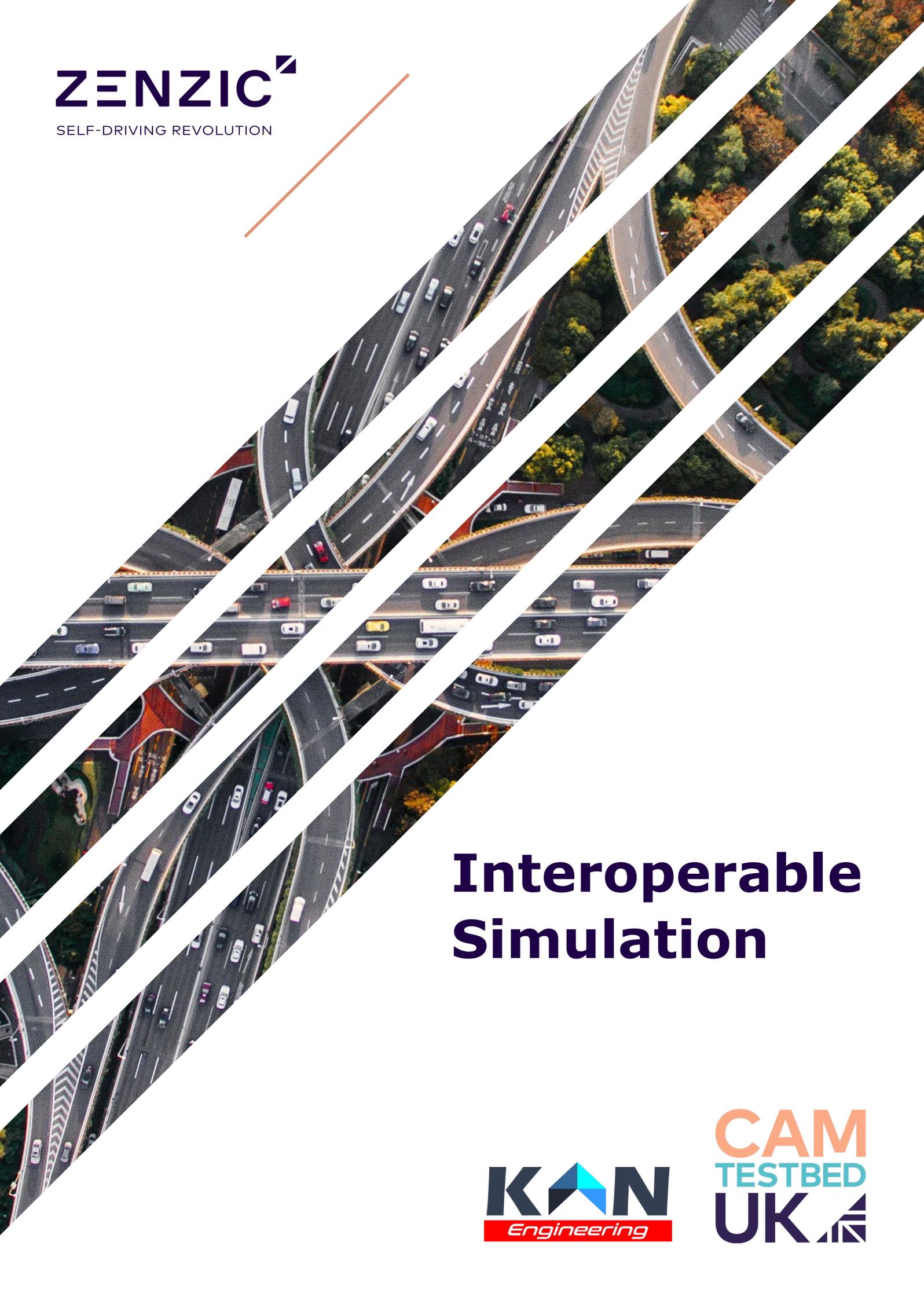


**ZENZIC**<sup>4</sup>

SELF-DRIVING REVOLUTION



# Interoperable Simulation

**KAN**  
Engineering

**CAM**  
TESTBED  
UK

## Disclaimer

This report has been written by KAN Engineering, contributed to by CAM Testbed UK and the University of Leeds and compiled by Zenzic. Any views expressed in this report are not necessarily those of Zenzic.

The information contained herein is the property of these organisations and does not necessarily reflect the views or policies of the customer for whom this report was prepared. Whilst every effort has been made to ensure that the matter presented in this report is relevant, accurate and up-to-date, Zenzic and/or any of the authors of this report cannot accept any liability for any error or omission, or reliance on part or all of the content in case of incidents that may arise during trialling and testing. In addition, Zenzic and/or any of the authors of this report cannot accept any liability for any error or omission, or reliance on part or all of the content in another context.

When in hard copy, this publication is printed on paper that is FSC (Forest Stewardship Council) and TCF (Totally Chlorine Free) registered.

**For further information on this report please contact the Zenzic team**

Email: [info@zenzic.io](mailto:info@zenzic.io)

Web: [zenzic.io](http://zenzic.io)

## About the principal author

### Dr Amir Soltani

Dr Amir Soltani is the founder and director of KAN Engineering and a visiting lecturer at Cranfield University, UK. He received his PhD degree in Automotive Mechatronic in 2014 from Cranfield University and completed his MSc (in 1993) and BSc (in 1990) in Mechanical Engineering. He has more than 30 years of experience working in the automotive and energy domains. He has been successfully involved in more than 70 cutting-edge multi-disciplinary research and development projects at various technical and managerial levels. His competencies include the development of mechatronic systems for automotive applications using Model-Based Development (MBD) approaches. He is a world-leading expert in advanced vehicle simulations, including the design and development of distributed simulation environments, for connected and automated vehicles.

In addition to the principal author, the project partners have also authored sections relating to their technical contributions to the project.

## Acknowledgements

- Zenzic Project Lead – Tristan Bacon
- Zenzic Project Manager – Gareth Purdy

Special thanks to the industrial contributors (Ford Motor Company, Amazon Web Services (AWS), IPG Automotive UK and RTI) whose contributions were instrumental in the project team generating significant and successful results.

Special thanks also to the Advisory Group for guiding and shaping this work. The advisory group comprised: Department for Transport (DfT), Driver and Vehicle Standards Agency (DVSA), Vehicle Certification Agency (VCA), Institute of Digital Engineering (IDE), Connected Places Catapult, BSI, Oxfordshire County Council, Ordnance Survey, Oxbotica, IPG Automotive, Aurrigo, University of Leeds Institute for Transport Studies, Centre for Connected and Autonomous Vehicles (CCAV), Department for Business, Energy & Industrial Strategy (BEIS) and Nissan.

# Contents

<b>Disclaimer</b>	<b>ii</b>
<b>About the principal author</b>	<b>iii</b>
<b>Contents</b>	<b>iv</b>
<b>Executive summary</b>	<b>1</b>
<b>1   Introduction</b>	<b>1</b>
<b>2   Taxonomy definition for UK simulation capabilities</b>	<b>3</b>
<b>3   Interoperable Simulation PoC, Phase 3</b>	<b>4</b>
3.1 Project objectives	4
3.2 Project partners	4
3.3 Simulation approach	8
3.4 Capabilities involved	11
3.5 Project use cases	12
<b>4   Cloud-based connectivity solution</b>	<b>18</b>
4.1 Connectivity options	18
4.2 Implementation	21
4.3 Connections	21
4.4 Future solutions	23
4.5 System latency	23
<b>5   KAN Simulation Manager (KANSM)</b>	<b>27</b>
5.1 KAN Core	28
5.2 KAN API	28
5.3 Simulation modules (Tools)	29
5.4 Test configuration	29
5.5 Test execution	29
5.6 KAN Scenario Engine (KANSE)	30
5.7 Data recording	30
5.8 Data visualisation	31
5.9 User Interface (UI)	31
5.10 Integrated simulation capabilities	32
5.11 Closed-loop simulation (V&V)	33
<b>6   KAN Scenario Engine (KANSE)</b>	<b>35</b>
6.1 Introduction	35
6.2 ASAM OpenSCENARIO standard and runtime behaviour	35
6.3 KANSE high level architecture	35
6.4 Functions and features	36
6.5 Current limitations	37
<b>7   System integration</b>	<b>38</b>
7.1 Introduction	38
7.2 UTAC	39
7.3 Applus IDIADA CAVWAY	39
7.4 University of Leeds	40
7.5 SMLL	41

7.6	HORIBA MIRA	41
7.7	WMG	41
7.8	Simulation setup	42
<b>8  </b>	<b>UTAC: MiL, SiL, ViL simulation and test track</b>	<b>48</b>
8.1	Interoperable simulation role	48
8.2	UTAC Millbrook Proving Ground	50
8.3	UTAC Millbrook private network	52
8.4	Integration with the Interoperable Simulation project	53
8.5	Summary	54
<b>9  </b>	<b>Applus IDIADA: V2V HiL Simulation</b>	<b>55</b>
9.1	Interoperable Simulation role	55
9.2	CAVWAY Proving Ground	55
9.3	IDAPT V2X units	56
9.4	MQTT-based control infrastructure	57
9.5	Environment model	58
9.6	OpenSCENARIO scripting	60
9.7	Web-based management and mapping	61
9.8	Integration with Interoperable Simulation	61
9.9	Summary	62
<b>10  </b>	<b>University of Leeds: Pedestrian-in-the-Loop</b>	<b>63</b>
10.1	Introduction	63
10.2	The HIKER	63
10.3	Integration	63
10.4	SimulatorD	64
10.5	Turner nodes	65
10.6	Discussion	65
10.7	Conclusions	67
<b>11  </b>	<b>SMLL: Scenario generation</b>	<b>68</b>
11.1	Introduction	68
11.2	Objectives and role in the project	68
11.3	The importance of real-world scenarios	68
11.4	Connectivity	69
11.5	Utilised capabilities at SMLL	69
11.6	Scenario generation pipeline	71
11.7	Detecting EEBL	72
11.8	Detecting AEB-P	73
11.9	Interoperability	75
<b>12  </b>	<b>HORIBA MIRA: Oracle module</b>	<b>76</b>
12.1	Introduction	76
12.2	Oracle challenges and background	76
12.3	HORIBA MIRA Oracle workflow	78
12.4	Integration benefits for HORIBA MIRA and CAM Testbed UK	80
12.5	Integration	82
12.6	Results	84
12.7	Discussion	85
<b>13  </b>	<b>WMG: V&amp;V pipeline</b>	<b>86</b>
13.1	Motivation of the V&V project	86

13.2	Scenario-based evaluation continuum overview	86
13.3	Scenario	86
13.4	Environment	87
13.5	Certification/safety evidence & argument	87
13.6	Simulation validation vs system testing	88
13.7	Scenario-based evaluation at the functional level	88
13.8	Integration with the Zenic Phase 3 ecosystem	89
13.9	Scenario preparation and retrieval	90
13.11	Euro NCAP AEB use case	91
13.12	Dynamic and static elements validation method	92
<b>14  </b>	<b>Simulation results</b>	<b>94</b>
14.1	Use case 1: AEB-O	94
14.2	Use case 2: AEB-P	95
14.3	Use case 3: EEBL	96
<b>15  </b>	<b>Discussions and conclusions</b>	<b>97</b>
<b>16  </b>	<b>Future work</b>	<b>99</b>
16.1	Advancement from a PoC to a minimum viable product	99
16.2	Further expansion	99
16.3	More applications	100
<b>17  </b>	<b>Appendix A – Taxonomy definition for UK simulation capabilities</b>	<b>101</b>
<b>18  </b>	<b>Appendix B – Existing scenario engines using OpenSCENARIO standard</b>	<b>125</b>
<b>19  </b>	<b>References</b>	<b>130</b>

## Executive summary

The scale of anticipated benefits from Connected and Automated Mobility (CAM) is significant, from substantially reducing road collisions, to improving accessibility, productivity and increased efficiency across our transport networks. Considering the required pervasiveness of CAM technologies and services to enable these benefits, ensuring their safety and security is critical.

