARMA Ecosystem CDA software (FHWA 2022a), and third-party software when needed, all of which perform their own calculations to control relevant entities in the world simulation. With a unified representation of the world model, movements of the HDS cab vehicle and other CDA vehicles are simulated in the same world.

Additional Architecture Considerations

While CARLA (CARLA 2022) is a potential USEP candidate for the integrated CDA-HDS system due to how it has been used in CDASim, its support for the 3D rendering needs of the HDS should be further evaluated. Additionally, a thorough investigation should be performed regarding real-time computation capabilities, time synchronization mechanisms, and map and scene data support for both the relevant ARCHER modules and the CARMA Ecosystem software (Williams et al. 2005; FHWA 2022a). These and other considerations specific to HDS are further discussed in chapter 4.

If CARLA (CARLA 2022) is deemed insufficient, other USEP candidates should be considered before shifting to adopt approach 1. Possible USEP candidates may include, but are not limited to, NVIDIA DRIVE Sim (NVIDIA n.d.a), rFpro (rFpro n.d.), dSPACE Automotive Simulation Models (dSPACE, n.d.), and IPG CarMaker (IPG, n.d.). However, the cost of commercial software products must be weighed carefully against their capabilities.

With the introduction of a USEP, the integrated CDA-HDS system would include three main components:

- USEP (e.g., CARLA (CARLA 2022)).
- ARCHER software (Williams et al. 2005).
- CARMA Ecosystem CDA software (FHWA 2022a).

Given the scale of the system, either a hub-and-spoke or a point-to-point architecture would be suitable. The rest of this report will use a hub-and-spoke architecture example.

USEP Summary

The recommended high-level architecture describes a system using a USEP to host the world simulation and to serve as the single source of ground truth of dynamic-state data. Attached to that USEP is the HDS and ARCHER ecosystem (Williams et al. 2005) (including commercial software for monitoring and controlling the HDS cab vehicle, motion base, and rendering hardware), as well as various CARMA systems (FHWA 2022a) that represent other world elements, such as ADS and CDA vehicles or connected infrastructure elements that make up the traffic environment. In the integrated system, the USEP should not perform all calculations needed for the world simulation. Relevant components in the integrated CDA-HDS can and will perform their own calculations. In this sense, the integrated CDA-HDS system is a connected distributed simulation. The USEP serves as a single-world model to ensure components' behaviors and interactions are all consistent and can be readily consumed by the ADS and CDA system.

The USEP will be extensible and capable of aligning with other relevant industry software. The introduction of USEP will also enable easier future development into connected simulation and distributed testing and support full-feature ADS and CDA behaviors while offering the opportunity to also perform computations for scenario entities that are not controlled by CARMA or ARCHER (FHWA 2022a; Williams et al. 2005).

DOCUMENT STRUCTURE

This report summarizes the motivations for the new architecture. The report also identifies the top-level subsystems and components in the new system, the responsibilities of the subsystems and components, whether the subsystems and components are new or reused, the interfaces among the top-level subsystems and components, and the system interfaces outside the HDS the USEP could support.

The remainder of this report is structured as follows:

- Chapter 2 details the existing HDS and CDA systems.
- Chapter 3 provides an overview of the landscape of driving and AV simulation.
- Chapter 4 discusses design considerations and requirements of the integrated CDA-HDS system.
- Chapter 5 describes the recommended architectural approach.
- Chapter 6 presents conclusions and recommendations.

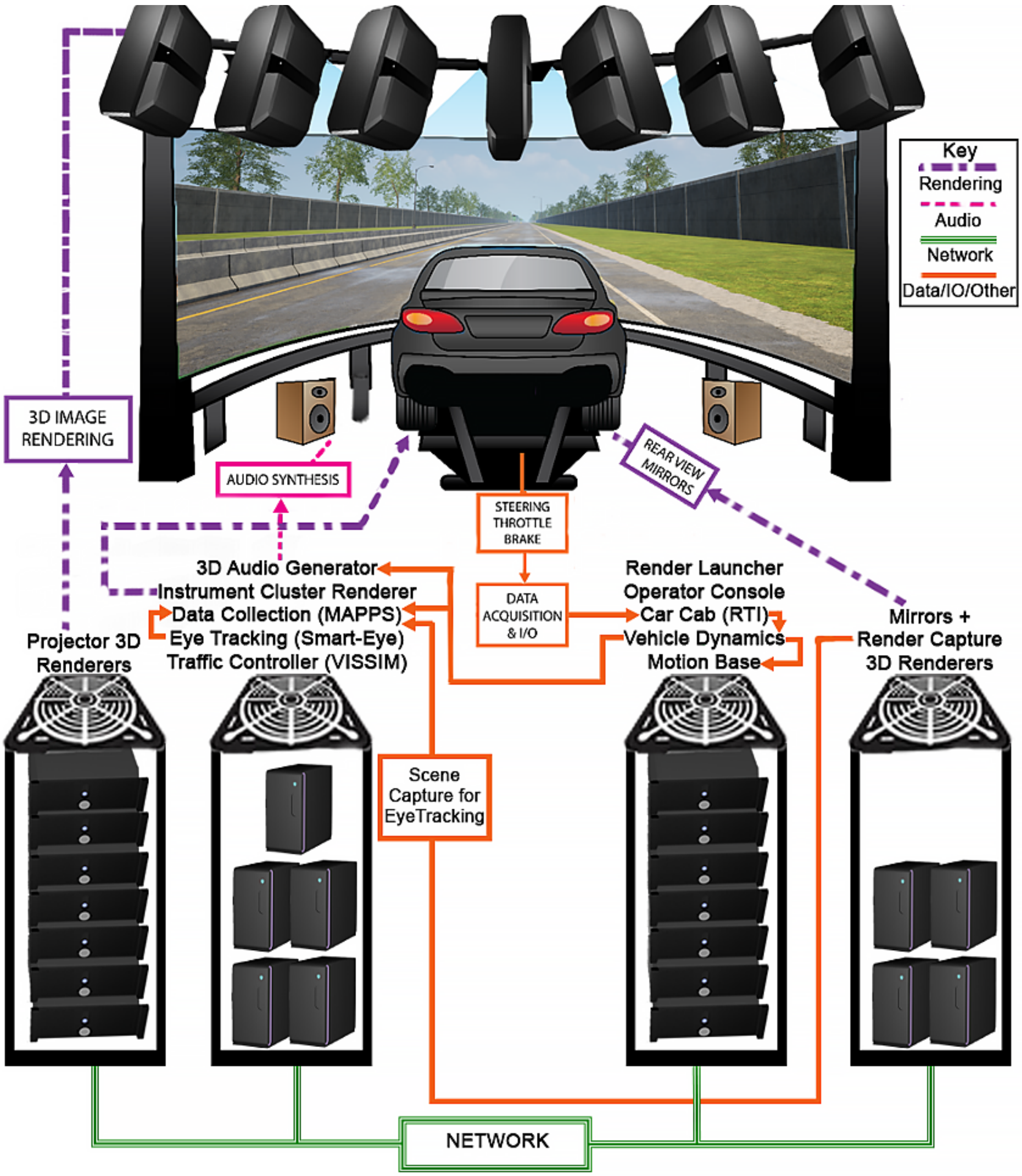
CHAPTER 2. CURRENT SYSTEMS

THE EXISTING HDS SYSTEM

The following section provides an overview of the HDS system and its major hardware and software components and processes.

HDS Background and Overview

The HDS provides an immersive driving experience in a safe, simulated environment in which the research focuses on driver reactions in various roadway situations. The HDS currently includes a modern passenger car cab with a full 6-degree-of-freedom motion base; seven video projectors with a resolution of 3,840 x 2,160 pixels (4K) that offer high-fidelity illumination; and fading and shadow-rendering capabilities, including weather, clouds, and water simulation. The HDS provides simulation of neighboring vehicles in the traffic stream and detailed simulation of the roadway and surrounding objects and structures. In addition to a visual and motion experience, research subjects also experience auditory and haptic sensory inputs. The HDS hardware system is shown in figure 1.



Source: FHWA.

I/O = input/output; MAPPS = Multimodal Analysis of Psychophysiological and Performance Signals; RTI = RealTime Technologies.

Note: References for the software mentioned in the figure are as follows: MAPPS (EyesDX 2009–2024); Motion Base (Moog n.d.); PTV Vissim® (PTV Group n.d.); RTI SimCreator (FAAC n.d.a); Smart Eye Pro (Smart Eye n.d.).

Figure 1. Illustration. HDS hardware system.

Current HDS Architecture

ARCHER (Williams et al. 2005) is the system (hardware and software) that drives the HDS. Figure 1 shows a high-level view of the ARCHER system. ARCHER runs on a cluster of computers, each with a specific role for generating the simulation. Working together, 3D-rendering computers generate the images produced on the seven projectors and three rearview mirrors and produce the instrument cluster display and map displays. Car cab controls are read by a data-acquisition and input/output (I/O) system, which is used by vehicle dynamics to update the participant vehicle's position. Values that the vehicle dynamics system calculates are then used for the motion base movement; the audio synthesis for engine, road, and wind sounds; and data collection. The eye-tracking system is also recorded along with video captures from the scene.

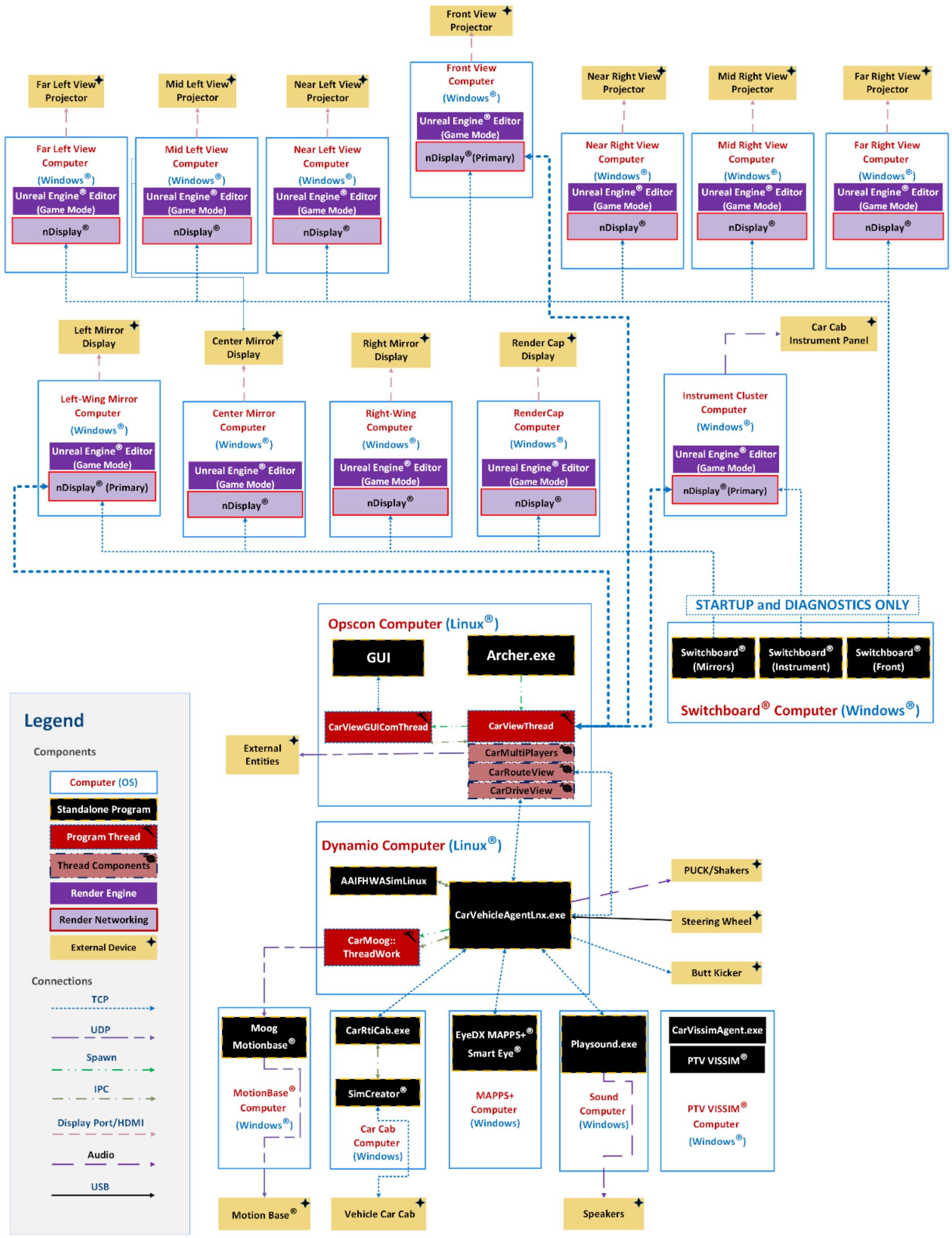
Four parallel main threads manage all of ARCHER's (Williams et al. 2005) activities (figure 2):

1. CarViewGUIComThread manages human–machine interface displays on the operator's console.

2. CarViewThread works with the nDisplay Unreal Real plug-in to assign different angles of view for the projector, rearview mirror, instrument cluster, and map views based on the cab vehicle's heading and position.

3. Traffic, pedestrians, bicycles, signals, and other dynamic objects are managed and updated within the ARCHER Unreal® Engine (Epic Games 2024) code and scripts (including the motion of all scenario vehicles and all scene elements).

4. Unreal Engine (Epic Games 2024) takes these updated states and renders the visual displays that form the human subject's impression of the world around them in every frame.



Source: FHWA.

GUI = graphical user interface; HDMI = High-Definition Multimedia Interface; IPC = interprocess communication; OS = operating system; TCP = Transmission Control Protocol; UDP = User Datagram Protocol; USB = Universal Serial Bus.
Note: References for the software mentioned in the figure are as follows: MAPPS (EyesDX 2009–2024); Motion Base (Moog n.d.); PTV Vissim® (PTV Group n.d.); RTI SimCreator (FAAC n.d.a); Smart Eye Pro (Smart Eye n.d.); Unreal Engine (Epic Games 2024).

Figure 2. Diagram. ARCHER II processes and threads.

CarVehicleAgent is responsible for the communications between various software modules that compute cab vehicle dynamics, drive the motion base, and drive the audio and tactile feedback that human subjects experience. CarVehicleAgent facilitates the passing of vehicle state variables required to generate the view updates. View state is passed from CarVehicleAgent through CarViewThread into the Unreal Engine (Epic Games 2024) assets rendering components. More specifically, CarVehicleAgent communicates hardware I/O between the motion base and physics engine and facilitates the passing of vehicle state variables required to generate the view updates into Unreal Engine nDisplay views.

The cab vehicle uses SimCreator software (FAAC n.d.a). The motion-based hardware and its software are from Moog Inc. (Moog n.d.). When a human subject operates the cab vehicle's steering wheel and pedals, the inputs are passed to SimVehicle (FAAC n.d.b), a component of the SimCreator software, for vehicle dynamics simulation. The outputs from SimVehicle are the updated vehicle state (location, orientation, velocity, and acceleration) that can be used to advance the states of the various sensory devices, including a new visual presentation of the changing environment.

Time Management and Synchronization

Currently, the HDS time synchronization is linked to the rendering frame rate that is more than 120 Hz. SimCreator (FAAC n.d.a) supports higher frequencies if needed. The HDS supports incorporating synchronous and asynchronous simulation components. Sampling at slower rates can also be supported for integration with simulations that have lower update rates (Perez, Bertola, and Philips 2016). Synchronous simulation is managed by the CarViewThread module that follows a typical centralized process where for each time step, all subprocesses must indicate that they have completed before the process can iterate to the next time step. Asynchronous processes can also be added (e.g., visualizations for the rearview mirror and sideview mirrors are asynchronous) but may lead to visualization discrepancies. If a subprocess takes longer than one-sixtieth of a second to complete, it will likely need to be implemented as an asynchronous process. The microscopic simulation PTV Vissim® (PTV Group n.d.) is an asynchronous process that is integrated with ARCHER (Williams et al. 2005). Since PTV Vissim simulates at 10 Hz, interpolation is implemented to estimate vehicle agents owned by PTV Vissim.

The HDS also supports a mode in which the frame rate is not fixed, and scenes are instead visualized as fast as possible. This alternative mode of operation supports the synchronous incorporation of subprocesses that cannot match the rendering frame rate.

Loading the Simulated World

The simulated world is represented as a collection of static and dynamic objects. The motion of many dynamic objects (e.g., neighboring vehicles, pedestrians) is managed by CarTrafficManager, an ARCHER submodule (Williams et al. 2005) of CarSceneManager (figure 2). One method of defining scene content is based on spatial organization and can be loaded and rendered in groups called tiles. Each tile typically represents anywhere from 1/2 to 2 mi around the cab vehicle's simulated location (tile size is based on the environment being studied; for highways, 2 mi works well, while in dense neighborhoods, 1/2 mi is used). As the cab vehicle moves through the scene, new tiles can be dynamically loaded into the renderer when needed without interrupting the HF experiment. This way, the HDS can simulate continuous driving for an indefinite distance through constantly varying scenery if tiles are present (Inman 2010). Tiles are one way of representing large scenes but are not a requirement. Many times, the entire world can be loaded in one chunk. The project team is still considering HDS designs for integrating multiple chunks with Unreal Engine (Epic Games 2024), which does not use tiles, per se.

Existing HDS External Interfaces

The current HDS architecture has an external interface to PTV Vissim (PTV Group n.d.). This PTV Vissim interface can be exploited to inject other vehicles as background traffic into the simulation. The PTV Vissim Driving Simulator API is currently used to integrate the cab vehicle into the PTV Vissim simulation. This API could also be extended to support a rudimentary simulation of one or multiple CDA-enabled vehicles as background traffic.

Cab Vehicle

The cab vehicle in the HDS can take manual inputs from a human driver to simulate driving. ARCHER (Williams et al. 2005) allows custom scripts to be developed and employed to prescribe the behaviors of the HDS cab vehicle in the simulated world. Past HF studies in the HDS have used customized scripts to emulate automated driving behaviors of SAE automation Levels 1–3 (SAE 2021a). However, no software is currently available to receive and apply external control commands from an ADS or CDA vehicle software stack that would convert the cab to an ADS or CDA vehicle with ADS or CDA vehicle software in the loop.

Existing Capabilities of ADS and CDA Behaviors

Some limited ability to simulate ADS and CDA behaviors exists through customized code within CarTrafficManager and through the PTV Vissim simulation (PTV Group n.d.). Such simulation is largely limited to scripted entity attributes, which can be updated on a per-time step basis. However, no actual CDA logic is incorporated. Other currently integrated third-party software (e.g., SimCreator (FAAC n.d.a)) also provides some CDA behavior capability, but bringing such capability into the HDS has not been explored.

PTV Vissim Behavioral ADS and CDA Driving Capabilities

A base level of ADS and CDA driving behavior can be simulated as background traffic using capabilities that exist in PTV Vissim 2020 (the currently integrated version) (PTV Group n.d.). These changes are parameterized to existing driving models that emulate AV behaviors and may not provide the level of realism that would likely be required for many HF research studies. The ADS and CDA capabilities in PTV Vissim include platooning, adjusted speed limit intersections, and a mix of different autonomous driving behaviors (cautious, normal, and aggressive).

SimCreator and SimDriver for Autonomous Cab Vehicle Control

The HDS currently uses the third-party software SimCreator (FAAC n.d.a) and SimDriver (FAAC n.d.c) to provide two functions: vehicle dynamics and car-cab I/O. SimCreator's autonomous vehicle control is contained in SimCreator's scenario controls. The HDS does not use these scenario controls, and therefore, SimCreator autonomous vehicle control cannot be used easily.

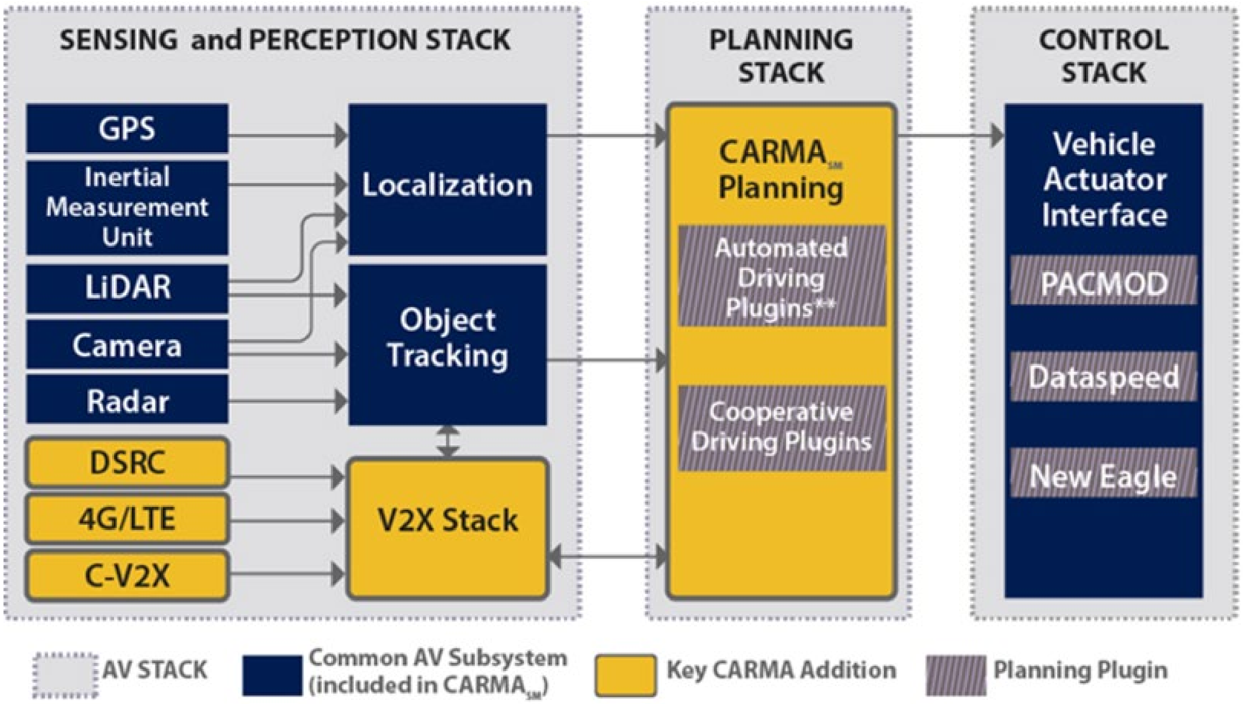
EXISTING CDA SYSTEMS

Since 2015, HRSO has led FHWA's CDA Program, formerly known as the CARMA Program (FHWA 2022a). The CDA Program focuses on transportation system improvements through research and testing for emerging automated driving and V2X communication technologies. Using V2X communications, CDA allows equipped vehicles to communicate with each other, with roadside equipment, and with other road users. SAE has standardized the classification of cooperation between roadway entities. Similar to the levels of automation defined in SAE J3016, the new standard, SAE J3216, defines the classes of cooperation (SAE 2021a, 2021b). CDA-enabled vehicles and infrastructure can share their status and driving intent (classes A and B, respectively) and seek and enter into cooperative agreements (classes C and D). Cooperation does not have to be between two vehicles; it could be between a signal controller and a bicyclist, between a traffic management center and a bus, or even among a group of several roadway entities of various types (SAE 2021b; Shi et al. 2020).

CARMA Ecosystem

The CDA Program has developed the CARMA Ecosystem, a suite of open-source software for CDA that includes vehicle and infrastructure technologies (FHWA 2022a). The CARMA Ecosystem currently consists of four products: CARMA Platform℠, CARMA Messenger℠, CARMA Streets℠, and CARMA Cloud℠ (FHWA n.d.b, 2023e, 2023f, 2023g). CARMA Platform is an ADS built on top of an open-source ADS platform for studying cooperative behaviors. In its fourth major iteration, the current CARMA Platform achieves conditional driving automation (at SAE Level 3) and enables ADS functionality to be used for cooperative automation strategies. Autoware®, and hence CARMA Platform, is built on a Robot Operating System™ (ROS) framework for intercomponent messaging (Autoware 2023; Open Robotics 2021).

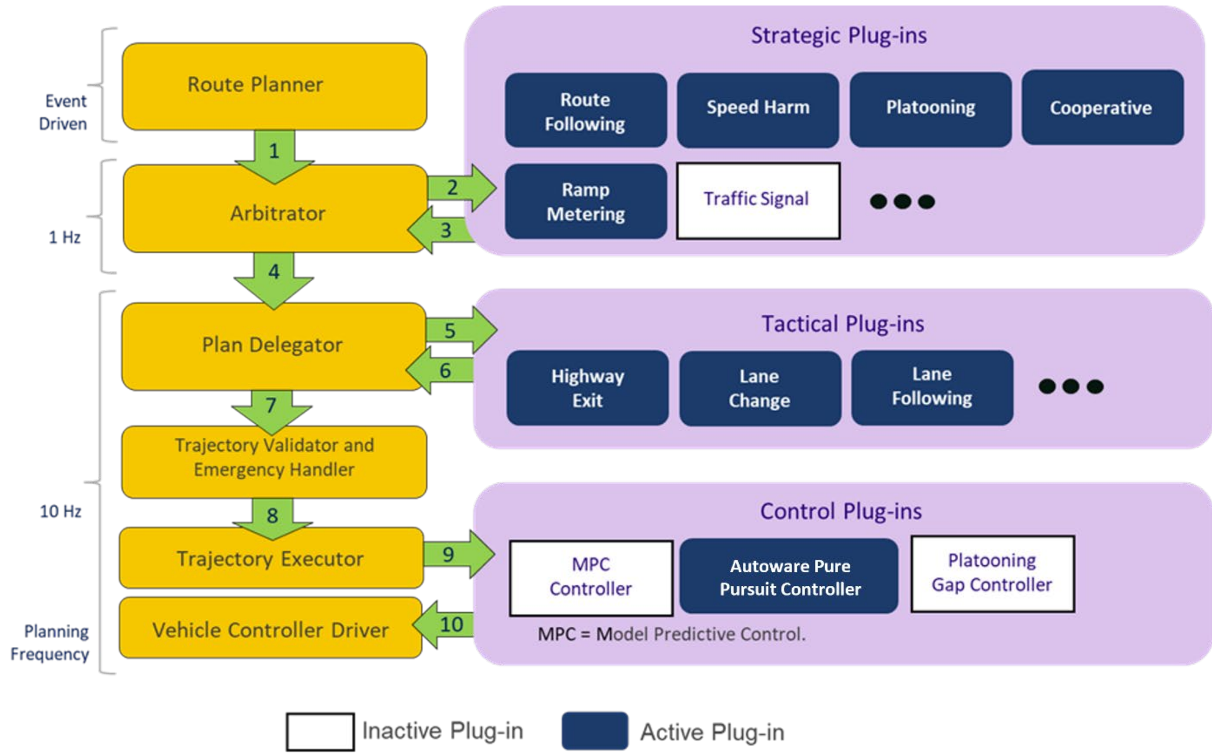
Figure 3 shows the CARMA Platform (FHWA n.d.b) stack model, which represents CARMA's software stacks. The sense stack is responsible for perception, localization, V2X communications, and the formation of a world model for interpreting the roadway, route, and external objects. The relevant data are passed to the plan stack, which consists of an arbitrated planning architecture (figure 4) that includes a set of strategic, tactical, and control plug-ins. The plan stack inputs a high-level route plan to a destination and converts this route into a sequence of maneuvers. The maneuvers are converted by the tactical plug-ins into a detailed trajectory plan. The trajectories are passed to CARMA's control plug-ins, which convert the trajectories into detailed steering and speed commands to be output to the act stack. The act stack is the interface with the vehicle controller module, which physically actuates the steering, accelerator, and brake pedal functions. The sense and planning stacks are the most important software stacks for integration into a separate simulation architecture. Simulated sensors and V2X components are required inputs to the sensing stack. Outputs from the plan stack can be leveraged in the simulation at multiple levels (i.e., detailed steering and speed commands from the control plug-ins can feed a vehicle dynamics model).



Source: FHWA.

*Supported by Autoware (Autoware 2023).
**Supported vehicle controllers: Dataspeed (Dataspeed Inc. 2024), PACMod (Open Robotics n.d.), and New Eagle (New Eagle 2024).
4G = fourth generation; DSRC = dedicated short-range communication; GPS = Global Positioning System; IMU = inertial measurement unit; LiDAR = light detection and ranging; LTE = Long Term Evolution; OBU = onboard unit.

Figure 3. Diagram. CARMA Platform stack model.



Source: FHWA.

Note: Users can choose which plug-ins to activate. This figure shows an example of both active (blue) and inactive (white) plug-ins. The three dots in the Strategic Plug-ins and Tactical Plug-ins boxes denote additional examples of those plug-ins.

Figure 4. Diagram. CARMA Platform arbitrated planning architecture and planning plug-ins.