Billions of testing miles are required for the realisation of safe and secure CAM, as well as development across a range of 'edge cases' that are either hazardous or expensive to test in the real world. This scale of development is not feasible in real world alone, but it is with the virtual world - with simulation. By utilising representative virtual models of physical test environments, flexibly and selectively running a range of test scenarios faster than real time, the challenge of testing miles and edge cases can be addressed.

However, this proposed solution is not without key assumptions and hurdles to clear. Virtual environments must be sufficiently equivalent to their physical counterpart, test scenarios must be efficiently allocated and targeted towards particular areas of system performance, and the simulation capability must be comprehensive enough to run these scenarios at the required fidelities and speeds in a flexible and seamless manner. Addressing these considerations is far from trivial.

In collaboration with Ford Motor Company, The University of Leeds, KAN Engineering and CAM Testbed UK, the UK's world-leading CAM testing and development ecosystem, this project has delivered a proof of concept (PoC) simulation demonstrator to address these considerations. This project (phase 3) builds upon existing capabilities developed as part of phase 2, utilising a distributed, System-of-Systems simulation architecture to integrate and demonstrate a novel, comprehensive and flexible approach to simulation.

To ensure real-world applicability of the project, global car manufacturer Ford have featured as the project customer. Ford have donated their Automatic Emergency Braking (AEB) system to feature as a System Under Test (SUT) for the project. Significant efficiency gains have been demonstrated across the 'V' development cycle for this SUT using the project's unique simulation approach. Considering the SUT is a production level system, the efficiencies demonstrated are a sure validation of the effectiveness and utility of the project's simulation approach. By using a 'common bus' architecture with common interfaces, simulation capabilities can be seamlessly and flexibly swapped in and out to quickly transition between different test setups, thus quickening the overall development process.

World-leading capabilities from across CAM Testbed UK and the University of Leeds have been integrated into the simulation framework, demonstrating the power of collaborative solutions to address particular CAM use cases in simulation. These capabilities span the whole V development cycle of scenario generation, simulation execution and analysis, forming a closed loop that can inform future testing.

The Smart Mobility Living Lab's novel scenario generation capabilities have been utilised, generating simulation scenarios from real-world road footage. This has ensured simulation testing is based on scenarios that are known to occur in the real world.

For the simulation execution, UTAC's simulation laboratory and the University of Leeds' Highly Immersive Kinematic Experimental Research pedestrian simulator have been integrated for real-time AEB for Pedestrian simulation between the two sites using Ford's AEB system as the SUT. Also, an Electronic Emergency Brake Light use case was demonstrated, utilising Vehicle-to-Everything (V2X) capabilities from Applus IDIADA, interacting with the capabilities at UTAC. This use case demonstrated how V2X capabilities can enhance the safety of our roads, disseminating alert messages to surrounding vehicles who can then mitigate, or even avoid, hazardous situations.

For analysis, the project has integrated HORIBA MIRA's Oracle capabilities. This has enabled holistic assessment of testing through the aggregation of performance and context metrics, giving a tangible, score-based assessment of the SUT that can inform which future scenarios should be tested.

The project has also integrated an end-to-end Verification and Validation pipeline into the architecture. This pipeline, created in a separate project led by WMG, is scalable, efficient and provides assurance for the safety and performance of an SUT and the simulation itself. Euro NCAP scenario generation was performed as part of this integration, utilising WMG's Scenario Description Language and also the Safety Pool™ scenario database.

The project generated several recommendations for future work. A summary of them is as such:

- **PoC to a minimum viable product** - The project's outcome confirmed the technical validity and robustness of the proposed solution. However, further work is required to upgrade the solution from a PoC to a minimum viable product with a higher Technology Readiness Level (TRL).
- **Capability and user expansion** - The solution has the capability and capacity to be further enhanced and expanded to include additional simulation capabilities, simulation service providers and customers from a range of international locations.
- **Confirming development efficiencies** - The validity and effectiveness of the solution to reduce the time and cost to market for Connected and Automated Vehicle (CAV) products and systems should be further confirmed.
- **Further real-world applications** - The solution should be further utilised and examined in real-world development applications. Some possible applications include undertaking a complete V development cycle for an automated vehicle or system and examining more SUTs across various TRL levels and / or autonomy levels.
- **Simulation for certification and assurance** - The robustness and reliability of the solution as a certification and assurance tool for CAVs, as well as the accuracy and validity of the simulation results, need to be thoroughly studied and confirmed.
- **Enhancing portability between testbeds** - An appetite for demonstrating and enabling more 'portable' customer journeys between testbeds, removing friction and improving efficiency as customers move testing from one testbed to another, was expressed by project

partners and stakeholders during the project. Future work could explore how this PoC solution could further accommodate portability (demonstrated in phase 2), with comparative evaluations of alternative technical solutions.

Fully realising the CAM opportunities at hand, through simulation, is beyond any one area of expertise or technology. Collaboration is crucial to realise the trusted, comprehensive and flexible capabilities required. This project has demonstrated how this is possible, with each partner playing a unique, instrumental role in a larger, collaborative delivery. It is precisely this collaborative approach that will drive the future success of CAM simulation, providing seamless and trusted testing for a safe and secure CAM future.

# 1 | Introduction

The critical role of simulation in the design, development, test, and validation of Connected and Automated Vehicles (CAV) has been widely acknowledged by the automotive industry. A wide range of (all-inclusive) advanced simulation software, tools and solutions are available in the market, ranging from free, open-source software and packages to expensive commercial-off-the-shelf (COTS) products and various simulators and Hardware-in-the-Loop (HiL) rigs from a wide range of global vendors. However, increasing the complexity and the number of functionalities of CAV systems and products implies running hundreds of thousands of scenario variations at different levels of fidelity and realism. It becomes apparent that there is no single simulation tool that can run large-scale complex enough simulations with adequate flexibility and comprehensiveness, especially for real-time industrial-level development applications, Verification and Validation (V&V), and certifications tasks.

Interoperable and distributed simulation tools and techniques seem to be a promising solution to address current Modelling and Simulation (M&S) challenges. This implies a new systemic approach to simulation, but the required technologies for interoperable simulation solutions within the automotive industry are still at the early development stages. It is challenging to integrate, run, and manage multiple simulation tools within a distributed co-simulation environment. The development of a standardised approach has not yet been perfected, and the ability to streamline and optimise this process is a technological gap in the automotive simulation domain.

As a continuation of the work carried forward from the previous Phase (phase 2), (Soltani and Reed, 2021), this project aimed to expand the integrated capabilities across the UK simulation eco-system further. A proof of concept (PoC) interoperable simulation system across CAM Testbed UK and beyond has been developed in this project to demonstrate the state-of-the-art technical solutions and tools for a distributed co-simulation system. The proposed simulation solution is based on a novel systemic, DDS-based, data-centric approach to CAV simulation. Importantly, the project addresses the challenges of Verification and Validation (V&V) of a System Under Test (SUT) for automated driving applications by proposing an end-to-end, closed-loop approach to the simulation, integrated with a large-scale scenario database.

The world-leading achievements of the project were based on close collaboration between the project partners, including Zenzic, KAN Engineering, UTAC, Applus IDIADA, the Smart Mobility Living Lab (SMLL), HORIBA MIRA, WMG and the University of Leeds. The project demonstrated new interoperable modelling and simulation capabilities beyond the capacity of any single partner's facilities. By having Ford Motor Company on board, as the project customer, the project team aimed to develop a simulation solution aligned with the automotive industry's needs and development process. To achieve the numerous technological breakthroughs, the partners worked collaboratively, exchanging background and foreground intellectual property.

To facilitate the implementation of simulation interoperability, a novel simulation manager tool (KAN Simulation Manager and KAN Scenario Engine) and a cloud-based connectivity solution have been developed by KAN Engineering, and its application has been examined by the project partners. The proposed interoperable distributed simulation platform enables seamless, flexible

and cost-effective access to a diverse range of simulation capabilities located over a wide range of geographical locations across the UK, by offering a cloud-based connectivity solution, easy-to-implement interfaces and a modular architecture.

This report is organised as follows. Section 2 points to a taxonomy definition (in Appendix A) for the simulation capabilities as part of the project deliverable. Section 3 presents an overview of the project background and introduces the testbed partners, the proposed systemic approach to the simulation, and the project use cases. Section 4 presents a technical description of the cloud-based connectivity solution implemented in this project and a discussion about latency. Section 5 explains the technical specifications of KAN Simulation Manager as the core software element of the distributed interoperable simulation architecture, which has been used in this project. Section 6 explains the technical specifications of KAN Scenario Engine as an add-on module to KAN Simulation Manager to enable an end-to-end execution of the simulation system based on ASAM open standard scenario files. Section 7 provides details about the system integration with the project partners and the use case setups. Sections 8 to 13 have been written by the project partners (UTAC, Applus IDIADA, University of Leeds, SMLL, HORIBA MIRA and WMG) and explain their simulation capabilities and their contributions to the project outcomes. An overview of the simulation results is presented in Section 14. Section 15 summarises the project outcome and conclusions. Finally, some topics for future work, as the possible next steps of this project, are proposed in Section 16.

## 2 | **Taxonomy definition for UK simulation capabilities**

As part of the project deliverable, a list of simulation related topics has been selected, and the associated definitions from various sources have been extracted. The list of topics and their definitions have been reviewed and confirmed by the project partners. The taxonomy definition for UK simulation capabilities is indicated in Appendix A of this report.

## 3 | Interoperable Simulation PoC, Phase 3

### 3.1 Project objectives

The project objective was to develop a PoC interoperable simulation system to demonstrate the state-of-the-art technical solutions for integrated capabilities across the UK simulation ecosystem. A unique offer was developed by leveraging the UK's simulation strength, while establishing a common and flexible integration framework. The project aimed to develop a solution aligned with customer needs and their development processes.