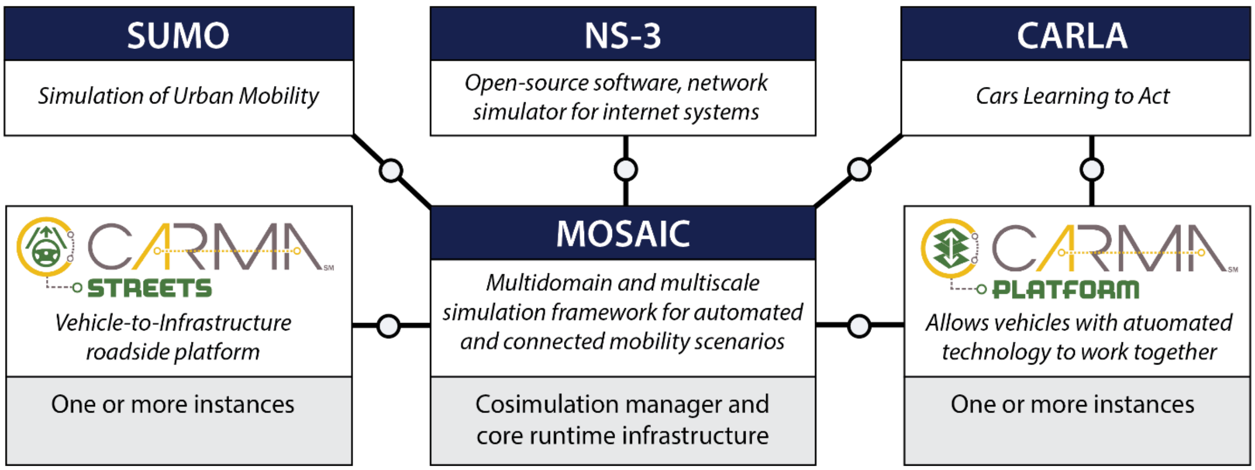
CARMA Messenger is also an in-vehicle software system that provides CDA connectivity in human-driven vehicles (FHWA n.d.b). In recent research, these vehicles represent first responders that are not automated but need to directly interact with CDA vehicles or infrastructure systems (e.g., to alert CDA vehicle traffic to the presence of first responders or to induce traffic signal override) (FHWA 2023h). CARMA Messenger could conceivably also be used for bicyclists, pedestrians, or other micromobility users.

CARMA Streets represents an infrastructure piece of CDA intended to be used for managing conflict areas (e.g., intersections) (FHWA 2023f). CARMA Streets provides an interface to roadside units, supports two-way communication between vehicles and infrastructure, and supports CDA by using edge computing to optimize travel through conflict areas. A part of CARMA Streets is the V2X Hub (FHWA 2023i), a set of software that facilitates data exchange needed in V2X communications by translating data elements into various standard protocols.

CARMA Cloud supports regional transportation systems management and operations through cloud-based management of transportation systems, data exchange, and multiple simultaneous remote services (FHWA 2023g). CARMA Cloud also hosts CARMA Analytics (FHWA 2022b), which supports the fusion, analysis, and management of data from CDA vehicles with traditional transportation data.

CDASim Cosimulation Tool

The CDASim cosimulation tool, shown in figure 5, was developed to provide an effective simulation method for testing CDA technologies related to operations and safety (FHWA n.d.a). The CDASim tool cosimulates a virtual driving environment (i.e., CARLA (CARLA 2022)), microscopic traffic simulation (i.e., SUMO (Eclipse 2021), and V2X communications (i.e., ns-3 (nsnam 2021–2024)) in a single cosimulation managed by MOSAIC (Eclipse 2024). CARLA supports ROS (Open Robotics 2021), which enables interfacing with CARMA Platform (FHWA 2023 n.d.b), with minimal changes to CARMA Platform's underlying architecture. CARMA Streets (FHWA 2023f) has also been integrated into the XiL architecture. This integration will allow for the simulation of CDA techniques that require vehicle-to-infrastructure communications.



Source: FHWA.

Figure 5. Diagram. CDASim architecture.

The CDASim cosimulation architecture (FHWA n.d.a) has a centralized time management and run-time infrastructure handled by MOSAIC (Eclipse 2024). The MOSAIC platform is inspired by the IEEE 1516-2010 standards for modeling and simulation high-level architecture (IEEE 2010). MOSAIC leverages this framework by wrapping each simulator into a federated object, which is linked to an ambassador that handles standardized connections with MOSAIC's run-time infrastructure. The run-time infrastructure handles the starting and stopping of simulators, data exchange between simulators, and time synchronization. MOSAIC's time management provides functionality to ensure that no simulators advance their time step until all simulators in the environment have processed the current time step.

FHWA's Distributed Testing Framework

FHWA's Distributed Testing framework (formerly known as the Virtual Open Innovation Collaborative Environment for Safety (VOICES™)) (Loughran et al. 2024) enables collaborative testing among participating entities (e.g., public sector, including State and local governments; private sector; and academic institutions) in a distributed virtual collaborative environment (FHWA 2024a). FHWA's Distributed Testing framework leverages Test and Training Enabling Architecture (TENA) as a common interface language used by all participants (U.S. Department of Defense (DoD) 2020, 2024). During the initial prototype work, the project team used TENA to interface between CDA vehicles and simulations hosted at geographically distributed entities. These entities interacted in realtime in a common testing environment that blended live, virtual, and constructive (LVC) simulations (DoD 2011).

Following DoD terminology, the LVC simulations are defined as follows:

- Live simulations refer to simulation instances with real roadway infrastructure, real physical vehicles, or other real roadway entities. The real physical vehicles can be operated by either human drivers or CDA systems.

- Virtual simulations refer to simulation instances with real human users or operators in a simulated travel environment.

- Constructive simulations refer to simulation instances with simulated vehicles or other road users operating in simulated environments following predefined driving logic.

TENA's ability to carry out distributed and blended LVC simulations is achieved through a common object model that enables semantic interoperability and a high-performance communication infrastructure (TENA middleware) for real-time data exchange (DoD 2020). Instances of TENA object models are called stateful distributed objects, and they carry data that describe relevant object attributes in the LVC simulations. FHWA's Distributed Testing framework (Loughran et al. 2024) uses TENA to connect multiple geographically distributed research and development sites to each other; the framework thus enables distributed testing of prototype ecosystems from various stakeholders to collaborate across their respective individual simulation environments. Each geographical site can have one or more nodes that represent a specific LVC simulation instance. All nodes across all sites are integrated into FHWA's Distributed Testing framework through TENA adapters and TENA middleware. The initial Distributed Testing prototype work leveraged the CARMA-CARLA cosimulation that is part of the CDASim cosimulation tool. Decoupling from the MOSAIC framework allows near real-time simulations of distributed entities (Eclipse 2024). The MOSAIC framework waits for all federate simulations to complete a time step before iterating to the next time step, making it difficult to guarantee real-time simulation. For the VOICES project (FHWA 2024a), distributed simulation systems are synchronized based on their system clocks (which may or may not be synchronized), and they do not have a shared time source. This design choice allows the simulation to run at near realtime but can lead to issues if different simulation entities significantly lag.

CHAPTER 3. DRIVING SIMULATION LANDSCAPE

GENERAL SIMULATION APPROACHES

Many organizations conduct transportation engineering research using simulation tools that can be classified into two categories: simulations and simulators. For this discussion, a simulation is a system (or system of systems) that automates the approximation of relevant actions and behaviors of roadway entities in the real world. A simulation could be all software or a combination of software and hardware. A simulator is a specialized type of simulation that has a human-in-the-loop and emphasizes studying the interactions of that human with the simulated world. FHWA's CDASim is an example of a simulation bringing together multiple systems, both software and hardware. In contrast, the HDS is an example of a real-time simulator that also brings together multiple systems, such as PTV Vissim (FHWA 2023j; PTV Group n.d.).

Simulators and simulation both have value for automotive driving research. A fully automated simulation can help researchers study inanimate interactions, such as vehicle dynamics, automated path planning, sensor perception, and traffic flow. Simulators are necessary to study how humans may react in various situations as a vehicle operator,[1] vehicle passenger, pedestrian, or other mobility user near a roadway. Combinations of both types of tools are often needed to provide a comprehensive picture of the interrelationships among traffic participants. For example, the Oak Ridge National Laboratory (ORNL) Real-Sim architecture is used to simulate cooperative merging across two physically simulated vehicles and for testing speed harmonization algorithms, all within a driving simulator architecture (ORNL 2022).

Simulators recreate sensory experiences of the surrounding world with the highest practical fidelity to make the human participants feel fully immersed in a realistic situation. These experiences include visual realism and auditory, motion, and haptic perceptions that make experiences more visceral. These sensory experiences may reflect scenes and scenarios that were built from recordings of actual driving performance, or they may be artificially generated scenes that appear realistic. The background traffic must often be realistic as well, to measure human reactions in a realistic traffic environment. In existing driving simulators, background traffic is generally managed by traffic microsimulation software, such as PTV Vissim, SUMO, and Aimsun® (PTV Group n.d.; Eclipse 2021; Aimsun n.d.; Solernou et al. 2020; Martin, Zlatkovic, and Tasic 2012; ORNL 2022). To maintain a realistic sensory experience for human drivers, the simulator must be low latency and time synchronized across the many different sensory outputs. An example is a simulator that uses simple network time protocol clients to allow simulation components to be synchronized within one-sixtieth of a second (Sawyer and Hancock 2012). While the artificial scenes may lack details and perceived authenticity, they allow researchers to work with a much wider variety of scenarios for a given budget.

[1]An operator is defined as a person sitting in the left front seat of a vehicle and is responsible for that vehicle's operation, whether the driver of a fully manual vehicle, a safety driver of a partially automated vehicle, or the passenger who turns on a vehicle's full automation capability and is responsible for monitoring and overriding the automation if necessary.

Design decisions by the simulator often require interoperability to ensure software and hardware components can be interchangeable. For example, ORNL's Real-Sim framework (ORNL 2022) provides a flexible interface that enables cosimulation of traffic, vehicle, and XiL components from a variety of different vendors and sources, with background traffic simulation supported by SUMO, PTV Vissim, and Aimsun (Eclipse 2021; PTV Group n.d.; Aimsun n.d.) and the virtual environment managed by CARLA, dSPACE, or IPG Automotive (CARLA 2022; dSPACE n.d., IPG n.d.). Similarly, the University of Leeds integrated an autonomous vehicle motion planner into its driving simulator using ROS, largely considered to be the standard used by autonomous vehicle developers (Solernou et al. 2020; Open Robotics 2021). A pure simulation may or may not be required to produce any of these sensory stimulations. If the research is studying sensor perception, then certain aspects of the scene will need to be modeled in detail. For example, to examine camera performance, a full visual scene depiction is required, but audio is not.

However, if sensor modeling is not part of the research, then objects such as vehicles can often be modeled as little more than points or boxes with appropriate dynamic characteristics. The University of Leeds Driving Simulator (UoLDS) is an example of a technically advanced simulator. UoLDS features a 5-m *x-y* motion table to emulate realistic vehicle motion. Within its hexapod dome, a 360° view is provided using six blended projectors, and the cab vehicle contains all relevant driver controls. UoLDS also has developed a pedestrian simulation architecture called Highly Immersive Kinematic Experimental Research (HIKER) (University of Leeds, n.d.). HIKER uses an array of 4K projectors in a 9- by 4-m walking space to provide a virtual reality environment that responds to participants' head position (using lightweight tracking glasses rather than a virtual reality (VR) headset). The University of Leeds can link UoLDS with HIKER to enable real-time interactions between pedestrians and drivers in a single virtual simulation environment.

The National Advanced Driving Simulator (NADS-1) is similar in structure to UoLDS, encapsulating the cab vehicle in a 24-ft dome that sits on a 64- by 64-ft *x-y* motion table to produce lateral and longitudinal accelerations. Four hydraulic actuators are attached to the cab vehicle to produce vibrations that emulate road feel. Visual simulation is achieved using 16 high-definition projectors that provide a 360° horizontal field of view and a 40° vertical field of view within the dome. NADS-1 can swap out different cab vehicles, such as full-sized cars, sport utility vehicles, trucks, and tractor-trailer cabs. Similar to UoLDS, NADS-1 can be linked with pedestrian simulators in a real-time connected simulation, as was demonstrated by FHWA's Exploratory Advanced Research Program (Kearney et al. 2022).

GENERAL APPROACHES TO SIMULATING ADS AND CDA IN DRIVING SIMULATORS

Integrating a driving simulator with AV simulation can increase the value of HF research related to AV behavior and areas related to AV behavior itself. For example, UoLDS was used to train an AI motion planner on real human driving experiences (in the simulator) (Solernou et al. 2020). The simulator was connected to an AV simulation that included the trained AI motion planner and demonstrated that the motion planner was able to realistically operate the cab vehicle through an evaluation scenario and to have human subjects act as passengers in this AV simulator to further verify the realism of the experience. ROS was used as the middleware to connect the driving simulator with the AV simulation, allowing for simpler integration of motion planning modules from different automotive OEMs or other groups developing autonomous vehicle platforms (Open Robotics 2021).

Similarly, NADS can interface with an AV at any automation level to study virtually any possible AV behavior without simulating its sensor inputs. In these experiments, human driver behavior and reactions to various driving situations (e.g., studying the timing required for human takeover requests) can be monitored. Not having an organic sensor simulation, NADS relies on the open-source CARLA simulation to provide this capability, as well as other open standards for operation (CARLA n.d.).

ORNL's Real-Sim driving simulator architecture uses an XiL approach to incorporate all components relevant to CAV driving simulation into the simulator (in either hardware or software form) (ORNL 2022). In demonstrations to date, ORNL has incorporated physical hardware of traffic signal controllers, V2X communication systems, CAV vehicle controllers, and vehicle powertrains into the simulation loop. From a software perspective, the Real-Sim framework maintains a flexible interface that enables the cosimulation of traffic, vehicle, and XiL components from a variety of different sources. For example, background traffic simulation can be managed by PTV Vissim, SUMO, or Aimsun, and the vehicle simulation environment can be managed by either CARLA, dSPACE, or IPG Automotive (PTV Group n.d.; Eclipse 2021; Aimsun n.d.; CARLA 2022; dSPACE n.d.; IPG n.d.). The Real-Sim architecture departs from many conventional driving simulators and does not incorporate a motion base for the cab vehicle. Instead, hub-coupled powertrain dynamometers are used to allow real vehicles to be inserted into the simulation, which can provide higher fidelity estimates of energy use. Since the inception of the Real-Sim framework, ORNL has been successful in demonstrating several complex use cases, including real-time CAVs, merging across two simulated vehicles, a signal controller in the loop, and a speed harmonization algorithm (ORNL 2022).