The goal of this project was to provide a technical solution for interoperable simulation across CAM Testbed UK and beyond. The project was considered as a continuation of the previous phase (Phase 2) to expand and enhance further the UK's capabilities in simulation interoperability. The project has demonstrated the feasibility of the proposed technical solution by successfully delivering a PoC demonstrator to the customer by utilising the various simulation capabilities involved. It showcased how existing simulation capabilities can be made interoperable such that CAM Testbed UK can offer new and valuable propositions for research, development and testing of CAV, which is beyond the capacity of any single CAM Testbed UK facility.

### 3.2 Project partners

The world-leading achievements and successful delivery of the project was a direct result of close collaboration between the following project partners:

**Project lead (Zenic):** the project owner and leader, responsible for the overall management, coordination, and delivery of the project.

**System Integrator (KAN Engineering):** the project technical lead and system integrator, responsible for the technical delivery of the project. KAN Engineering has worked closely with Zenic and a wide range of stakeholders (including the project customer, partners, subcontractors, and contributors) to ensure the successful results of the project.

**Customer (Ford):** The project has greatly benefited from the involvement of Ford Motor Company, as the project customer, who brought Automatic Emergency Braking (AEB), as a high Technology Readiness Level (TRL) System Under Test (SUT), to be virtually tested in this project. The project customer has utilised the UK PoC distributed simulation facilities for the virtual testing of their SUT. The successful delivery of the project was a direct result of a close collaboration between the project customer and partners, at the technical and managerial levels.

**Project partners:** The leading organisations from CAM Testbed UK and beyond, who brought some of their world-leading simulation capabilities into this project:

- Applus IDIADA (CAVWAY testbed)
- HORIBA MIRA
- Smart Mobility Living Lab (SMLL)
- UTAC

- University of Leeds
- WMG (Midlands Future Mobility testbed)

The project partners have worked closely with the project lead (Zenzic), system integrator (KAN Engineering) and project customer (Ford) on the development and implementation of the integrated distributed simulation system to test and validate the customer's SUT. A detailed description of the project partner capabilities involved and their technical contributions to the project delivery, are explained in Sections 8 to 13 of this report.

**Project Contributors:** The following global companies kindly contributed to the project delivery by enabling free access to their simulation tools, products, and services to be utilised for the delivery of this PoC project and provided technical support:

- RTI<sup>1</sup>: access to their DDS middleware software product (RTI Connex) and technical support on the design and implementation of APIs.
- Amazon Web Service (AWS)<sup>2</sup>: access to their web services including cloud-based connectivity, execution, data storage and management, user interface and data authentication.
- IPG<sup>3</sup>: access to their own designed Vehicle-in-the-Loop (ViL) platform and technical support on Model-in-the-Loop (MiL), Software-in-the-Loop (SiL) and ViL simulation using IPG CarMaker software.

### **Applus IDIADA**

Applus IDIADA is headquartered in Spain but with engineering and test offices in the UK. Our own CAM Testbed UK proving ground, CAVWAY, is under construction near Oxford.

We provide services in automotive engineering development of many kinds. For this simulation project our electronics and software offering was key, covering system engineering, safety and cybersecurity analysis, implementation and testing. We also sell electronic control units of our own design.

However, we are better known for our large and busy proving ground near Barcelona, which has specific areas for CAV testing and more in the pipeline, and a high-resolution digital twin with an immersive dynamic simulator. With those facilities we host and optionally support test work by OEMs. In parallel, we offer certification services both for regulatory homologation and consumer testing (Euro NCAP). We also have a proving ground in China.

### **HORIBA MIRA**

HORIBA MIRA is a global provider of pioneering engineering, research and test services to the automotive, defence, aerospace and rail sectors. HORIBA MIRA works in close collaboration with global vehicle manufacturers and their Tier 1 & 2 supply chain, providing comprehensive support

---

<sup>1</sup> <https://www.rti.com/en/>

<sup>2</sup> <https://aws.amazon.com/>

<sup>3</sup> <https://ipg-automotive.com/en/>

ranging from individual product tests to turnkey engineering design, development and build programmes.

With over 70 years of experience in developing some of the world's most iconic vehicles, HORIBA MIRA's engineers utilise the latest test facilities and simulation tools to make vehicles and journeys safer, cleaner and smarter. Through their major test facilities, specialised proving ground, engineering experience and international presence, HORIBA MIRA can achieve their vision of positively influencing every journey in the world by making them safer, cleaner and smarter.

### **Smart Mobility Living Lab (SMLL)**

Smart Mobility Living Lab (SMLL) is a London-based real-world connected environment for testing and developing future transport and mobility solutions. It is the world's most advanced urban testbed of its kind with the sole purpose of accelerating the creation of mobility solutions that are clean, efficient, safe, reliable and convenient for everyone using public and private roads in London, to develop and validate new mobility and transport technologies.

The locations of the Royal Borough of Greenwich and Queen Elizabeth Olympic Park in Stratford provide a complex uncontrolled testing environment, interacting with live traffic and other road users. The testbed is designed to demonstrate and evaluate the use, performance, environmental impact, safety and benefits of Connected and Automated Mobility (CAM) technology and future transport services.

**Figure 3-1: SMLL locations on London map**



Source: Author Generated (SMLL)

On a practical level, SMLL services are based around the three integrated pillars of Test, Simulate and Innovate, with a full range of transport technical consultancy provided by TRL, DG Cities and the London Legacy Development Corporation (LLDC). Alongside testing and trialling, SMLL facilitates a community of large corporations and SMEs with an aim to stimulate innovative collaborative R&D projects within the future mobility sectors. SMLL also provides a simulated environment to complement its real-world testing so that customers can begin to extend testing within a virtual environment.

SMLL recognises that simulation will play a large role in testing, developing, validating and ultimately certifying ADAS and ADS systems and therefore has invested in a digital twin and simulation capabilities that it can offer its customers to support the real-world testing.

### **UTAC**

UTAC is a leading group in the field of development and validation testing, automotive homologation and new technologies related to the autonomous, connected and electric vehicle, provides vehicle testing and validation services and equipment to the automotive, transport, tyre, petrochemical and defence sectors.

The group is active in the fields of testing, approval and regulation, special vehicle design, conception and manufacture of test systems, training, consulting, audit and certification, technical control, standardisation and events. UTAC is the only official Euro NCAP test centre in France (accredited since 2001) and has a unique position in Europe thanks to its ISO 17025 accredited test laboratories.

The group has 8 test centres in France, the UK, Finland and Morocco, test laboratories in the USA and subsidiaries in Germany, Russia, China and Japan. UTAC currently employs around 1,280 people at its various sites.

UTAC is the French technical service for homologation of vehicles regarding EU regulations. UTAC also acts as a Technical Service for the United Kingdom Vehicle Certification Agency (VCA) and the Netherlands RDW. UTAC in the UK also act as an accredited test laboratory for the Taiwanese vehicle safety certification centre (VSCC) and the Japanese national traffic safety and environment laboratory (NTSEL).

### **University of Leeds**

The Institute for Transport Studies at the University of Leeds is a leading research centre worldwide, ranking 18th in the Shanghai Ranking of Academic Subject of Transport in 2021. It is also the largest department of its kind in the UK and has a world leading set of highly immersive facilities shaping the VirtuoCity programme, linking innovation expertise from across the University of Leeds, and also to a wide range of external partners and stakeholders.

VirtuoCity has three main laboratories, the largest academic research driving simulator in the UK, using a 5m x-y motion table and a hexapod dome that houses a full vehicle cab, a Truck Simulator, and the Highly Immersive Kinematic Experimental Research (HIKER), the world's largest high-resolution cave based pedestrian simulator.

In order to provide the Interoperable Simulator with a Pedestrian-in-the-Loop (PiL) laboratory we have opened the HIKER to CAM Testbed UK, so that a pedestrian in the HIKER can interact in a safe manner and in real time with a vehicle driving on a remote proving ground. These interactions represent a critical challenge for automated vehicles that need to negotiate their right of way with pedestrians in urban environments.

## **WMG**

WMG is a department of the University of Warwick, one of the UK's leading research-led universities. WMG provides research, education and knowledge transfer expertise in engineering, manufacturing and technology. With an established, peer-recognised team of leading academics working alongside engineers with strong industrial background, WMG prides itself for having exceptional links with industry, with a critical vision to drive innovation from R&D to commercial impact, and works successfully with over 1000 companies on collaborative applied research driven by future market needs. The WMG centre for High Value Manufacturing (HVM) Catapult is focused on the global challenge of Low Carbon Mobility, driven by strategic priorities in the automotive, commercial, off-road rail and marine transport sectors. The centre of excellence for connected and autonomous vehicles research at WMG fulfils our ambitions of shaping the future of transport mobility in the UK. We are home to the Energy Innovation Centre (EIC), National Automotive Innovation Centre (NAIC) and Advanced Propulsion Centre (APC).

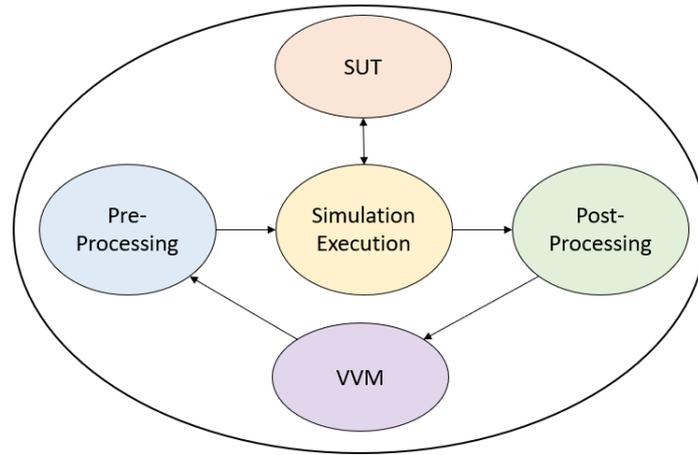
WMG have been successfully leading and participating in several national and international collaborations such as OmniCAV, Midlands Future Mobility, as well as working on various CCAV and IUK funded CR&D projects. Our unique international position ensures that the concepts developed as part of the Zenic V&V project are aligned to international standardisation and regulatory discussions. Our project team members represent the UK on the UNECE FRAV and VMAD sub-groups. Furthermore, project team members serve as project lead in various ASAM standards and ISO standards groups.

### **3.3 Simulation approach**

Being the continuation of the work carried forward from the Phase 2 Interoperable Simulation PoC demonstrator for CAM Testbed UK (Soltani and Reed, 2021), the proposed simulation solution is based on a systemic, data-centric, and end-to-end approach to CAV simulation.

As discussed in (Soltani and Reed, 2021), an end-to-end approach for simulation-based testing of a CAV system or product (SUT) must incorporate three main sub-systems (components): pre-processing, execution, and post-processing. Verification and Validation Methodologies (VVM) can be implemented on top of this basic architecture to close the loop in the V cycle of automotive development process, as depicted in Figure 3-2.

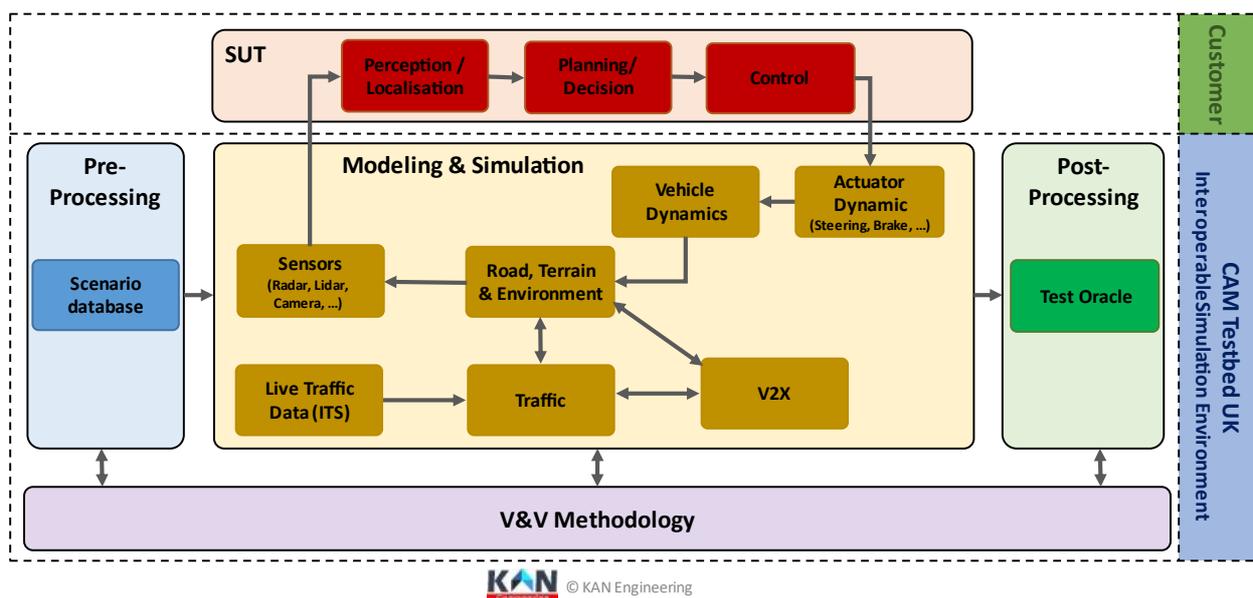
**Figure 3-2: Closed-loop simulation environment as a System-Of-Systems**



Source: Author Generated (KAN Engineering)

Our proposed approach is systemic since the components of the simulation can be decomposed and classified using the principle of System-of-Systems (SoS). With this approach, the SUT can be either a basic component of the simulation system or a complex combination of various sub-systems. In the highest level of abstraction, this classification involves the environment, traffic, vehicle, and sensor models interfacing with the SUT algorithms (such as perception, planning and control), as presented in Figure: 3-3. The architecture has been expanded in this phase of the project, to include integration with scenario databases (pre-processing), test oracles (post-processing), V2X connectivity, PiL, DiL and ViL capabilities for the simulation execution as well as the real test track. The system provides feedback from the test oracle to the scenario generation for iterative closed-loop simulation (V&V).

**Figure 3-3: Simulation Approach as a System-of-Systems**

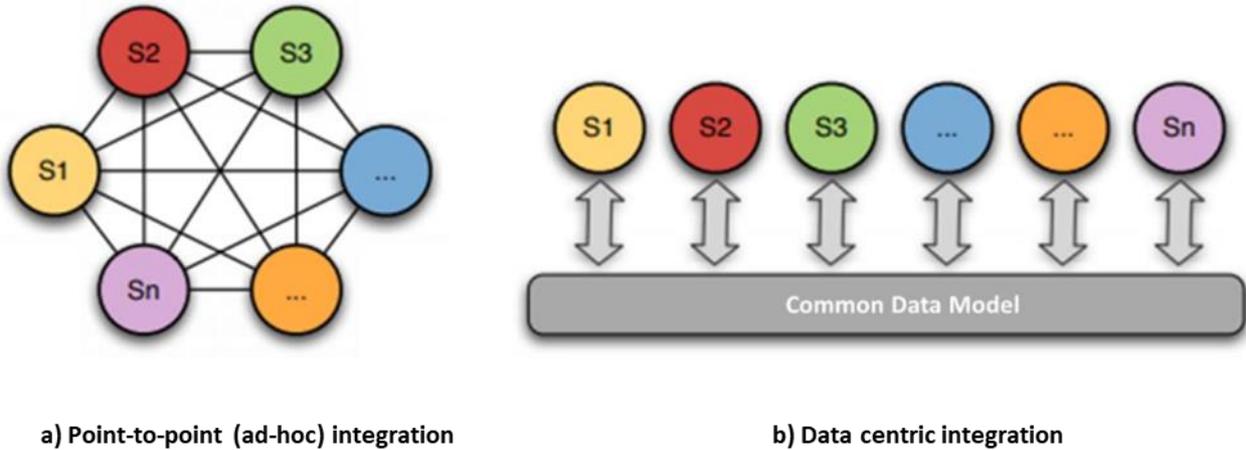


 © KAN Engineering

Source: Author Generated (© KAN Engineering)

To interface and orchestrate all the sub-systems within the simulation system, a simulation manager software has been developed by KAN Engineering using a DDS data bus middleware. Having a data bus architecture means that the communication between the simulation sub-system components are data-centric. The existence of a generalised data model for the representation of the simulation sub-systems enables seamless and expandable integration between any number of sub-systems using an API integration at each component, bypassing the need for complex ad-hoc architectures, as depicted in Figure 3-4. This enables the simulation system to outperform traditional point-to-point based integration methods on scalability, efficiency, and flexibility (Soltani and Reed, 2021).

**Figure 3-4: Point-to-point architecture vs data bus architecture**

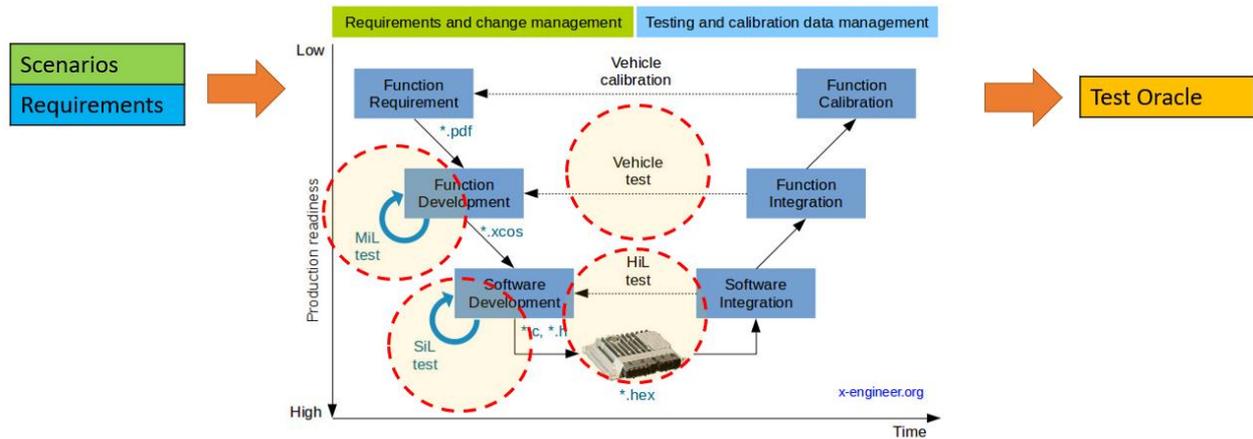


Source: Adapted by Author (KAN Engineering)

The proposed solution enables an end-to-end simulation due to the simulation system's systemic approach and data-centric implementation. The flexibility of the architecture permits the components of the simulation to be added and removed with ease, allowing the user to adhere to the automotive V model during each phase of V cycle product development (Figure 3-5), including MiL, SiL, HiL, DiL, ViL and real test track testing. In addition, the capability of the simulation manager to run on a closed-loop simulation with the test oracle providing new parameter values on each iteration, enables the user to identify edge cases and improve SUT performance before carrying out real-world testing.

**Figure 3-5: V model of development as an end-to-end solution**

# V Development Cycle



Source: (X-Engineer, no date), Modified by Author (KAN Engineering)

## 3.4 Capabilities involved

In this phase of the project, the same simulation architecture, as developed in the previous phase of the project (phase 2) has been used, but it has been expanded to include further simulation capabilities from the project partners, to demonstrate the flexibility and expandability of the proposed interoperable simulation architecture. In addition to what has been integrated in phase 2 of this project (Soltani and Reed, 2021), the following simulation capabilities have also been integrated into the simulation architecture, KAN Simulation Manager APIs and DDS middleware to form an integrated, distributed simulation system:

- **UTAC, driving simulator (DiL):** The static driving simulator from VI-Grade integrated with the IPG vehicle dynamics simulation through Concurrent Simulation workbench. More details about UTAC’s driving simulator is provided in Section 8 of this report.
- **UTAC, real test track:** The real data from the vehicle driving on the UTAC test track (Mile Straight) fed into the simulation environment using UTAC’s 4G/5G connectivity. The integration between Millbrook’s 4G/5G network and the simulation setup has been performed by employing a DDS data bridge developed by KAN Engineering. More details about UTAC’s test tracks and private 4G/5G network are provided in Section 8 of this report.
- **Applus IDIADA, Vehicle-to-Everything (V2X) HiL rig:** including the communication hardware and the associated software components designed by IDIADA. This incorporates live radio communications into an interoperable simulation system at the HiL level. More details about IDIADA’s V2X HiL rig are provided in Section 9 of this report.
- **University of Leeds, PiL simulator (HIKER):** The inclusion of real pedestrians into the interoperable simulation environment has been implemented by including the University of Leeds’ pedestrian simulator (so-called HIKER: Highly Immersive Kinematic Experimental

Research), the world's largest high-resolution cave based pedestrian simulator. More details about the University of Leeds' PiL Simulator are provided in Section 10 of this report.