COMPREHENSIVE 3D SIMULATION ENVIRONMENTS AND PLATFORMS FOR ADS AND CDA DEVELOPMENT

Central to the integrated CDA-HDS architecture is a comprehensive world simulation that supports ADS and CDA development and validation. The world simulation needs to account for all vehicles and other objects of relevance, support the HDS cab vehicle operations, enable CARMA systems to be incorporated as software in the loop, accommodate traffic simulation, and meet the HDS rendering needs (FHWA 2024b, 2024c). The integrated CDA-HDS architecture will represent infrastructure elements and communications among various CARMA systems, as appropriate. To support the rendering necessary for human visual input and the various modes of input for any machine sensors involved, this simulation will need to provide realistic 3D models of all objects and surfaces in the vicinity.

Many existing driving simulators currently outsource vehicle dynamics and physics engines to third-party software services. For example, the Real-Sim framework (ORNL 2022) supports a variety of different vehicle dynamics and virtual simulation environments, including CARLA, dSPACE, VTD, and IPG Automotive (CARLA 2022; dSPACE n.d.; IPG n.d.). Similarly, a low-complexity driving simulator was built using a multirobot simulator software package, and many other commercial software sets have been used for vehicle dynamics, including CarSim®

(Mechanical Simulation Corporation 2024), Realtime Technologies (FAAC n.d.d), and CarMaker® (IPG n.d.; Swanson et al. 2013).

Three products—CARLA, NVIDIA DriveSim, and rFpro (CARLA 2022; NVIDIA n.d.a; rFpro n.d.)—are established simulation platforms used for ADS development (Malik, Khan, and El-Sayed 2022). These products can provide the comprehensive simulation capability needed by the integrated CDA-HDS system because they provide the following:

- Full range of features.
- Good vehicle dynamics models.
- Realistic dynamics models for other entities (e.g., pedestrians, cyclists).
- Traffic rules.
- Support for complex roadway situations, such as intersections and traffic signals.

The simulations also provide thorough 3D visuals with lighting and weather models and are well supported for future extensibility. CARLA is open source and built on a 3D simulation engine (Unreal Engine (Epic Games 2024)), while DriveSim and rFpro are commercial products (CARLA 2022; NVIDIA n.d.a; rFpro n.d.). The NVIDIA simulation system has excellent ties to the most common graphics hardware and the NVIDIA Drive systems (software and computing hardware). The rFpro system is a mature product that has been used for vehicle development purposes for more than 15 yr and has proven extensible.

CARLA

CARLA is a simulation designed for testing autonomous driving systems (CARLA 2022). CARLA's ability to coordinate other simulations and simulators that focus on different aspects of driving, such as traffic modeling or vehicle dynamics, makes it an ideal choice as the primary or host simulation in a multisimulation environment. Using CARLA as the host simulation makes integrating different simulations into a cohesive framework possible, which can lead to more realistic and accurate testing of autonomous driving systems. As part of the CDASim cosimulation tool (FHWA n.d.a), CARLA is used as the host simulation for CARMA Platform (FHWA n.d.b). Orchestrated by Eclipse MOSAIC (Eclipse 2024), other simulations, such as ns-3 (nsnam 2021–2024) and SUMO (Eclipse 2021), are also integrated into a cohesive framework with CARLA. The CDASim environment enables developers to test CDA systems in a realistic and accurate way. Simulation testing is a key aspect of developing and validating CDA systems, and CARLA's wide range of built-in sensors and modules, active development community, and ongoing research make it an effective tool for testing autonomous driving systems. CARLA's active development community and ongoing research mean it is constantly being updated with new features and capabilities, which can provide developers with access to the latest tools and techniques for testing and validating autonomous driving systems.

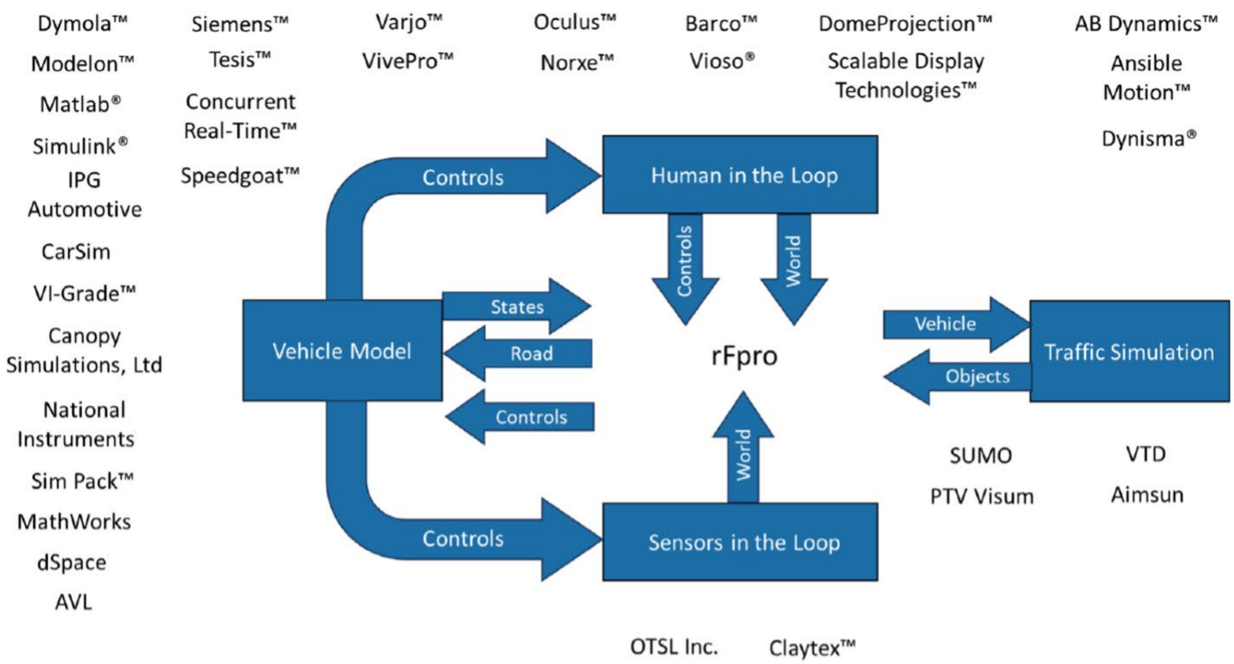
CARLA's open-source development model offers significant advantages over proprietary simulation products by enabling transparent development and widespread collaboration (CARLA 2022). This openness allows for rapid innovation, as developers from around the world can contribute to CARLA's features, keeping CARLA at the forefront of simulation technology. On the other hand, one potential issue with the open-source model is the variability in the quality of these contributions. While open-source projects benefit from diverse inputs, they can also suffer from inconsistencies unless a strong framework exists for quality control. In contrast, other proprietary products, where updates and feature integrations are tightly controlled and consistent, may lack the same level of community-driven innovation seen in open-source environments.

NVIDIA DRIVE Sim

NVIDIA DRIVE Sim is a simulation for the testing and development of autonomous vehicles (NVIDIA n.d.a). DRIVE Sim is built from NVIDIA Omniverse (NVIDIA n.d.b), which has open and extensible standard-building capabilities for 3D applications. DRIVE Sim provides a real-world-based simulation environment for developers to test their algorithms and control logic in a safe and controlled scenario. DRIVE Sim provides a physics-based simulation environment and a wide range of sensor models and other factors for the developer to customize. DRIVE Sim also comes with several built-in scenarios, which makes testing algorithms in different scenarios faster and easier. In addition, DRIVE Sim provides a variety of development tools to integrate with common third-party software and frameworks. DRIVE Sim has built-in facilities to support training for AI algorithms as well.

rFpro

rFpro is a simulation environment designed for vehicle development and testing as well as safety research (rFpro n.d.). rFpro is widely used by the automotive industry, with suppliers seeming to prefer this method for driving simulators used in either product development or safety research. Because of this preference, other companies have produced companion products for many years, and the ecosystem has become mature. A diagram of rFpro's association with external components is shown in figure 6.



Source: FHWA.

OSTL = Open Source Technology Lab.
Note: References for the products in the figure are as follows: AB Dynamics (AB Dynamics n.d.); Aimsun (Aimsun n.d.); Ansible Motion (Ansible Motion n.d.); AVL (AVL n.d.); Barco (Barco n.d.); Canopy Simulations, Ltd (Michelin n.d.); CarSim (Mechanical Simulation Corporation 2024); Claytex (Claytex n.d.); Concurrent Real-Time (Concurrent Real-Time n.d.); Domeprojection (Domeprojection n.d.); dSPACE (dSPACE n.d.); Dymola (Dassault Systemes n.d.); Dynisma (Dynisma n.d.); IPG Automotive (IPG n.d.); MathWorks (MathWorks 2024a); Matlab (MathWorks 2024b); Modelon (Modelon n.d.); National Instruments (National Instruments n.d.); Norxe (Norxe n.d.); Oculus (Meta 2024); Open Source Technology Lab (OSTL n.d.); PTV Vissim (PTV Group n.d.); rFpro (rFpro n.d.); Scalable Display Technologies (Scalable Display Technologies n.d.); Siemens (Siemens n.d.); Sim Pack (Dassault Systemes n.d.); Simulink (MathWorks 2024c); Speedgoat (Speedgoat n.d.); SUMO (Eclipse 2021); Tesis (Tesis n.d.); Varjo (Varjo n.d.); VI-Grade (VI-Grade n.d.); Vioso (Vioso n.d.);VivePro; VTD (Hexagon n.d.).

Figure 6. Illustration. An example commercial world model simulation approach.

In figure 6, the simulation environment is depicted centrally, while external, optional capabilities may be added to the simulation environment to either enhance it or as a component of the simulation. Because the simulation environment is built to be extensible, adding functionality does not require the simulation environment to be modified. Instead, an existing interface API is used to extend or improve the simulation based on the current use case or study.

CONNECTED AND DISTRIBUTED SIMULATION WORK

Linking geographically distributed simulation systems together is often desirable for running a more complex experiment to combine the depth of specialization of each system. Such distribution need not be cross-country; the contributors could be across a hallway in the same facility. A distributed simulation brings together disparate systems that can stand alone but complement each other's capabilities. Connected and distributed simulation capabilities are being implemented in many existing driving simulator architectures. UoLDS and Real-Sim can integrate multiple cab vehicles into a single simulation. Similarly, UoLDS and NADS both can connect with a separate pedestrian VR simulator. These capabilities, which allow multiple test subjects to interact in the same virtual environment, should be considered when defining the future extensible interfaces for the integrated CDA-HDS.

NADS has been connected to the Synchrono® system (Synchrono n.d.) to provide a high-fidelity, immersive human-driver experience in traffic that consists of many simulated AVs, where the AV simulation provides high-fidelity physics for vehicle dynamics and sensor observations. The key to making multiple AVs work in a high-fidelity simulation is fast communication of all relevant state data. Communications occur on a specialized network architecture using the high-performance Message Passing Interface (MPI) protocol, and MPI was successfully adapted to the lower reliability link to NADS (Benatti et al. 2022). NADS has been considered for connection to VR simulators of pedestrians, bicyclists, and possibly other vulnerable road users (VRUs) at universities. Such a connected simulation would allow research into cataloging typical behaviors of these VRUs and studying the reactions of drivers to various VRU situations.[2]

The Real-Sim cosimulation environment connects several simulations of background traffic, detailed vehicles, infrastructure controllers, and V2X communication systems into an XiL simulation system that can support a variety of experiments (ORNL 2022). The Real-Sim cosimulation environment can optionally use real hardware for integrating physical vehicle controllers, signal controllers, communication systems, and sensor drivers, allowing for simpler reuse of software between real-world ADS and CDA vehicles and the Real-Sim simulator cab vehicle. As part of this effort, the Real-Twin system has been developed that partially automates the process of generating digital twin scenarios and environment data in a way that is standardized and portable so that its scenarios are reusable (ORNL 2022).

Both MPI and the Data Distribution Service (DDS™) are accepted industry-standard protocols for efficient, high-frequency, tailorable, publish-and-subscribe messaging in large simulation environments (Argonne National Laboratory n.d.; Project Chrono 2024). MPI is a robust message-passing standard capable of handling multiple program setups and is widely used in supercomputing clusters, while DDS allows users to configure very low-level details of the service to exactly fit the configuration requirements. DDS is the standard protocol used by ROS (Open Robotics 2021), which is the underlying framework for the CARMA Platform (FHWA n.d.b) and CARMA Messenger (FHWA 2023e) CDA vehicle systems.

At STOL, the VOICES proof-of-concept project has contributed to the field of distributed simulation (FHWA 2024a). Through the VOICES project, USDOT intends to seed a distributed testing platform designed to facilitate collaborative research, development, and testing among diverse stakeholders engaged in transportation automation and connectivity technologies.[3] FHWA's Distributed Testing framework (Loughran et al. 2024) is built on DoD technology for widely distributed LVC test environments, referred to as TENA, as the backbone of its distributed synthetic test environment (DoD 2020). The flexible Distributed Testing platform

[2]University of North Carolina at Charlotte, University of Virginia, and University of Illinois. 2022. "Distributed Multi-Agent Virtual Reality Simulator for Road User Safety" (unpublished technical memo).

[3]USDOT. 2020. "Virtual Open Innovation Collaborative Environment for Safety." Predecisional draft – not for distribution.

allows collaboration among live agents (e.g., human drivers, pedestrians, or real-world traffic signal controllers), virtual agents (e.g., humans in a VR headset), and constructive systems (e.g., full software simulations of vehicles or infrastructure components). In the 1990s, the DoD developed its high-level architecture, a standard for distributed simulation that built on its previous successes with the Distributed Interactive Simulation standard (DoD 2024). The high-level architecture has been incorporated into an IEEE standard and is designed to enable interoperability and reuse (IEEE 2010). The high-level architecture defines individual simulations as federates and defines a standardized run-time infrastructure that provides services for information exchange, time synchronization, and federated management.

CHAPTER 4. DESIGN CONSIDERATIONS AND SYSTEM REQUIREMENTS

The long-term goal for CDA-HDS integration is to incorporate CARMA Ecosystem software (FHWA 2022a) into the HDS and connect HDS to the FHWA Distributed Testing framework (Loughran et al. 2024) to provide a tool that can evaluate the benefits and advance CDA and CAV technologies.