- **IPG/Ford, ViL simulator:** Ford and IPG provided their ViL platform to be used in this project. Ford and IPG automotive are collaborating under project ViVID to create a ViL pilot to demonstrate how this technology delivers a cost-effective solution that enables ADAS and Automated Driving System (ADS) development and validation in multiple complex or dangerous scenarios. With this ViL setup, various ADAS features (including AEB-P) could be assessed in the test track without the requirement of the physical (soft) target being involved. The pilot vehicle is a Ford Transit, and the following hardware is installed:
  - XPack4 HiL system (robust housing)
  - ADAS ECU: Contains ADAS logic of the features under assessment
  - Other vehicle ECUs (i.e. ABS) also included
  - Camera Box: Sealed box containing Ford camera facing a high resolution monitor playing IPG movie
  - Castle Box: Sealed box containing two opposed arrays of parking sensors, one transmitting the virtual environment from CarMaker and the other transmitting the received environment to the ADAS ECU
- **SMLL, scenario generation:** A novel methodology to generate scenarios from real road scenes has been developed by SMLL and utilised in this project. More details about SMLL's scenario generation setup are provided in Section 11 of this report.
- **HORIBA MIRA, test oracle:** HORIBA MIRA's test oracle has been utilised in this project to assess the outcome of the simulation execution and study the behaviour exhibited when a SUT is confronted by a given driving situation. More details about HORIBA MIRA's test oracle are provided in Section 12 of this report.
- **WMG, V&V pipeline:** An end-to-end, scenario-based V&V pipeline has been developed by WMG. This pipeline has been utilised in this project and integrated with the distributed simulation architecture to form an end-to-end closed-loop simulation setup. More details about the V&V pipeline and WMG's contribution to the project are provided in Section 13 of this report.

### 3.5 Project use cases

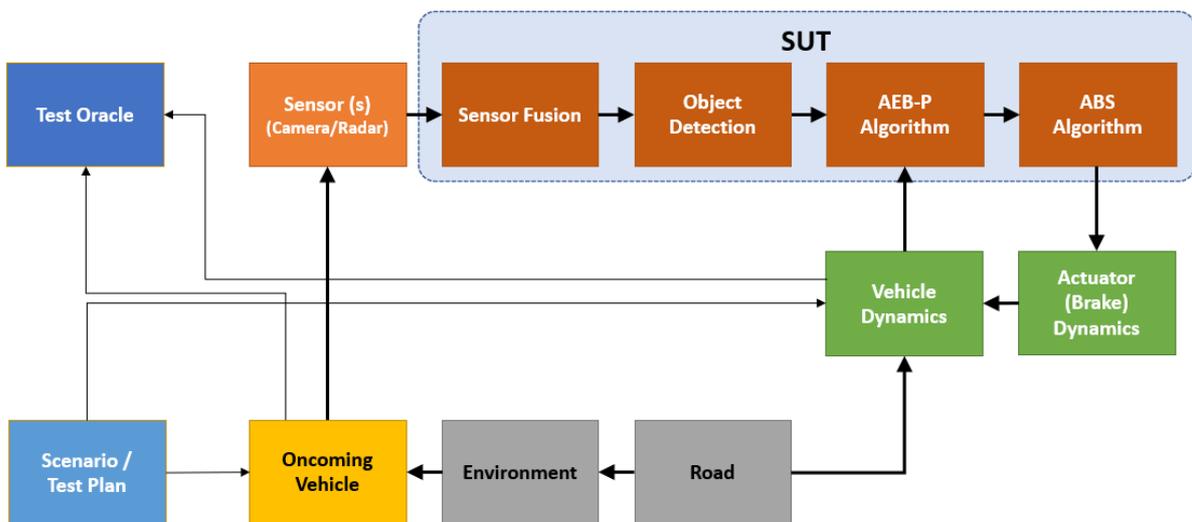
Various NCAP scenarios related to ADAS systems have been defined as the project use cases to demonstrate interoperable simulation capabilities among the testbed partners. The SUTs were Automatic Emergency Braking Oncoming vehicle (AEB-O), Automatic Emergency Braking Pedestrian (AEB-P), and Electronic Emergency Brake Light (EEBL) systems, which are categorised as part of vehicle collision mitigation systems, a sub-set of Advanced Driver Assistance Systems (ADAS). (ISO, 2013, 2017; SAE, 2017)

**Use case 1: AEB-O**

This use case involves the evaluation of AEB capability in a variety of collision scenarios. The collision scenarios are defined considering the ego vehicle with AEB capabilities driving straight ahead and approaching another vehicle moving in the same lane or with significant lateral overlap. The ability to avoid, or reduce the severity of, a collision is the most crucial aspect of a vehicle's behaviour in certain collision scenarios. (BSI, 2021)

A systemic approach is adopted by considering the components, as shown in Figure 3-6. the SUT is the AEB functionality which includes features such as sensor fusion, object detection, AEB and ABS (Anti-lock Braking System) algorithms that execute perception, planning and control tasks. The simulation environment is perceived by the sensor system (camera), and the sensor data is input to the SUT (AEB functionality). The algorithms will send the processed signals to the actuator (braking) system of the test vehicle. The environment contains both the static elements of the environment (such as the road logic/map, which is defined either using OpenDRIVE or IPG Carmaker road formats) and the dynamic elements of the environment, such as the traffic actors (vehicles).

**Figure 3-6: A systemic decomposition of the Automatic Emergency Braking – Oncoming (AEB-O) system**



Source: Author generated (KAN Engineering)

MiL, SiL and DiL testing is carried out to validate the AEB functionality against the developed scenarios. MiL testing is carried out by considering the model provided by the customer (Ford) which has been designed to implement the AEB function in Simulink, and is integrated with the ego vehicle model in IPG Carmaker using Carmaker for Simulink (Carmaker4SL). SiL testing is carried out by generating C code from the Simulink model and integrating it into an IPG Carmaker executable. The capability of the SiL is extended to DiL, and the C code generated is integrated with the driving simulator at UTAC with the aid of KAN-API, which enabled the injection of driving simulator commands (throttle, brake, steering, gear) and traffic data

(positions, velocities, accelerations). The simulation setup details for MiL, SiL and DiL are provided in Section 7 of this report.

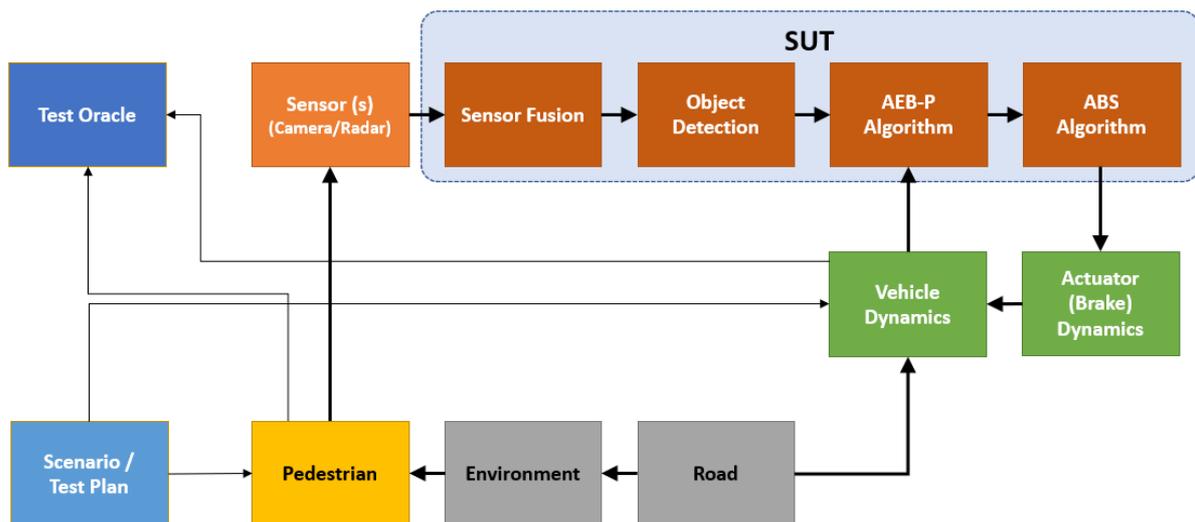
For MiL, SiL and DiL a scenario-based testing approach is adopted. The scenarios have been developed in OpenSCENARIO format, which includes car-to-car rear moving and car-to-car rear braking scenarios. In these scenarios, both test and traffic vehicles are moving in the same direction. The other scenario defined involves a traffic vehicle oncoming into the test vehicle lane from the opposite direction.

**Use case 2: AEB-P**

This use case involves minimising the severity of collision scenarios in which a pedestrian crosses the path of the test vehicle. Based on the scenarios defined with the pedestrian and the ability of the test vehicle, AEB functionality is evaluated. For this use case, the SUT is the AEB functionality to be evaluated with the pedestrian model (Euro NCAP, 2020).

A systemic approach is adopted by considering the components as shown in Figure 3-7, similar to the AEB-O use case, with the only difference being that the traffic vehicle is replaced by a pedestrian. The SUT for this use case is the AEB functionality which includes algorithms like sensor fusion, object detection, AEB and ABS algorithms which act as high-level controllers. In addition to the implementation similarities with AEB-O, the traffic actors (pedestrians) can be either virtual or real.

**Figure 3-7: A systemic decomposition of the Automatic Emergency Braking – Pedestrian (AEB-P) system**



Source: Author generated (KAN Engineering)

MiL, SiL and DiL testing is carried out to validate the AEB functionality. AEB-P MiL testing is the same as for AEB-O, except the scenarios include the pedestrian actors. For SiL testing a similar approach is adopted, as provided in the previous Section, and it is extended to show the capabilities of Pedestrian-In-the-Loop (PiL) simulation. KAN API was used to integrate with the University of Leeds pedestrian simulator (HIKER), as well as programme pedestrians from

OpenSCENARIO via the KAN Scenario Engine or IPG Carmaker. This allowed the UTAC Millbrook simulator to publish traffic actor data from the ego vehicle and scenario to the DDS data bus, as well as subscribe to the HIKER's pedestrian data. DiL testing was carried out by adding UTAC's driving simulator to the simulation setup, which enables real-life driver interaction with the other components in the simulation. The simulation setup details for MiL, SiL and DiL are provided in Section 7 of this report.

The MiL, SiL, and DiL testing scenarios include a pedestrian walking in the same direction as the test vehicle in front of its path, and another in which the pedestrian crosses directly in front of the path of the test vehicle as defined by NCAP test procedures.