The vision for the integrated CDA-HDS system is as follows:

- Enable the HDS cab vehicle (and its associated hardware and software, including ARCHER (Williams et al. 2005)) to participate in CDA applications with other vehicles and with infrastructure. These applications would be used under the following circumstances:
  - Human driving.
  - Automated driving while the human subject in the vehicle is merely a passenger.
  - Automated driving, where the human subject is allowed to take over control as needed.
- Enable the HDS vehicle (which may be equipped with CARMA Platform or CARMA Messenger (FHWA n.d.b, 2023e) to send and receive messages from other vehicles (which may be equipped with CARMA Platform or CARMA Messenger), infrastructure (using CARMA Streets (FHWA 2023f)), and the cloud (using CARMA Cloud (FHWA 2023g)).
- Enable researchers to study how a human driver reacts to cooperation strategies involving the following:
  - Other entities (e.g., other vehicles and infrastructure), but not the driver.
  - The HDS vehicle (i.e., the vehicle driven by the human driver) and other entities.
- Enable researchers to study how a human passenger reacts to cooperation strategies (including possibly taking over manual control of the vehicle) involving the following:
  - Other entities (e.g., other vehicles and infrastructure), but not the driver.
  - The HDS vehicle (i.e., the vehicle driven by the human driver) and other entities.
- Enable the use of CDA strategies and features previously developed and tested in CARMA (FHWA 2022a) both by the HDS vehicle and other surrounding vehicles or infrastructure.
- Enable the implementation of future CDA strategies and features into the HDS (both for the HDS vehicle and other surrounding vehicles) that will be developed in the future (including by external CARMA partners).

- Allow for easier collaboration of CAV and AV between studies that rely either on the HDS or CARMA onroad testing (FHWA 2022a).
- Provide opportunities for combining onroad and simulation-based testing scenarios.
- Provide a system where HF testing can be evaluated through simulation in the HDS.

The following example scenarios may be employed in the integrated CDA-HDS system to explore potential research questions:

- Drive a manual vehicle (Level 0 (SAE 2021a)) in the HDS and interact with other CDA-enabled entities (vehicles or infrastructure), including mixed-fleet interactions. This research would include scenarios with just physical interactions (i.e., no cooperation capabilities) and both physical interactions and wireless communication interactions (i.e., an HDS vehicle equipped with CARMA Messenger (FHWA 2023e)).
- Drive a Level 1–Level 5 (SAE 2021a) CAV (i.e., a vehicle equipped with CARMA Platform (FHWA n.d.b)) in the HDS and interact with Level 0–Level 5 CAVs in a mixed fleet (i.e., CARMA Messenger (FHWA 2023e) and CARMA Platform-equipped vehicles).
- Drive a CARMA vehicle(s) in the field (i.e., either a public road or a closed test track) and bring in the data from the drive to replay in the HDS.
- Drive a Level 1–Level 5 (SAE 2021a) automated vehicle in the HDS and interact with CARMA vehicle(s) in the field (hardware-in-the-loop scenario).
- Conduct HF testing of simulated CDA concepts, which can be driven in the HDS, before testing in the field with vehicles and infrastructure equipped with CARMA software (FHWA 2022a).

DESIRED CAPABILITIES

The following four capabilities are desired for the integrated CDA-HDS system:

1. Simulate ADS vehicles as background traffic in a virtual world in the HDS.
2. Simulate two or more CDA-enabled vehicles in a virtual world with a human driver operating the HDS cab vehicle.
3. Simulate CDA-enabled roadway infrastructure with one or more CDA-enabled vehicles in a virtual world with a human driver operating the HDS cab vehicle.
4. Simulate one or more CDA-enabled vehicles and CDA-enabled roadway infrastructure in a virtual world with a CDA-enabled HDS cab vehicle.

Capability 1 is the simplest, only requiring the integration of a simulated ADS vehicle into the HDS virtual world as background traffic. This capability would be similar to integrating CARMA Platform's (FHWA n.d.b) driving logic into a virtual world without any need for vehicle-to-infrastructure or vehicle-to-vehicle messaging, which has been done previously with the CDASim tool and the VOICES project (FHWA 2024a).

Capability 2 builds on capability 1, with the goal of simulating two or more CDA-enabled vehicles as scenario vehicles in the virtual world with a human driver operating the HDS cab vehicle. This scenario imposes new requirements since CDA-enabled vehicles rely on communication between those vehicles to enable cooperation. Interaction with a human driver means the simulated CDA vehicles need to react appropriately to the human-driven cab vehicle, and the cab vehicle potentially can display V2X communications from the surrounding CDA vehicles to the driver.

Capability 3 incrementally expands on desired capability 2, adding the ability to simulate CDA-enabled roadway infrastructure, which requires integrating CARMA Streets or CARMA Cloud services into the HDS virtual world (FHWA 2023f, 2023g). Capability 3 also requires that the infrastructure can communicate with CDA-enabled background traffic. Capability 3 does not specifically require communications between the cab vehicle and infrastructure, but CDA-initiated changes to the infrastructure (e.g., traffic signals) need to be rendered and visualized in the HDS so that the human driver can react to them.

Capability 4 builds on desired capabilities 1–3, requiring simulation and communication between CDA vehicles for background traffic, CDA-enabled roadway infrastructure, and a CDA-enabled cab vehicle. The primary difference between capability 3 and capability 4 is the goal of controlling the cab vehicle with CARMA Platform logic (FHWA n.d.b). Because capability 4 will physically control the cab vehicle, safety considerations of the human driver and the limitations of the motion base will be required.

Given that the primary purpose of the HDS is for HF research, detailed modeling of the V2X communications processes at a physical layer is considered a lower priority. Instead, V2X communications will be simulated at the application layer. Simulations at the application layer will reduce computational complexity and allow more flexibility in modeling scenarios in which the communications malfunction (e.g., a communication delay could be injected).

OPERATIONAL NEEDS OF THE INTEGRATED CDA-HDS SYSTEM

The primary driver for the architecture is the desire to study HF in an environment with CDA interactions. Since the CARMA Ecosystem (FHWA 2022a) already provides a wealth of this capability, integrating the ecosystem with the HDS to enable such HF studies is logical. Doing so is much more efficient in terms of expenditures for study setup and execution, given the rich variety of studies with CDA involvement that will be possible. This integrated system could also support research on CDA topics by validating that certain previously performed use cases are practical from the HF perspective.

The following possible HF research questions could be supported by this new capability:

- How do drivers react to seeing various ADSs nearby?
- How do drivers react to a platoon of CDA vehicles nearby?
- How do drivers react to a CDA vehicle performing a cooperative lane merge?
- How do drivers react when engulfed in a CDA-dominated speed harmonization zone?
- How do drivers react to traffic signals that are dynamically controlled due to CDA vehicle interactions?
- How does the operator of a CDA vehicle of varying automation levels react when involved in a variety of cooperative activities with other CDA vehicles, such as a platoon or other use cases previously studied as part of the CDA Program?
- How does the operator of a CDA vehicle of varying automation levels react when interacting with a CDA-enabled VRU?
- How does the operator of a CDA vehicle of varying automation levels react when acting on some critical shared-perception data to evade a dangerous situation that is not obvious to the operator?

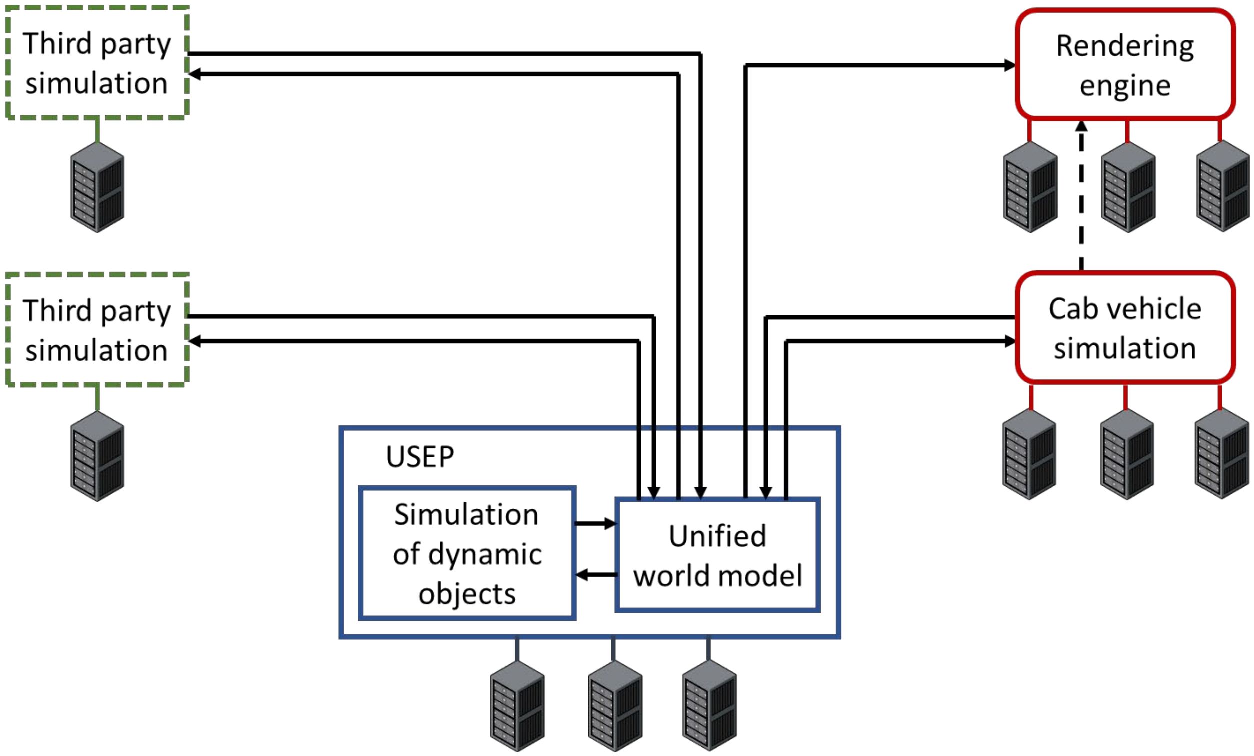
Several of these questions involve the cab vehicle having CDA capabilities and allowing the driver to override the automation and take back control of the vehicle. Also, studies of such questions will inherently involve the capability to simulate V2X message passing among CDA vehicles and infrastructure components. In some studies, the cab vehicle may be given the ability to display some subset of information available through V2X communications, regardless of whether it is being automated for that study or not. Further, in any given scenario, the level of automation or the class of cooperation available to the participating vehicles may need to be varied to study how these gradations affect driver performance or confidence. Additionally, future integration of this CDA-HDS system with live entities (vehicles or infrastructure components) that can interact with virtual entities in the simulated scenario is desirable. This CDA-HDS system integration would be similar to the LVC simulation capability currently supported through FHWA's Distributed Testing framework (Loughran et al. 2024).

The current implementation of CARMA Platform (version 4.3.0) (FHWA n.d.b) has a human–machine interface that is designed for engineers who are doing indepth research on CARMA's functionality and performance. CARMA Platform is not designed for consumer-grade users to feel comfortable with the system. Any research that requires such a user interface will probably have to build the interface from scratch for that project; defining such a user interface is outside the scope of this architecture.

USEP CONSIDERATIONS AND REQUIREMENTS

All simulated road user entities need to operate in sync with each other, and they all must have an opportunity to observe the same state of the world at any moment (according to their individual sensory capabilities). To achieve this consistency across all entities, the integrated CDA-HDS system will be centered on a single world model hosted by a single simulation engineering platform called USEP. This single world model will represent the ground truth dynamic state data for the roadway and all entities in the simulation (vehicles, pedestrians, signage, traffic signals, etc.) and will provide it to other systems in the simulation as necessary. The USEP will be a simulation platform unto itself, providing capabilities of 3D world simulation needed for ADS and CDA software to operate. The USEP would normally provide the primary simulation engine for the ADS and CDA vehicles. However, any given roadway entity or vehicle could be simulated externally (e.g., the cab vehicle) by third-party models or simulations as plug-ins to the USEP. In this case, the third-party simulation would read from the USEP the relevant world and object states that are needed for its own calculation and then simply pass updated state information of the entity it is responsible for back to the USEP. With all the world-state data served from one place, serving those data to all state consumers, such as the HDS rendering engine, is straightforward. This structure, shown in figure 7, supports the separation of responsibilities, maximum cohesion of purpose, and minimal coupling between components.

In figure 7, arrows represent the flow of data around the state of one or more world entities. The dashed arrow is an optional data flow. This structure separates the world-state data, the generation of that data (simulation), and the use of that data (rendering). All simulations in use will feed their state updates to the unified world model, and then all pieces of the system can subscribe to that world model to get a uniform set of complete state data at any time. An optional data flow goes directly from the cab vehicle simulation to the rendering engine, which is a legacy connection, and may be necessary to meet performance constraints; however, those same data also will flow into the USEP to form a complete picture of the world there. In figure 7, the motion simulation within the USEP would nominally represent the ADS and CDA vehicles participating in a scenario. However, the USEP could simulate more than the ADS and CDA vehicles if appropriate.



Source: FHWA.

Note: Red boxes with rounded corners represent legacy HDS components. Blue boxes with square corners represent the new USEP components. The dashed arrow represents an optional data flow.

Figure 7. Diagram. The USEP data flows and distribution of responsibilities.

The project team considered an alternative approach to the USEP, which was to modify the existing ARCHER and PTV Vissim components to inject CDA behaviors into the HDS (Williams et al. 2005; PTV Group n.d.). The Vissim driving simulator and traffic signal APIs provide hooks to integrate custom behavior into vehicle and signal entities in the HDS background traffic. However, doing so may require multiple world simulations with different physics models and limits the extensibility of the system. More specifically, PTV Vissim is a traffic flow simulation that has 3D-modeling support, but it does not lend itself well as a world model or world simulation for ADS and CDA software because it lacks key capabilities such as interfaces for ADS and CDA software, physics engines, vehicle dynamics, and 3D-sensor simulation.

The project team also considered using SimDriver (FAAC n.d.c) for integrating capability 4 with minimal changes to existing ARCHER (Williams et al. 2005) code, since ARCHER already uses SimCreator (FAAC n.d.a) for its vehicle dynamics. The available SimDriver could convert the cab into an ADS. SimCreator works well in SimCreator-defined scenarios (which the HDS does not plan to use) but often requires modification when working with custom scenarios. Therefore, SimCreator's flexibility is insufficient for the anticipated uses.

ENABLING THE CAB VEHICLE FOR ADS AND CDA

In addition to using CARMA Platform (FHWA n.d.b) to control some of the vehicles in the background traffic for an HF experiment using the HDS, the project team also wanted CARMA Platform to be able to control the cab vehicle, per capability 4. Doing so would make the cab vehicle an ADS and would give it optional communication capabilities to make it CDA-enabled as well.

For CARMA Platform (FHWA n.d.b) to control the cab vehicle, a new software interface between CARMA Platform and ARCHER (more specifically, the CarVehicleAgentLnx module) will be needed (Williams et al. 2005). Through this new interface, the commands coming from CARMA Platform (speed and steering) will be fed into CarVehicleAgentLnx and then to SimVehicle for vehicle dynamics calculation. The cab vehicle currently includes actuators to translate the incoming commands into the physical motion of the steering wheel. Optionally, new actuators could be considered for the motion of the brake and accelerator pedals to reflect the control actions that would happen in a real ADS vehicle as it drives. (Some vehicles may opt to not provide physical motion of the pedals if they are drive-by-wire, and the speed-related commands can be directly fed into the controller area network bus; this design choice is based partly on the desire for driver experience.) This data stream can also be routed to the cab vehicle's motion platform, if necessary. This new interface to the USEP could be refactored from those already present in the CDASim and will convert CARMA Platform's ROS data streams to the appropriate ingest format (Open Robotics 2021). A new component will be needed for translating these ROS streams into commands for the existing hardware API for the vehicle control actuators.