### Use case 3: EEBL

**Figure 3-8: Electronic Emergency Brake Light (EEBL)**



Source: Continental<sup>4</sup>

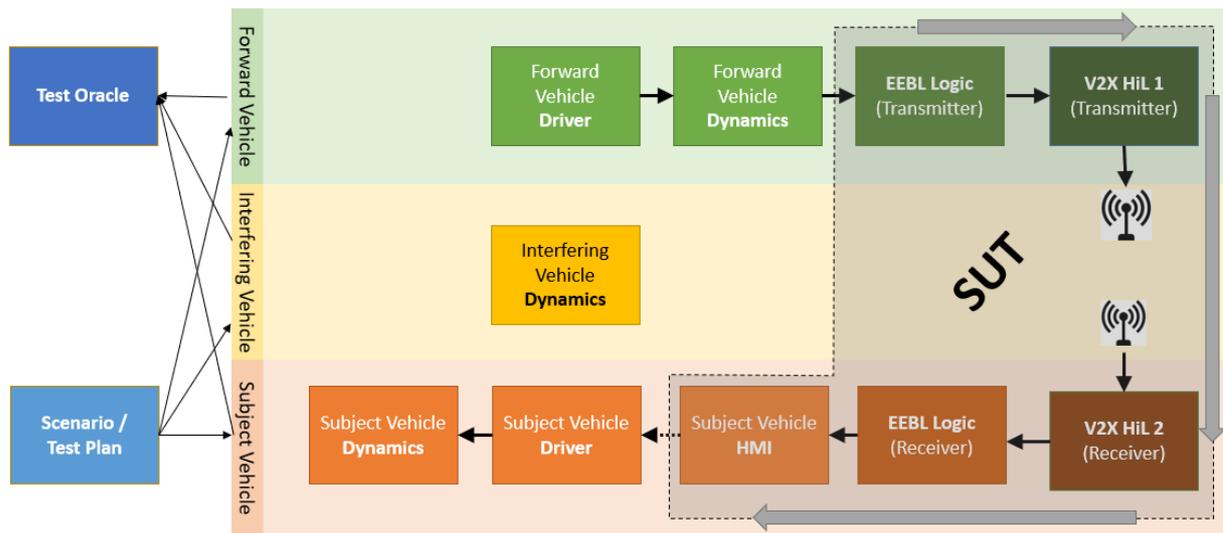
EEBL is a Vehicle to Vehicle (V2V) communication-based system. Its functionality is to alert the driver of a hard braking event ahead by a visual warning message on the HMI device of the test vehicle. Considering the driver's vision may be obstructed by other vehicles ahead or bad weather conditions, EEBL is particularly effective in alerting a driver of hazardous, unseen events ahead. Following an alert, the driver can reduce the test vehicle speed to avoid or reduce the severity of a collision. EEBL allows a vehicle (forward vehicle) to broadcast a self-generated emergency braking event to other cars in the proximity. The system under test is the test vehicle (subject vehicle) with the EEBL functionality. The EEBL system does not feature the ability to operate the vehicle to achieve the desired speeds. The driver is always responsible for the vehicle's safe functioning (BSI, 2020a).

A systemic approach is adopted by considering the EEBL system components, as shown in Figure 3-9. The SUT includes the EEBL algorithm, Subject Vehicle (SV), HMI device and V2X HiL components. The scenario-based testing is carried out by defining various scenarios on UTAC's

<sup>4</sup> <https://www.continental-automotive.com/en-gl/Passenger-Cars/Autonomous-Mobility/Functions/Cruising-Driving/Emergency-Brake-Assist/Electronic-Emergency-Brake-Light>

“mile straight” track to validate the functionality of the SUT. UTAC Millbrook provided the OpenDRIVE model of their mile straight track. Applus IDIADA enhanced it with graphic components in RoadRunner, and then exported it in Unity and CARLA formats for distribution among the other project partners. As a result, the interoperability of environmental model data (model interoperability) throughout the testbed partners was clearly demonstrated. V2X connectivity was demonstrated by integrating the EEBL systems with the V2X HiL components from IDIADA with the aid of KAN Simulation Manager.

**Figure 3-9: A systemic decomposition of the Electronic Emergency Brake Light (EEBL) system**



Source: Author generated (KAN Engineering)

MiL, SiL, DiL and ViL testing was carried out to validate the EEBL functionality. MiL testing was carried out by developing the EEBL algorithms for both transmitting and receiving vehicles using Simulink, and integrating them with the IPG default vehicle model in IPG Carmaker using Carmaker4SL at the UTAC site. For SiL testing, the EEBL algorithm was generated in C++ code and integrated with KAN HMI. The CAYWAY bridge from Applus IDIADA, acting as a gateway to the IDIADA V2X infrastructure, was also integrated with KAN API. This was further expanded in DiL testing by integrating UTAC’s driving simulator through Concurrent real-time software and hardware platform (SIMulation Workbench)<sup>5</sup> to the simulation architecture. Real vehicle testing has been performed with a ViL setup integrated with the EEBL system. This demonstrates the capability of the proposed solution to integrate the simulation environment with the test track. The simulation setup details for MiL, SiL, DiL and ViL are provided in Section 7 of this report.

The MiL, SiL and DiL testing scenarios defined for this use case include a forward vehicle (FV) moving in front of the subject vehicle (SV). The rate of FV deceleration is varied from one test

<sup>5</sup> SIMulation Workbench | Concurrent Real-Time ([concurrent-rt.com](http://concurrent-rt.com))

case to another to evaluate the functionality of the EEBL system, according to the ISO standard (BSI, 2020a).

## 4 | Cloud-based connectivity solution

When considering the simulation environment as a System of Systems (SoS), connectivity plays an essential role in implementing any distributed simulation environment, as explained in the phase 2 project report (Soltani and Reed, 2021). The performance of the data connection between various simulation sub-systems directly affects the quality, accuracy, latency and performance of a distributed simulation execution and the expected outcomes.

A distributed architecture simulation setup could be implemented in a single location using a Local Area Network (LAN) or between various geographical locations using various (Public internet and other wide-area-networks) connectivity solutions. In the previous phase of this project, the connectivity between two CAM Testbed UK partners (UTAC-Millbrook and SMLL) was established using a direct VPN connection (IP tunnelling). Although a direct VPN connection is a simple and straightforward solution, it has its limitations and challenges when the number of participants increases. As explained in Section 3, we had several partners involved in this phase of the project, and so a direct connection between the project partners was not an easy solution to implement, as it involves liaising with multiple IT teams with different IT policies. As a flexible and expandable solution, cloud-based connectivity has been proposed and implemented in this project. A cloud-based solution provides extra benefits and services to the simulation setup, including data storage and management, access level control, security, and extra processing capabilities.

A technical description of cloud-based connectivity solutions, including their benefits and challenges, including latency, is outlined in the following section.

### 4.1 Connectivity options

As a result of rapid advancements in wired and wireless communication technology, a wide range of connectivity options and solutions are available on the market, ranging from simple and inexpensive internet-based connections like Virtual Private Network (VPN)-over-Internet to more complex and expensive proprietary solutions like Software-Defined Wide Area Network (SD-WAN), Multiprotocol Label Switching (MPLS) VPN, and Virtual Private LAN Service (VPLS) (Business Network Solutions, 2021).

VPLS is an ethernet-based point-to-multipoint layer 2 VPN that allows users to link geographically distributed ethernet Local Area Network (LAN) sites over an MPLS backbone (Juniper Networks Inc, 2017).

Even though MPLS and VPLS use the same underlying technology, they are designed for completely different uses and have unique benefits. MPLS offers additional scalability and flexibility using VPN that automatically separates traffic to improve performance, whereas VPLS is suitable for companies that want complete control over their routing by incorporating LAN and WAN in a way that improves security and makes incident management simple (Exponential-e Ltd, 2022). To handle forwarding over private wide area networks, MPLS, as a scalable and protocol-independent solution, directs traffic using the quickest path depending on "labels"

instead of network addresses. MPLS greatly improves the speed of traffic, so users don't experience downtime when connected to the network (Palo Alto Networks, 2022). The public internet can be another low-cost option that companies may explore, but it comes with some drawbacks such as higher latency and lower security (Bhardwaj, no date). Public internet is suitable if the desired update rates are around 10-20 Hertz (Hz).

A comparison between various connectivity options and solutions is summarised in Table 4-1.

**Table 4-1: Various connectivity solutions and their specification**

	VPLS (Virtual Private LAN Service) Point-to-Point/Multipoint	MPLS (Multiprotocol Label Switching) Point-to-Any point	Public Internet
OSI Model Layer	Layer 2 VPN	Layer 3 VPN	Layer 3
Providers	BT Ethernet Connect, O2 Digital Links, Vodafone Dedicated Ethernet etc	BT IP Connect, Vodafone IP-VPN	All Internet service providers
Latency/Jitter	Deterministic guaranteed low latency	Deterministic guaranteed low latency	Prone to more jitter as it uses congested internet
Security	Secure and private, encryption is optional.	Secure and private, encryption is optional.	Encryption is recommended
Future expansion/flexibility	More suitable for fixed sites	Can connect to points across the globe	Can connect to points across the globe
Control	More control over IP architecture, can deploy multicast capabilities, VLAN etc	Managed by service provides	Control over routing
QoS (Quality of Service) Prioritizing traffic	Supports QoS features	Supports QoS features	Not available
Deployment time	60-70 working days	60-70 working days	Depends on time required to configure the endpoints
Cost	High	High	Low

Source: Author Generated (KAN Engineering)

## 4.2 Implementation

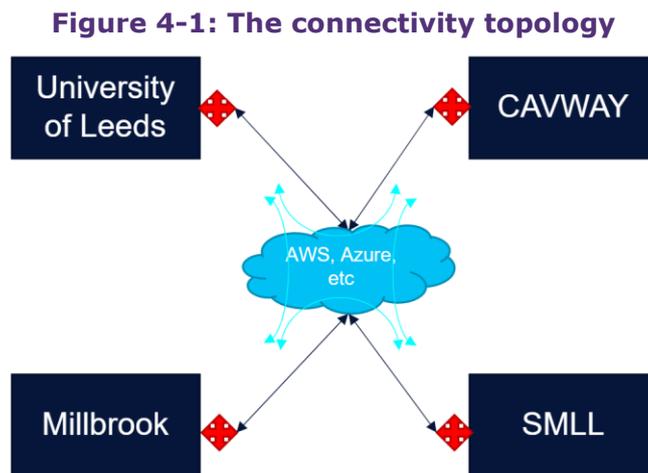
The scaling of connectivity across several sites must be seamless. The benefits of cloud access are larger than ever before, and the ability to extend simulation infrastructure to the cloud is crucial. In addition to that mentioned above, the implemented cloud-based methods provided benefits to the simulation system for data storage management, recording and post-processing.

The public internet was chosen as the connectivity solution for this PoC project demonstrator. It is a cost-effective and easy to implement solution for the project partners to join.

## 4.3 Connections

A network topology is the arrangement with which computer systems or network devices are connected to each other. Topologies define both the physical and logical aspects of the network. Both logical and physical topologies could be the same or different in the same network (Tutorialspoint, 2021).

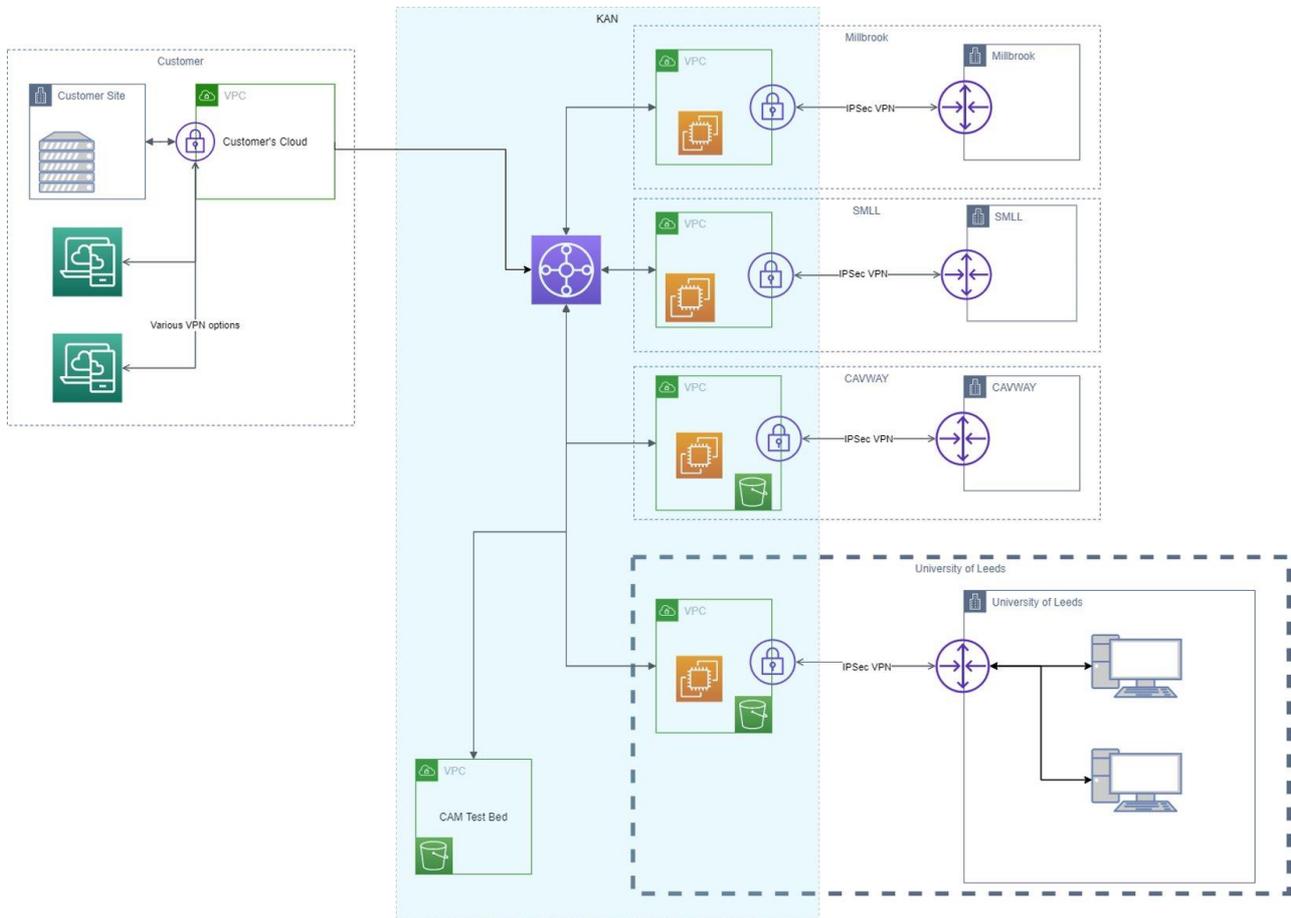
A star topology for the connectivity between the testbed partners has been used in this project. Similar to the configuration used in Phase 2, an IPsec Tunnel was established from each site to a cloud service (only the necessary computers from each site), forming a virtual private network enabling connectivity between each site via the cloud. The hosts: University of Leeds, Applus IDIADA, UTAC (Millbrook) and SMLL were connected to the cloud (AWS) using a point-to-point connection.



Source: Author Generated (KAN Engineering)

The partners had a choice between two connection options to remotely connect their on-premises networks to AWS: either VPN or AWS Direct Connect. A VPN-over-Internet solution was used in this phase with all the partners.

**Figure 4-2: The cloud-based connectivity between the customer and testbeds via AWS**



Source: Author Generated (© KAN Engineering)

The schematic diagram of the cloud-based connectivity between the customer and testbeds (UTAC-Millbrook, Applus IDIADA, University of Leeds, SMLL) via AWS is depicted in Fig xx. A dedicated Amazon Virtual Private Cloud (VPC) was been created for each testbed, enabling the option to isolate and control the data flow between the testbeds. Each of the sites had a customer gateway device/firewall, which was used to create an IPsec tunnel to the AWS Cloud. The data was sent or received through an IPsec tunnel to/from the AWS VPC. An AWS Elastic Compute Cloud (EC2<sup>6</sup>) instance was in each VPC, hosting the data routing module which routes the relevant data to relevant testbeds depending on the use case. All the VPCs were connected to each other with AWS VPC peering. However an AWS transit gateway could be used for connectivity in the future if a greater number of partners were to be involved.

Customer engineers from remote locations can access the simulation telemetry by joining their network to the transit gateway to monitor the simulation results in a near real-time manner.

<sup>6</sup> <https://aws.amazon.com/ec2/>

Also, it enables them to view simulation test configuration and parameters. Recorded results stored in the cloud could be accessed or even further analysed within the cloud resources.

## 4.4 Future solutions

The limitation of using VPN-over-Internet is the uncertainty of latency and jitter, which may pose some degradation on the quality of the simulation results if we run geographically distributed co-simulation at hundreds of Hertz. For requirements above 50Hz, VPLS and MPLS-based connections might be the preferred technical solution.

Furthermore, a hybrid strategy could be implemented to make the connectivity more flexible and efficient. In this solution, we can use MPLS, AWS Direct Connect, Azure Express Route or similar connections at critical links and the public Internet-based VPN for the non-critical links. This solution is technically feasible and cost-effective.

## 4.5 System latency

In any simulation or system of simulators, latency is a major concern. Latency issues have become more noticeable and problematic as network simulations across large distances have grown. Lower network latency allows for closer real-time access with shorter delay times. Lower latency is better as it represents the delay between taking action and receiving the result and ensures a stable and consistent connection, reducing connection loss, lags, and buffers (Dey, 2021).

In a distributed system, the latency can be considered either as the network latency or end-to-end latency. The network latency is the round trip delay between the client and the server, known as "ping". The system latency is the duration of the entire end-to-end measurement. The time it takes for a system to send a request and receive a response from a server is measured as Round Trip Time (RTT). One-way latency is the length of time spent by a packet from one point to another across the network. The total RTT is the sum of the one-way latency in both the uplink and downlink directions. It is influenced by a few key components of the network:

- **Distance** - The time to get the signal back is directly proportional to the distance between client and server.
- **Transmission medium** - The medium through which a signal is routed. It could be copper cable, optical fibre, wireless, or satellite, all of which are very different and can have a significant impact on RTT.
- **The number of network hops** - The time it takes for intermediate routers or servers to process a signal increases RTT. The RTT increases as the number of hops a signal must travel increases, resulting in increased latency.
- **Traffic levels** - When a network is overloaded with traffic, RTT often increases. Low traffic times, on the other hand, can result in a shorter RTT (Demian, 2021).

### Latency measurement

Simulation systems rely primarily on high transaction rate operations, such as CPU per second cycles, memory and storage, read/write requests and server requests, so they are most

vulnerable to latency. The increased uncertainty for service delivery in a cloud environment is due to the larger and less predictable workload.

When sending data from one site to another, such as UTAC to the University of Leeds, some of the factors that can contribute to latency are queuing and processing at routers, gateways and bridges between the source and destination nodes, transcoding delays caused by the VPN-over-Internet connectivity and propagation and transmission delays due to the speed of the communications link.

RTI PerfTest was used in this project to determine the latency by running a combined latency and throughput test application. The test measured the loaded latency at a given throughput level. It helped in areas such as publishing configuration, sample size, subscribing configuration and throughput of the network.

### **A discussion about latency in simulation systems**

As explained before, latency is a major concern in any simulation system, regardless of being centralised or distributed. This becomes more critical when we want to run the simulation in hard real-time or faster than real-time, especially for SUTs that rely on very fast response time of their control systems or high volumes of data. Some common examples are vehicle dynamics actuation systems which require a few milliseconds (or less) response time, or perception or sensor fusion systems that require high volumes of raw data from CAV sensors (such as cameras or Lidar).

Generally speaking, the overall latency in any distributed simulation system could be caused by three main sources:

1. the execution of the individual simulation tools themselves,
2. the network between simulation systems, required to connect individual simulation tools together to form a distributed co-simulation system,
3. the required middleware, APIs and simulation manager software, to form and manage a distributed simulation architecture.

Addressing the first type of latency (individual simulation tools) is a known subject within the simulation domain, and there are several technical solutions available to address it, ranging from more powerful hardware processors (CPU, GPU, RAM, ...) and parallel processing, to faster peripheral and I/O connections and more optimised software codes.

The second source of latency is the networking between the simulation tools, which (to some extent) is applicable only to distributed simulation systems. In the case of LAN, the additional latencies to the simulation system because of networking could be as low as a few microseconds, thanks to the available high-speed switching and (fiber-optic) networking technologies. The main concerns arise when we network several simulation tools at dispersed geographical locations and very long distances.

One solution to mitigate the networking latency issue is to use more advanced (high bandwidth, low latency) connectivity solutions such as MPLS, or AWS Direct Connect. In these cases, the

networking latency could be as low as a few milliseconds, which is acceptable in most (non-time-critical) use cases. In addition, there are some technical solutions that have been developed in recent years, such as the ones proposed in (Vogel *et al.*, 2019) and (Venu and Joe, 2014) to mitigate and/or compensate the networking latencies in a distributed simulation system. Working on further technical solutions to alleviate the networking latency in a distributed simulation system could be an interesting topic for further research and development to be undertaken by the leading UK universities and research institutions.

The third source of latency in a distributed simulation system is the simulation manager, APIs, and middleware software to the simulation system. As an example, it is shown in (Puthuff, 2021) that the performance Inter-Process Communication (IPC) latencies of a default ROS2 middleware (which is built on top of the DDS standard) are on the order of milliseconds down to a few hundred microseconds depending on the data size.

**Table 4-2: One-way latency for three different ROS 2 middleware (RWM) implementations**

Implementation	Data Size in Bytes						
	64	256	1k	4k	16k	60k	1M
ROS 2 & RMW(X)	438	436	434	452	432	491	964
ROS 2 & RMW(Y)	275	277	279	282	306	322	632
ROS 2 & RMW(Z)	231	235	238	231	248	264	506

One-way latency in microseconds

Source: (Puthuff, 2021)

It is shown that this latency could be reduced to a few microseconds using RTI's Connex DDS<sup>7</sup> software, exhibiting up to 96% improvement on the latency in comparison to a ROS2 middleware framework (Puthuff, 2021). KAN Simulation Manager has been built based on the same technology and product (Connex DDS), from RTI. The overall measured (one way) latencies for KAN Simulation Manager core and its APIs, range between a few milliseconds down to a few hundred microseconds. Further optimisation of the software codes and implementation of advanced latency mitigation could reduce this latency to the range of tens to hundreds of microseconds.