CARMA Platform is designed to operate a vehicle from rest (FHWA n.d.b). As such, it never manipulates the shift lever but assumes the car starts in park and then is automatically shifted into drive by the drive-by-wire subsystem and stays there for the duration of an experiment. The HDS experiments that simulate the cab as an ADS or CDA vehicle will require the subject (driver) to initiate the automation by taking some type of action. CARMA Platform already provides this capability in its user interface, so no new functionality is needed. However, if the driver elects to take over manual control at any time (by moving the steering wheel or a pedal), a component is needed to detect that override and switch the control output stream from CARMA Platform to the vehicle hardware so that the USEP will no longer see inputs from CARMA. This statement applies to the pedal commands only; for the steering wheel, the control output can always come from the cab hardware, as it will always reflect commands coming from CARMA as well as from the human driver.

TIME SYNCHRONIZATION

Time synchronization is important for ensuring consistency among all systems, which allows for accurate visualization of all entities in the integrated CDA-HDS system. Since the HDS experiments will have humans and real vehicle hardware-in-the-loop, it must operate within a small tolerance of realtime.

Previous CDA projects have leveraged two different approaches. The CDASim cosimulation tool (FHWA n.d.e) has a centralized time management system that regulates all simulation operations. The centralized time management system waits for all simulation entities to complete before iterating to the next time step. Like the current HDS, it could be linked to a 60-Hz clock that ensures it does not operate faster than realtime. The drawback of the centralized time-synchronization approach is that it can run slower than realtime if any of the subprocesses take longer to process. Because of this limitation, the STOL Distributed Testing framework (Loughran et al. 2024) manages time synchronization with a pseudo-real-time approach. In this approach, distributed simulation systems are synchronized on their actual clocks (which may or may not be synchronized), but they do not have a shared source of time, allowing the simulation to work in near realtime. However, this approach can potentially have drawbacks if different simulation entities significantly decohere (e.g., disagree on the state of the world at a specific time). Since the existing HDS system appears to support both approaches, the decision on which to use can be deferred until initial integration reveals the likely system performance. If speed allows, then a centralized approach is preferred, but if processing power is unavailable to ensure this, a pseudo-real-time approach will be used.

MAP CONSIDERATIONS

Digital roadway maps provide the basis for all the environment representation and the various ADS systems' world models. Therefore, the key for all participating systems is to use the same map information. Doing so will be straightforward using the Association for Standardization of Automation and Measuring Systems (ASAM) OpenDRIVE® standard as the baseline for all roadway maps (ASAM 2023). Most USEP simulation candidates use OpenDRIVE maps for road geometry but may use other data structures for roadside scenery, such as buildings. CARMA Platform and CARMA Cloud use OpenDRIVE maps as well (FHWA n.d.b, 2023g). CARMA Streets (FHWA 2023f) can also read OpenDRIVE maps and correlate them with the SAE J2735 MAP messages it publishes to approaching vehicles (SAE 2016). SUMO and PTV Vissim (Eclipse 2021; PTV Group n.d.) ingest maps in their own formats but provide tools for converting from OpenDRIVE. Future versions of the HDS and its ARCHER software (Williams et al. 2005) use scenarios that can be partially represented by OpenDRIVE maps.

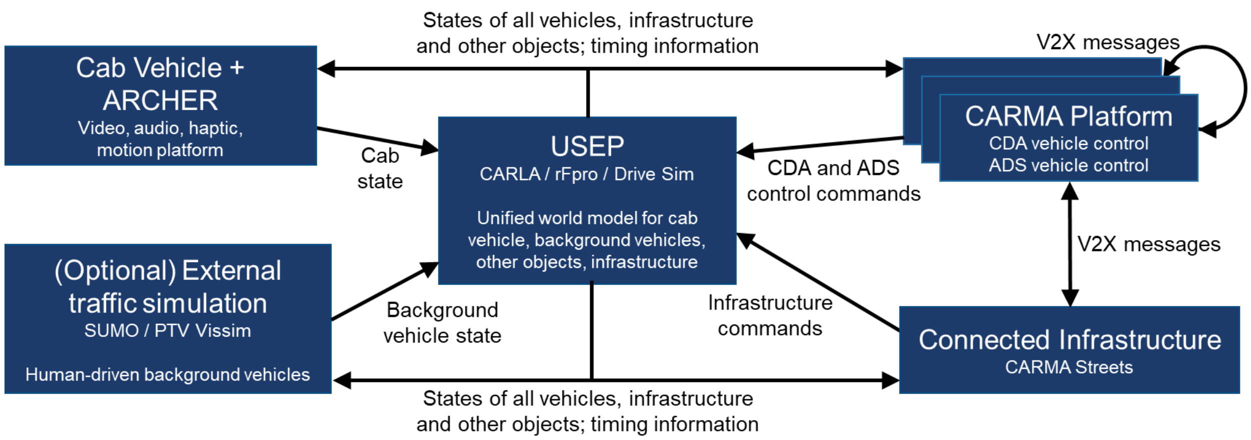
INTEROPERABILITY AND EXTENSIBILITY CONSIDERATIONS

Designing the new system with interoperability and extensibility considerations to keep pace with evolving needs for driving research and technology advancements is important. Keeping a central, authoritative set of world model data in one system supports such flexibility. Then, all parts of the system are subscribed to that world model, and everything is expected to be working from an identical state for any given entity. If a new level of physics fidelity or other behavior is needed for any given entity, such an external model can be plugged into the USEP API. With rigorous separation of concerns among the systems, subsystems, and components, revision or replacement of any given functionality will only have a localized impact, meaning that most of the system does not need to be touched to incorporate such changes. Such isolation of changes naturally results in higher productivity for the team maintaining and operating the system.

CHAPTER 5. INTEGRATION APPROACH

USEP APPROACH

The project team has chosen the USEP approach as the path forward because it provides the maximum flexibility and extensibility for a solution to grow with the latest CDA technology as it becomes available over the next several years and because it has a simplified format for adding functionality.[1] This architecture is described in the remainder of this chapter and is depicted in figure 8. A few alternatives that prioritized minimal disruption of the status quo have been discarded, as they all involve the common problem of trading away future flexibility for short-term cost savings.



Source: FHWA.

Note: ARCHER (Williams et al. 2005); SUMO (Eclipse 2021); PTV Vissim (PTV Group n.d.); CARLA (CARLA 2022); rFpro (rFpro n.d.); DRIVE Sim (NVIDIA n.d.a); CARMA Platform (FHWA n.d.b); CARMA Streets (FHWA 2023f).

Figure 8. Illustration. USEP concept for the HDS.

The central player in this approach is the USEP, which will be implemented as one of the major vehicle simulation platforms that can provide a detailed, dynamic, 3D simulation of the surrounding environment, including multiple vehicles and other objects in motion. The implementations under consideration are CARLA, rFpro, and NVIDIA DRIVE Sim (CARLA 2022; rFpro n.d.; NVIDIA n.d.a). Until a decision is made, this simulation will be referred to as the USEP. The key concept in this approach is that the USEP represents and manages the entire world model for all participating components in the simulation exercise. In this way, USEP acts as the sole source of ground truth information about all world data for all players. While the USEP manages and distributes these data, it does not necessarily generate all the data. Simulations external to the USEP simulations will handle the physics calculations of one or more of the objects in motion (such as a human-driven cab vehicle in the HDS), making this system a

[1]An example of a simplified mechanism for adding functionality is if the USEP is implemented with rFpro (rFpro n.d.), it can use a Simulink interface (MathWorks 2024), which provides a hook for external models of dynamic systems. The Simulink Toolbox provides domain-specific features, such as signal processing, control systems, and communications.

cosimulation. Each participating simulation provides the updated state (location, speed, etc.) of its specific objects to the USEP to update its world model. In particular, the cab vehicle will have its own simulation. Since none of these external simulations will be computing motion for all objects, they will depend on the USEP to provide updated state information for the remainder of the simulated world that they do not compute.

On the periphery of this diagram are the cab vehicle, some CARMA Platform (FHWA n.d.b) CDA vehicles, and possibly some human-driven scenario vehicles represented by an external simulation. In each case, these periphery software packages are responsible for processing the sensor perception of the vehicles they represent. These observations are then used to decide their respective control commands. Therefore, the only perception processing that USEP will perform will be for the CDA and ADS and human-driven vehicles that are fully controlled by it. This perception processing will entail the modeling of whatever sensors and perception systems are onboard a given vehicle based on its visibility of the world. (For example, representing the visual imagery captured by an ADS vehicle's windshield camera and processing those raw images as required by its guidance system, such as converting it to a list of nearby objects. Similar perception modeling will also be needed for the simulated human-driven background vehicles, which use human-driver behavior models.)

Physics Simulation

The physics being modeled in any driving simulation is an analysis of the dynamics affecting the motion of the objects. Dynamics calculations model the various forces acting on an object, such as a vehicle, and determine how these forces change the object's motion. In a vehicle, these forces result from applying control commands (that come from driver inputs or from a robotic controller), such as accelerating or turning, and from the external world, such as pavement and wind. The HDS scenarios will involve vehicles but often may involve other types of objects as well. The USEP will manage system-level timing so that the whole system operates in approximately realtime since the human test subjects will require such behavior to support realism. The overall system will expect each of its components to synchronize state information at a constant rate. Any individual component may elect to subdivide this frequency and operate at a faster (multiple) rate as long as its interactions with other components can maintain that systemwide frequency.

Simulating Vehicle Dynamics

The focus of the HDS is on the cab vehicle, where the experimental subjects (human operators or passengers) sit to experience the scenario. Additionally, other vehicles that act as traffic normally surround the cab vehicle to create a realistic driving scenario. These additional vehicles are referred to as scenario vehicles. Scenario vehicles may be any mix of simulated ADS vehicles, CDA vehicles, and human-driven vehicles.

Cab Vehicle

The cab vehicle will be controlled by the human driver for desired capabilities 1–3 and by its own instance of CARMA Platform software (FHWA n.d.b) for desired capability 4. The resulting control commands will be routed to the cab vehicle's dynamics simulation, SimVehicle, (FAAC n.d.b). This legacy simulation for the HDS comes with existing connections to the vehicle controls and to the motion, vision, haptic, and audio feedback systems already installed. SimVehicle will provide an updated cab vehicle state to the USEP simulation at each system time step. Since the HDS visual feedback loop runs much faster than the system loop, it will only provide states at the slower rate.

CDA or ADS Scenario Vehicles

All the AVs, whether they have communication ability or not, will be controlled by CARMA Platform (FHWA n.d.b), and their dynamic reaction to those control signals will be simulated by the USEP directly. The USEP simulation will maintain a representative vehicle model for each of the CDA and ADS vehicles in the scenario. Each of these will be controlled by a separate CARMA Platform instantiation that communicates control signals to the USEP as inputs to the USEP's vehicle dynamics calculation. The resulting state change of each vehicle will then be reported back to its respective copy of the CARMA Platform software to update its guidance logic.

Human-Driven Scenario Vehicles

Scenario vehicles that are (simulated to be) driven by humans but fill in the remainder of the scenario traffic can either be represented and managed by the USEP simulation or delegated to an external traffic simulation, such as SUMO or PTV Vissim (Eclipse 2021; PTV Group n.d.). If the scenario vehicles are managed by the USEP, the USEP will instantiate an internal vehicle model for each and use its own dynamics model to advance that vehicle's motion based on the driver behavior model used. Alternatively, the scenario traffic may be delegated if the number of vehicles involved is too large for the USEP to handle in realtime. If the vehicles are simulated externally, then that external simulation will manage the states and dynamics of those vehicles but will provide state updates to the USEP at each time step. For the external simulation to accurately produce the dynamics of these extra vehicles, the simulation will need to know about all the vehicles in the scenario, as their presence will affect how the human-driven vehicles will behave.

Dynamics of Other Moving Objects

Objects other than vehicles, such as pedestrians, bicyclists, scooter users, wheelchair users, or uncontrolled objects (e.g., a bouncing ball), may be in motion in the experiment scenario. The USEP simulation will control any of these objects that are present in the scenario. The exact type and number of such objects and their possible behaviors will depend on the simulation chosen for the USEP. These other dynamic objects will need to support visually lag-free updates.

INTEGRATING THE SYSTEM COMPONENTS

Researchers must ensure that the USEP and SimCreator (FAAC n.d.a) environments are not only compatible but also functionally the same. In other words, the common elements that will affect entity behavior (e.g., roadway geometry) need to be identical. Anomalies between simulation environments have the potential to be time consuming and frustrating to resolve. If problems are not discovered until after a study has taken place, the study's results could be invalidated. Having all simulations start from the same ground truth environment description is an effective way to avoid many of these problems (e.g., using a common .xdor data file to describe roadway geometry or Universal Scene Description for visual geometry conformance and standardization (Pixar 2021)). Having the cab vehicle get ground truth data from the USEP will help to ensure this common environment description. The design-time mitigation of such a risk is to have the unified world model managed solely by the USEP component and to have it manage the timing of all contributing simulations, either enforcing a systemwide clock pace or allowing some components to lag and updating those corresponding objects in the unified world model accordingly.

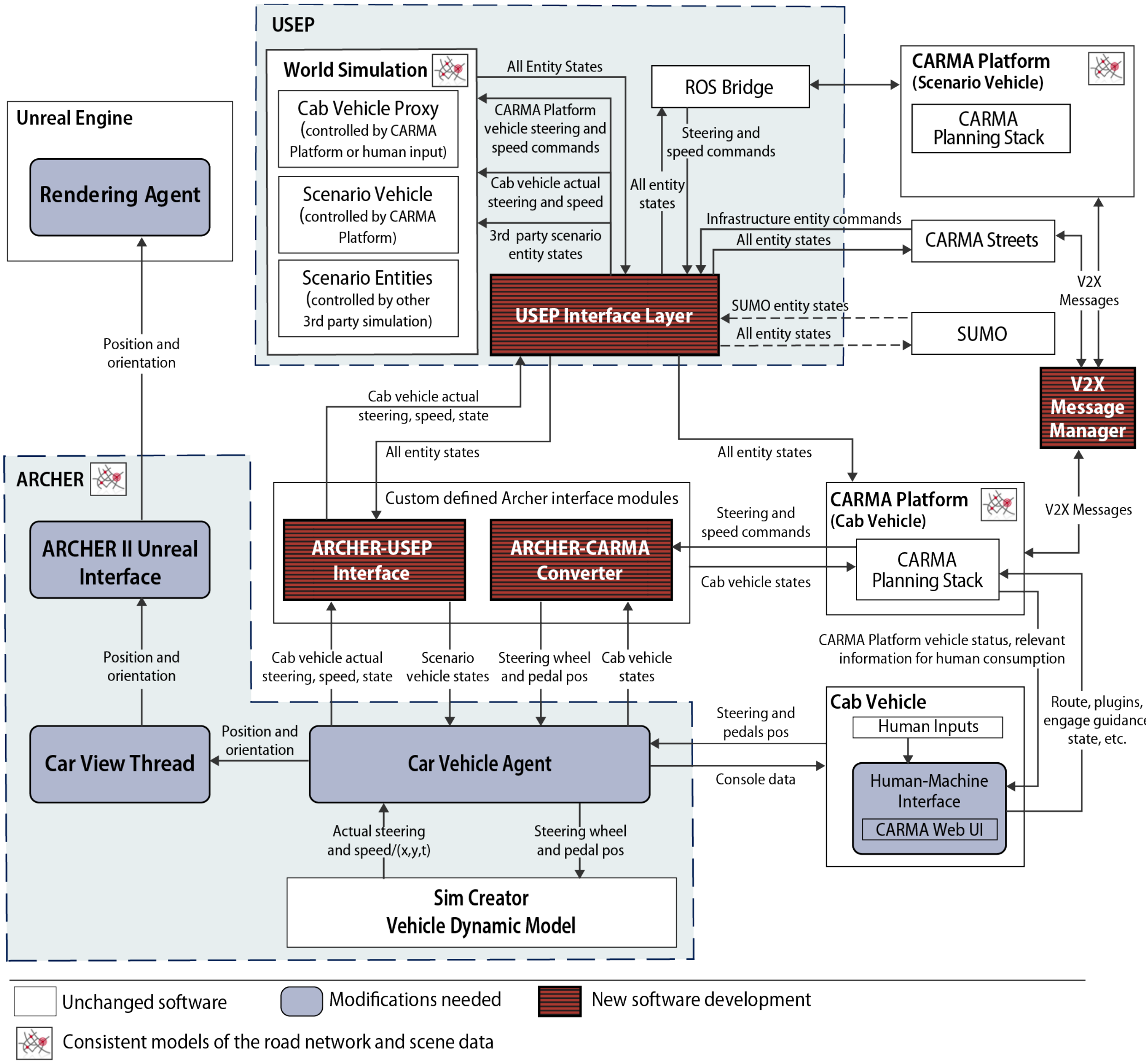
Some past HF experiments in the HDS required the use of dynamic maps (content generated on the fly based on human subject behavior). This high-level architecture of the integrated system could support dynamic maps. However, CARMA Platform currently requires a fully defined route at the beginning of its operation (FHWA n.d.b). How the system will react if the map is changed while route following is underway is unknown. This situation will need to be addressed to support the new CDA-HDS integration if the use of dynamic maps is deemed a requirement.

A more detailed view of the system structure is shown in figure 9. The USEP is shown here in an expanded view, as an assembly centered on the unified world model simulation system but also including some interface components. The ROS bridge (Open Robotics 2021) is a set of existing code that connects CARMA Platform to CARLA in the CDASim (FHWA n.d.b; CARLA 2022). The USEP interface layer may be zero or more components that translate data formats, protocols, or timings to fit the specific needs of the product ultimately chosen as the world simulation. At the time of this writing, the project team is evaluating several candidates for the world simulation. The decision can be postponed until the detailed design commences.

Desired capabilities 1 and 2 (CDA scenario traffic) are primarily represented in the top part of figure 9, where the USEP cosimulation is connected to the CarVehicleAgent component (an HDS wrapper for SimCreator (FAAC n.d.a)). This structure allows continued use of the legacy simulation for the cab vehicle so that the cab software can view the world data through the lens of its accustomed interfaces. In this architecture, CARMA Platform (FHWA n.d.b) shares steering and speed commands of a scenario CDA vehicle (upper right of figure 9) with the USEP simulator through a ROS (Open Robotics 2021) bridge that allows CARMA Platform to use the same ROS interfaces in simulation that are currently used in the physical vehicles. The USEP's vehicle dynamics models are then used to determine the true vehicle state (position, orientation, velocity, and acceleration) of the CDA vehicle, and USEP's sensor simulation modules can provide radar, light detection and ranging (LiDAR), and camera data as inputs to CARMA Platform's perception stack, when necessary. The CDA vehicle state, output from USEP's vehicle dynamics model, is sent (via two interface components) to CarVehicleAgent. CarVehicleAgent uses the incoming CDA vehicle state to place a vehicle proxy in its local world model, allowing it to be displayed as a neighbor vehicle in the scene. In the reverse direction, CarVehicleAgent shares the latest state of the cab vehicle with the USEP simulator. The USEP then updates its local proxy for the cab vehicle with these state data. The USEP closes the loop by pushing the updated cab vehicle and CDA vehicle state information, along with simulated sensor data, to CARMA Platform, representing the CDA vehicle via the ROS bridge. If additional scenario vehicles are human driven, they are simulated entirely within the USEP (or delegated to an external simulation), and their data are handled the same way the USEP handles CDA vehicle data, except that control commands are generated by the designated simulation.

The description here of a single CDA scenario vehicle can be expanded to represent any number of CDA vehicles simultaneously within the limits of the hardware resources.

The right side of the diagram includes SUMO as a candidate for providing the other (non-CDA) scenario vehicle simulation described previously (Eclipse 2021). The SUMO box and data flows are dashed lines to indicate that it is an optional addition and not required for system operation.



Source: FHWA.

UI = user interface.

Figure 9. Diagram. Architecture of the integrated CDA-HDS system.

For desired capability 3, where connected infrastructure plays a part, the upper part of figure 9 shows CARMA Streets performing the role of infrastructure controller, which may be dictating the cycles of a traffic signal controller, coordinating traffic flow in the nearby roadway, or simply communicating status information with passing vehicles (FHWA 2023f). In any case, CARMA Streets communicates directly with any CDA vehicles in the scenario with simulated V2X messages. Since none of the world simulation products provide V2X messaging capabilities, these messages are simulated outside the USEP by a separate V2X Message Manager. This component acts as a specialized messaging broker, ensuring that originated messages are delivered to the appropriate scenario participants. The V2X Message Manager component could be configured to simulate imperfections, such as radio range limitations, dropped messages, or corrupted message content, as desired.

Changes to CarViewThread may be needed; however, the project team anticipates that view state is still passed between CarVehicleAgent and the rendering pipeline. The ARCHER II Unreal Interface (Williams et al. 2005) will pick up the view state and pass it to the Unreal engine (Epic Games 2024) via a rendering agent in the interface. Auxiliary view-related data may also be required, including driver’s eye camera data and possibly additional virtual camera orientations.

To enable desired capability 4, where the cab vehicle is robotically controlled by CARMA Platform, a separate instance of CARMA Platform is used for cab vehicle control and is shown in the lower right portion of figure 9 (FHWA n.d.b). A switch (hardware or software) will indicate whether the cab vehicle will be controlled by CARMA Platform (automated) or by a human subject. This choice dictates the source of the control commands going from the cab vehicle into CarVehicleAgent. In a situation where the cab is under robotic control during an experiment, two needs must be met. First, the steering wheel needs to turn to reflect the robotic steering commands so the human subject will believe the turning motion is real. To achieve this, a motor will need to be attached to the steering linkage and controlled according to the robotic steering control signal. Second, the capability must exist for the human driver to override the robot and take over manual control at any time. The most intuitive ways to do this are for the driver to tap the brake pedal or to grab the steering wheel. In real CARMA vehicles, these options are available and sensed by the low-level controller, which breaks off the robotic control signal so that it will not interfere with the human’s desires. In the HDS, the cab vehicle hardware will also need to sense these human override gestures and send an appropriate signal to the cab software. Such a signal would be able to toggle the switch mentioned at the beginning of this paragraph, disengaging the CARMA commands. Once the cab vehicle has been put into manual mode (either from a manual switch setting at the beginning of a scenario or through en route override action), it must stay in manual mode for the remainder of the drive since the current version of CARMA Platform is not designed to reengage. The possibility of modifying CARMA Platform to allow en route reengagement exists.

Regardless of whether the cab is controlled by the human subject or CARMA Platform (FHWA n.d.b) at any moment, figure 9 shows both of them sending steering and speed commands into CarVehicleAgent. CarVehicleAgent decides which of these signals to listen to and which to ignore. CarVehicleAgent uses the resulting signal to control the motion platform and other experiential hardware and to send appropriate data back to the cab vehicle’s console for feedback to the human occupant(s).

The ARCHER–CARMA converter receives cab vehicle perception information, such as localization and external objects, and vehicle status from CarVehicleAgent. The converter transforms these data into a ROS message that can be understood by CARMA and publishes it to a prepared ROS topic (Open Robotics 2021). CARMA Platform subscribes to this topic and uses the information to update its view of the world (FHWA n.d.b). Additionally, the CARMA Web user interface software serves as a human–machine interface, displaying the current vehicle state on a tablet in the cab vehicle.

LEVERAGING TENA

While TENA is not part of this architecture specification, the approach does not preclude it, and the addition of TENA connectivity would be straightforward (DoD 2020). The project team recommends that TENA be considered as a future enhancement when the desire to connect the HDS to any other remote system (e.g., another driving simulator) arises. TENA's performance will need to be evaluated for the real-time data update performance. However, requesting data updates using TENA should not be a problem. Across the approaches previously discussed, adding TENA adapters to the system could improve the connectivity of the HDS and allow for distributed studies and operations. This additional feature is particularly attractive due to the ongoing work at STOL. Several project researchers have integrated, or are in the process of integrating, TENA adapters so they can participate in a network test via FHWA's Distributed Testing framework (Loughran et al. 2024). Additionally, FHWA's Distributed Testing Framework and TENA would allow the HDS to operate in a true cosimulation environment, where independent, distributed simulations can interact and cooperate.

TENA adapters are small applications built to interface with the TENA middleware, which is an environment similar to microservices or message-passing distributed computing architectures. The TENA methodology focuses on shared data objects, which define data to be passed within the simulation. Controls for who and what can join a specific simulation exist as well as tools that can debug and display data in the simulation. Generally, TENA allows for a geographically distributed simulation with defined data objects being passed between entities. The distributed nature of TENA and the defined data objects could benefit the HDS, barring any significant performance issues. Leveraging TENA would allow the HDS to participate in simulations or experiments with other entities, such as driving simulators, equipment manufacturers, or research institutes. For all the desired capabilities, a TENA adapter would be built to send and receive data related to the object's class, so a traffic vehicle adapter, roadside infrastructure adapter, and cab vehicle adapter would be built. A corresponding set of adapters is also required at the alternate end(s) of the simulation. CARMA Platform, V2X Hub, and CARLA (FHWA n.d.b, FHWA 2023i; CARLA 2022) have previously built adapters that have been used in various experiments, so the only new interfaces would be on the HDS end for initial use (FHWA 2024a). Adapters could be built for rFpro or DRIVE Sim if needed (rFpro n.d.; NVIDA n.d.a). For safety reasons, the cab's motion control should stay local so that only outputs from the cab vehicle would be output to a TENA adapter.

CHAPTER 6. CONCLUSIONS

With an integrated CDA-HDS system, FHWA will be able to study how drivers behave and how they cooperate with the transportation ecosystem, including other human drivers, CAVs from various sources, and infrastructure elements that are under development.

The architecture presented in this report, centered on the USEP, describes the recommended approach to carry the HDS forward through the 2020s. The CDA-HDS integration will enable additional collaboration and new research opportunities in the future. Once implemented, this architecture will also increase the speed at which simulator studies can take place to improve roadway safety and operations.

REFERENCES

AB Dynamics. n.d. "Testing the future of mobility" (web page). https://www.abdynamics.com/#, last accessed November 14, 2024.

Aimsun. n.d. "aimsun.next" (web page).https://www.aimsun.com/aimsun-next-overview, last accessed June 17, 2024.

Ansible Motion. n.d. "Advanced driving simulators for automotive product development" (web page). https://www.ansiblemotion.com/, last accessed November 14, 2024.

Argonne National Laboratory. N.d. "The Message Passing Interface (MPI) Standard" (web page). https://www.mcs.anl.gov/research/projects/mpi/, last accessed January 22, 2024.

ASAM. 2023. *ASAM OverDRIVE* (software). Version 1.8.0.

Autoware. 2023. "Welcome to the Autoware Foundation" (web page). https://autoware.org/, last accessed October 16, 2023.

AVL. n.d. "Reimagining Motion for a greener, safer, better world of mobility" (web page). https://www.avl.com/en-us, last accessed November 14, 2024.

Balse, A. U., M. Bellamkonda, Z. R. Cooper, A. Ghiasi, Z. Huang, R. Laqui, E. M. Leslie, et al. 2024. *2022 CDA Annual Report*. Report No. FHWA-HRT-24-089. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-24-089.pdf, last accessed December 6, 2024.

Barco. n.d. "Discover how Barco can help you" (web page). https://www.barco.com/en, last accessed November 14, 2024.

Benatti, S., C. Schwarz, A. Young, A. Elmquist, R. Serban, and D. Negrut. 2022. *A Geographically Distributed Simulation Framework for the Analysis of Mixed Traffic Scenarios Involving Conventional and Autonomous Vehicles*. SAE Technical Paper 2022-01–0839. Warrendale, PA: SAE International. https://doi.org/10.4271/2022-01-0839, last accessed October 23, 2024.

CARLA. 2022. *Cars Learning to Act* (software). Version 0.9.14.

Claytex. n.d. "What are you interested in simulating?" (web page). https://www.claytex.com/, last accessed November 14, 2024.

Concurrent Real-Time. n.d. "Concurrent Real-Time" (web page). https://concurrent-rt.com/, last accessed November 14, 2024.

Dassault Systemes. n.d. "Dymola" (web page). https://www.3ds.com/products/catia/dymola, last accessed November 14, 2024.

Dataspeed Inc. 2024. "Drive-by-Wire-Kit" (web page). Drive-by-Wire Kit, An Autonomous Vehicle By-Wire Solution | Dataspeed, last accessed December 10, 2024.

DoD. 2011. *Modeling and Simulation (M&S) Glossary).* https://www.acqnotes.com/Attachments/DoD%20M&S%20Glossary%201%20Oct%2011.pdf, last accessed January 2, 2025.

DoD. 2020. "Test and Training Enabling Architecture" (web page). https://www.trmc.osd.mil/tena-home.html, last accessed December 10, 2024.

DoD. 2024. "Department of Defense Modeling and Simulation Standards" (preview on GitHub repository). https://github.com/open-dis/dis-tutorial/blob/master/DoDModelingAndSimulationStandards.md, last accessed October 18, 2023.

Domeprojection. n.d. "Innovative Solutions for Professional Projection Systems" (web page). https://www.domeprojection.com/, last accessed November 14, 2024.

dSPACE. n.d. "Automotive Simulation Models (ASM)" (web page). https://www.dspace.com/en/inc/home/products/sw/automotive_simulation_models.cfm, last accessed November 20, 2024.

Dynisma. n.d. "Motion Simulation Redefined" (web page). https://www.dynisma.com/, last accessed November 14, 2024.

Eclipse. 2021 *Simulation of Urban Mobility* (software). Version 1.15.0.

Eclipse. 2024. *MOSAIC* (software). Version 21.1. https://www.eclipse.org/mosaic/, last accessed September 23, 2024.

Epic Games. 2024. *Unreal Engine* (software). Version 5.3. https://www.unrealengine.com/en-US/blog/unreal-engine-5-3-is-now-available, last accessed December 10, 2024.

EyesDX. 2009–2024. "Products" (web page). https://www.eyesdx.co/products.html, last accessed December 10, 2024.

FAAC. n.d.a. "SimCreator: Real Time Simulation & Modeling System" (web page). https://www.faac.com/realtime-technologies/products/simcreator/, last accessed October 16, 2023.

FAAC. n.d.b. "SimVehicle: Real Time Vehicle Dynamics Model" (web page). https://www.faac.com/realtime-technologies/products/simvehiclelt/, last accessed October 16, 2023.

FAAC. n.d.c. "SimDriver: Autonomous Vehicle Control Solution" (web page). Accessed October 16, 2023. https://www.faac.com/realtime-technologies/products/simdriver/, last accessed October 16, 2023.

FAAC. n.d.d. "Realtime Technologies: Simulation for Research" (web page). https://www.faac.com/realtime-technologies/, last accessed October 16, 2023.

FHWA. n.d.a. "cdasim" (software and configuration files in GitHub repository). https://github.com/usdot-fhwa-stol/cdasim, last accessed April 12, 2023.

FHWA. n.d.b. *CARMA Platform* (software). Version 4.3.0.

FHWA. 2022a. "CARMA Program Overview" (web page). https://highways.dot.gov/research/operations/CARMA, last accessed September 17, 2024.

FHWA. 2022b. *Cooperative Driving Automation Program*. Publication No. FHWA-HRT-22-039. Washington, DC: Federal Highway Administration. https://doi:org/10.21949/1521729, last accessed December 11, 2024.

FHWA. 2023a. *Perception Sharing for Cooperative Driving Automation (CDA)*. Publication No. FHWA-HRT-24-026. Washington, DC: Federal Highway Administration. https://doi.org/10.21949/1521466, last accessed October 23, 2024.

FHWA. 2023b. *Work Zone Management for Light Vehicles Using Cooperative Driving Automation (CDA)*. Publication No. FHWA-HRT-24-028. Washington, DC: Federal Highway Administration. https://doi.org/10.21949/1521468, last accessed October 23, 2024.

FHWA. 2023c. *Adaptive Traffic Signal Control Optimization in a Cooperative Driving Automation (CDA) Environment.* Publication No. FHWA-HRT-24-025. Washington, DC: Federal Highway Administration. https://doi.org/10.21949/1521465, last accessed October 23, 2024.

FHWA. 2023d. *Integrated Highway Prototype Using Cooperative Driving Automation (CDA).* Publication No. FHWA-HRT-24-024. Washington, DC: Federal Highway Administration. https://doi.org/10.21949/1521464, last accessed October 23, 2024.

FHWA. 2023e. *CARMA Messenger* (software). Version 4.5.0.

FHWA. 2023f. *CARMA Streets* (software). Version 4.3.0.

FHWA. 2023g. *CARMA Cloud* (software). Version 4.6.0.

FHWA. 2023h. *Move Over Law for Traffic Incident Management Using Cooperative Driving Automation (CDA)*. Publication No. FHWA-HRT-24-027. Washington, DC: Federal Highway Administration. https://doi.org/10.21949/1521467, last accessed October 23, 2024.

FHWA. 2023i. *V2X Hub* (software). Version 7.4.0.

FHWA. 2023j. *Saxton Transportation Operations Laboratory Research Tools to Advance Cooperative Driving Automation (CDA) Research*. Publication No. FHWA-HRT-24-034. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-24-034_0.pdf, last accessed October 24, 2024.

FHWA. 2024a. *VOICES Cooperative Driving Automation (CDA) Proof-of-Concept Systems Integration Test 1*. Report No. FHWA-HRT-24-072. Washington, DC: Federal Highway Administration. https://doi.org/10.21949/1521742, last accessed April 29, 2024.

FHWA. 2024b. *Saxton Transportation Operations Laboratory Cooperative Driving Automation (CDA) Tools to Improve Road User Safety*. Publication No. FHWA-HRT-24-036. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-24-036_0.pdf, last accessed December 11, 2024.

FHWA. 2024c. *Saxton Transportation Operations Laboratory Research Tools to Enhance Connected and Automated Vehicle (CAV) Capabilities*. Publication No. FHWA-HRT-24-35. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-24-035.pdf, last accessed December 11, 2024.

Hexagon. n.d. "Virtual Test Drive" (web page). https://hexagon.com/products/virtual-test-drive, last accessed November 14, 2024.

IEEE. 2010. *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) – Framework and Rules*. IEEE Standard 1516-2010, inactivated March 2021. Piscataway, NJ: IEEE.

Inman, V. 2010. *An Evaluation of Signing for Three-Lane Roundabouts*. Publication No. FHWA-HRT-10-030. Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/HWA-HRT-10-030.pdf, last accessed October 23, 2024.

IPG. n.d. "CarMaker" (web page) https://www.ipg-automotive.com/en/products-solutions/software/carmaker/, last accessed November 20, 2024.

Kearney, J., J. Plumert, C. Schwarz, S. Baek, E. O'Neal, W. Wang, and D. McGehee. 2022. *Learning About Driver and Pedestrian Behaviors Through Connected Simulation Technology.* Report No. FHWA-HRT-20-059.Washington, DC: Federal Highway Administration. https://highways.dot.gov/sites/fhwa.dot.gov/files/FHWA-HRT-20-059.pdf, last accessed December 6, 2024.

Loughran, A., Y. Lou, R. Laqui, and D. Chou. 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 December 6, 2024.

Malik, S., M. A. Khan, and H. El-Sayed. 2022. "CARLA: Car Learning to Act — An Inside Out." *Procedia Computer Science* 198: 742–749. https://doi.org/10.1016/j.procs.2021.12.316, last accessed October 23, 2024.

Martin, P. T., M. Zlatkovic, and I. Tasic. 2012. *Utah Traffic Lab Driving Simulator: User Manual and Flex Lanes Scenario Development*. Salt Lake City, UT: University of Utah.

MathWorks. 2024a. "MathWorks" (web page). https://www.mathworks.com/, last accessed December 31, 2024.

MathWorks. 2024b. "MATLAB" (web page). https://www.mathworks.com/products/matlab.html, last accessed December 31, 2024.

MathWorks. 2024c. "Simulink" (web page). https://www.mathworks.com/products/simulink.html, last accessed October 22, 2024.

Mechanical Simulation Corporation. 2024.*CarSim* (software). Version 2024.3.

Meta. 2024. "This is Meta Quest" (web page). https://www.meta.com/quest/, last accessed December 31, 2024.

Michelin. n.d. "About Canopy" (web page). https://simulation.michelin.com/canopy, last accessed November 14, 2024.

Modelon. n.d. "Design better with cloud-based simulation" (web page). https://modelon.com/, last accessed November 14, 2024.

Moog. n.d. "Shaping the way our world moves" (web page). https://www.moog.com/, last accessed October 16, 2023.

National Instruments. n.d. "Discover NI's next-generation USB DAQ hardware" (web page). https://www.ni.com/en, last accessed November 14, 2024.

New Eagle. 2024. "Drive-by-Wire Kits Widen Horizons for Production-Scale Autonomous Technology." *Autonomy* (blog), New Eagle. April 4, 2019. https://neweagle.net/blog/autonomous-tech-horizons/, last accessed December 10, 2024.

Norxe. n.d. "Norxe" (web page). https://norxe.com/, last accessed November 14, 2024.

nsnam. 2021–2024. *ns-3* (software). Version 3.35. https://www.nsnam.org/, last accessed September 23, 2024.

NVIDIA. n.d.a "NVIDIA DRIVE Sim" (web page). https://developer.nvidia.com/drive/simulation, last accessed September 23, 2024.

NVIDIA. n.d.b. "NVIDIA Omniverse" (web page). https://www.nvidia.com/en-us/omniverse/, last accessed September 23, 2024.

Open Robotics. n.d. “pacmod” (web page). pacmod - ROS Wiki, last accessed December 10, 2024.

Open Robotics. 2021. “ROS – Robot Operating System” (web page). https://ros.org/, last accessed October 23, 2024.

ORNL. 2022. “Real-Sim/Real-Twin: Multi-Resolution X-in-the-Loop Simulation for Connected and Automated Vehicles.” Presentation online December 15, 2022.

OSTL. n.d. “Open Source Technology Lab: A Fair Playground for Technology Innovation” (web page). https://www.ostl.org/index.html, last accessed November 14, 2024.

Perez, W. A., M. A. Bertola, and B. Philips. 2016. *Active Traffic Management: Comprehension, Legibility, Distance, and Motorist Behavior in Response to Selected Variable Speed Limit and Lane Control Signing*. Report No. FHWA-HRT-16-037. Washington, DC: Federal Highway Administration.

Pixar. 2021. “Introduction to USD” (web page). https://openusd.org/release/intro.html, last accessed October 22, 2024.

Project Chrono. 2024. “State Synchronization with DDS and MPI” (web page). https://api.projectchrono.org/development/state_synchronization.html#:~:text=DDS%20is%20a%20more%20recent,are%20published%20and%20subscribed%20to, last accessed September 1, 2024.

PTV Group. n.d. “PTV Vissim: Multimodal Traffic Simulation Software” (web page). https://www.ptvgroup.com/en-us/products/ptv-vissim, last accessed October 16, 2023.

rFpro. n.d. “rFpro” (web page). https://rfpro.com/, last accessed September 23, 2024.

SAE. 2016. *Dedicated Short Range Communications (DSRC) Message Set Dictionary*. SAE Standard J2735_201603. Warrendale, PA: SAE International.

SAE. 2021a. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. SAE J3016_202104. Warrendale, PA: SAE International.

SAE. 2021b. *Taxonomy and Definitions for Terms Related to Cooperative Driving Automation for On-Road Motor Vehicles*. SAE J3216_202107. Warrendale, PA: SAE International.

Sawyer, B. D., and P. A. Hancock. 2012. “Development of a Linked Simulation Network to Evaluate Intelligent Transportation System Vehicle to Vehicle Solutions.” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 56, no. 1: 2316–20. https://doi.org/10.1177/1071181312561487, last accessed October 23, 2024.

Scalable Display Technologies. n.d. “Scalable Display Technologies: Innovating the World of Projection & Display” (web page). https://www.scalabledisplay.com/company/, last accessed November 14, 2024.

Shi, E., T. M. Gasser, A. Seeck, and R. Auerswald. 2020. "The Principles of Operation Framework: A Comprehensive Classification Concept for Automated Driving Functions." *SAE International Journal of Connected and Automated Vehicles* 3, no. 1: 27–37. https://doi.org/10.4271/12-03-01-0003, last accessed October 23, 2024.

Siemens. n.d. "Siemens" (web page). https://www.siemens.com/us/en/, last accessed November 14, 2024.

Smart Eye. n.d. "Smart Eye" (web page). https://www.smarteye.se/smart-eye-pro/, last accessed December 10, 2024.

Solernou, A., R. Romano, I. Souflas, F. Hajiseyedjavadi, E. Paschalidis, and N. Merat. 2020. "Integration and Training of a ROS Autonomous Driver for Human-like Driving Style in a Complex Multi-Component Driving Simulator." In *Proceedings of the Driving Simulation Conference 2020 Europe VR*. Driving Simulation Association, Antibes, France: Driving Simulation Association. 11–18.

Speedgoat. n.d. "Speedgoat: Real-Time Simulation and Testing" (web page). https://www.speedgoat.com, last accessed November 14, 2024.

Swanson, K. S., A. A. Brown, S. N. Brennan, and C. M. LaJambe. 2013. "Extending Driving Simulator Capabilities Toward Hardware-in-the-Loop Testbeds and Remote Vehicle Interfaces." In *2013 IEEE Intelligent Vehicles Symposium (IV)*. Gold Coast City, Australia: IEEE. 122–27. https://doi.org/10.1109/IVS.2013.6629458, last accessed October 23, 2024.

Synchrono. n.d. "Synchrono Manufacturing Software" (web page). https://www.synchrono.com/, last accessed November 26, 2024.

Tesis. n.d. "Tesis" (web page). https://www.tesis.com/en/, last accessed November 14, 2024.

University of Leeds. n.d. "Highly Immersive Kinematic Experimental Research (HIKER) Pedestrian Lab." *Virtuocity*. https://uolds.leeds.ac.uk/facility/hiker-lab/, last accessed November 25, 2024.

Varjo. n.d. "The world's most advanced virtual and mixed reality" (web page). https://varjo.com, last accessed January 2, 2025.

VI-Grade. n.d. "VI-Grade" (web page). https://www.vi-grade.com/, last accessed November 14, 2024.

Vioso. n.d. "Projection Software: Warping and Blending for Multiple Projectors" (web page). https://vioso.com/, last accessed November 14, 2024.

Weaver, S. M., S-F. Chao, and B. H. Philips.2022. *Driver Adaptation to Vehicle Automation: The Effect of Driver Assistance Systems on Driving Performance and System Monitoring*. Report No. FHWA-HRT-22-072. Washington, DC: Federal Highway Administration. https://www.fhwa.dot.gov/publications/research/safety/22072/index.cfm, last accessed October 23, 2024.

Williams, J. R., T-C. Chou, B. L. Wallick, and J. M. Moyer. 2005. "Advanced Rendering Cluster for Highway Experimental Research." *Proceedings of Driving Simulation Conference North America*. Orlando, FL. 49–58. Advanced_Rendering_Cluster_Highway_Experimental_Research.pdf, last accessed October 23, 2024.

Recycled
Recyclable

Recommended citation: Federal Highway Administration, *Cooperative Driving Automation—Highway Driving Simulator Architecture: High-Level Architecture Design* (Washington, DC: 2025) https://doi.org/10.21949/1521547

HRSO-30/05-25(WEB)E