---

<sup>7</sup> [Connex Product Suite for Intelligent Distributed Systems \(rti.com\)](https://www.rti.com)

**Table 4-3: One-way latency: ROS 2 vs Connex DDS (native implementation)**

Implementation	Data Size in Bytes						
	64	256	1k	4k	16k	60k	1M
ROS 2 & RMW(Z)	231	235	238	231	248	264	506
Connex DDS	9	9	9	10	11	19	339
Latency Reduction	96%	96%	96%	96%	96%	93%	33%

One-way latency in microseconds

Source:(Puthuff, 2021)

From a high-level systemic view there are important factors that could have considerable impact on the overall latency of a distributed simulation system. These factors could be classified into three main design recommendations:

- **The system architectural design:** A good system architecture design could avoid any unnecessary complications to the simulation architecture and reduce the network traffic and workload, resulting in better performance for the overall latency of the system. For example, there are several uses cases in which we can confidently bypass or simplify some of the simulation sub-systems to just focus on the target SUT test requirements and objectives.
- **Fidelities of the sub-systems:** Defining the right levels of fidelity for the simulation sub-systems (sensors, actuators, etc.) is one of the critical technical aspects of any simulation task. Careful consideration of the SUT requirements and test objectives to conclude the right level of fidelity for the sub-systems are among the most crucial points for the optimum design and implementation of any distributed simulation system and its architecture.
- **The data/signals to be sent to the network:** As a general guideline (similar to best practice for vehicle network architectures such as CAN bus), it is not always required to send and receive all the available data and signals to/from the network. The only simulation data that should be sent to the network are the ones needed (to be received) by other consumers (i.e. other simulation tools).

## 5 | **KAN Simulation Manager (KANSM)**

The current technologies for developing distributed simulations within the automotive sector are still at the early stages. It is a challenging, timely and overly complex task to integrate, run and manage the multiple simulation tools within a distributed co-simulation environment, leading to a bottleneck in the current working simulation procedures. Furthermore, the development of a standardised approach has not yet been perfected, and the inability to streamline/optimize this process presents a technological gap in the automotive simulation domain.

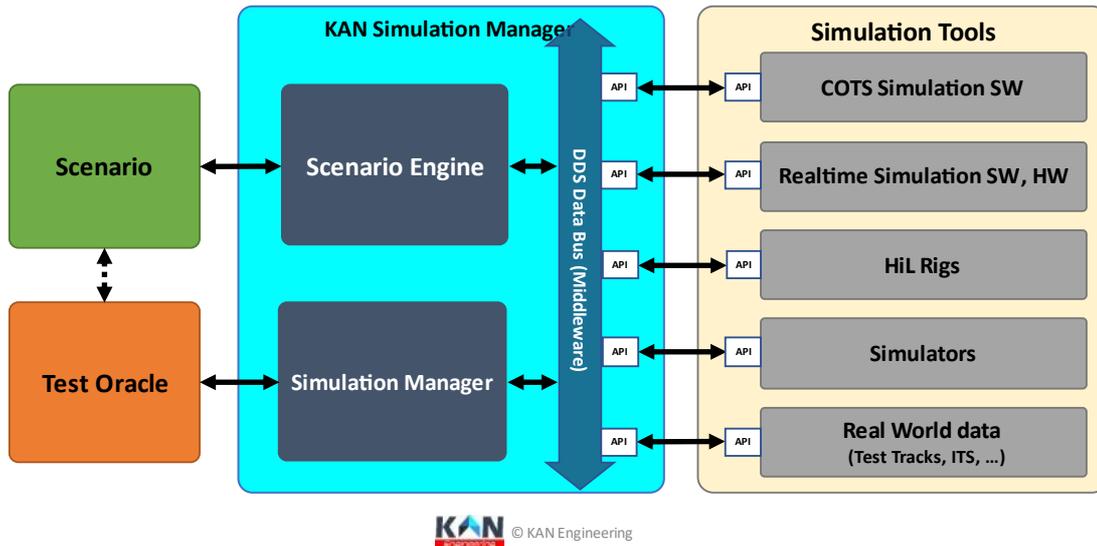
To address the above challenges, KAN Engineering has designed and developed a novel software product (so named KAN Simulation Manager, KANSM) to facilitate the implementation of a distributed simulation environment for an end-to-end development process. KANSM is a model/component-based simulation execution environment. It provides a flexible, expandable distributed architecture for seamless integration of different simulation tools from off-line simulation to real-time simulation, HiL simulation, simulators and real-world testing. KANSM contains a built-in execution control module, KAN Core, which gives access to synchronisation of the systems and simulation parameter setting and DDS based communication interfaces and APIs with common data types, along with the option to have custom data types as needed for further integrations.

The KAN Scenario Engine (KANSE) module (as an add-on feature of KANSM) can set and execute scripted scenarios obtained from various (internal or external) sources and databases. KANSE enables the same scripted scenarios to be used with different simulation software and simulators.

KANSM has modules to handle data recording in local networks or within cloud systems. Routing the data over different networks or the cloud is another feature that enables the integrated simulation tools to perform near-real-time co-simulations with modules running across geographically separated locations. The recorded data can be exported into various formats such as csv, protobuf, etc. which can be evaluated by a test oracle or other V&V methodology to determine a pass/fail, or even rating, for the performance of the SUT, in open-loop or closed-loop fashions.

**Figure 5-1: KAN Simulation Manager (KANSM) overall architecture**

## KAN Simulation Manager



Source: Author Generated (© KAN Engineering)

### 5.1 KAN Core

KAN Core provides the methods to perform the execution control and soft real-time scheduling for the module. It acts as means to receive commands and configuration from KANSM. It is responsible for maintaining the synchronisation between different modules. It has methods to get simulation parameters and scenario parameters from KANSM, which can then be used within the module.

#### Data bus

KANSM is a distributed simulation platform based on OMG Data Distribution Service (DDS™) (DDS Foundation, 2021a). This is a data-centric software framework for transporting and managing real-time data. Data is read from and written onto the data bus, with a predefined data type, that enables multiple modules to share data to perform co-simulation.

#### Data types

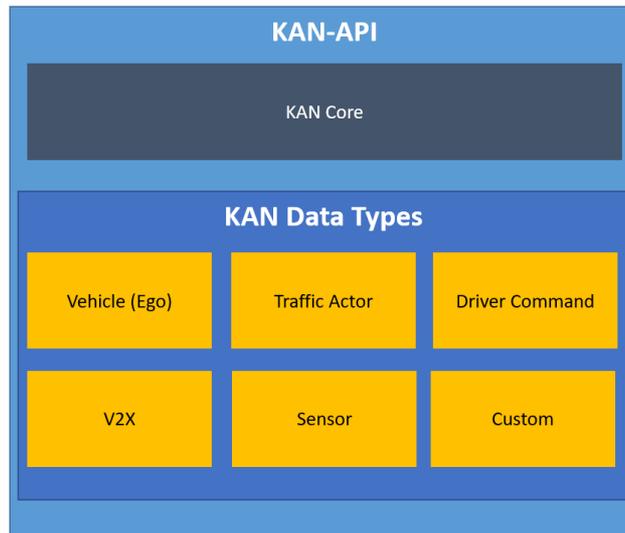
The predefined data types that are supported include vehicle data types (To represent ego vehicles data), driver commands, traffic actor type (simplified data to represent traffic actors), V2X data, sensor data and even custom data types which can be serialised data.

### 5.2 KAN API

KAN API is a set of C++ APIs linked to the KAN Core module, which provides means to interface simulation software to the distributed data bus, as depicted in Figure 5-2. It consists of several

communication APIs using predefined data types relevant to CAV simulation and custom data communication interfaces to accommodate different use cases.

**Figure 5-2: KAN Simulation Manager Core and APIs.**



Source: Author Generated (KAN Engineering)

### 5.3 Simulation modules (Tools)

A module represents a simulation software, HiL rig or simulator with integration to KANSM using custom-built interface applications or any other application built using KAN SDK. Modules utilise the KAN APIs to read and write data from / to the DDS bus and to receive commands and configuration from KANSM.

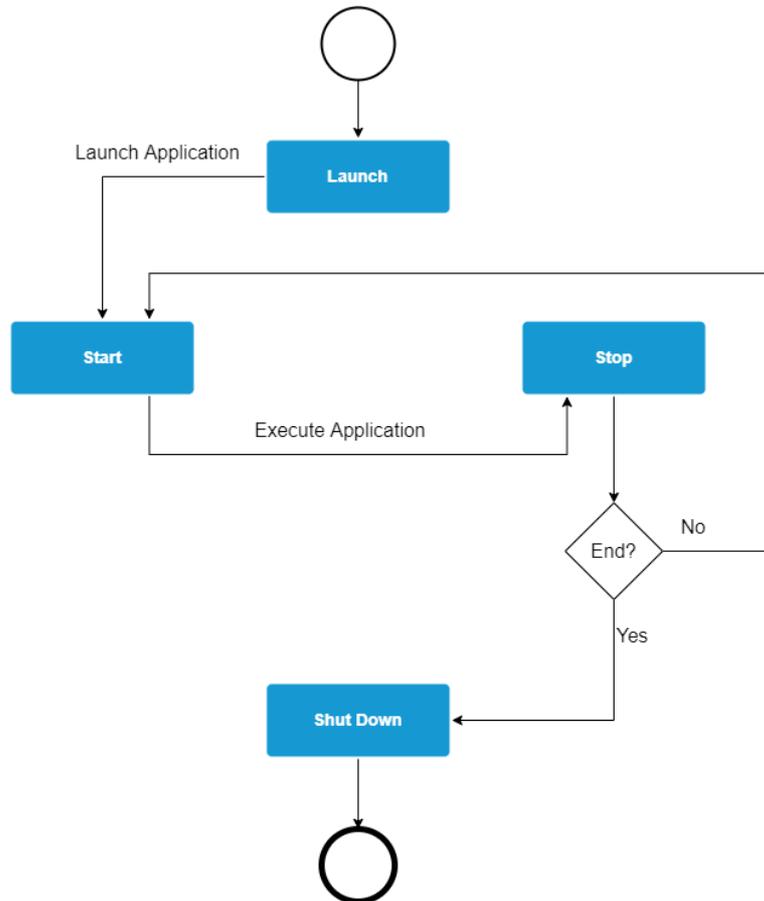
### 5.4 Test configuration

Test configuration specifies the simulation modules which need to be integrated and the hardware components (nodes) in which the simulation software need to be executed. Setting the simulation execution parameters, such as time step and other parameters, is performed in the test configuration. Simulation parameters for each module can be adjusted for different test cases. The user can modify the parameters in each simulation module according to the test scenario.

The KANSM contains a local database where the scenario files and their associated files are stored. The user can choose which scenario to execute for the simulation run: from the (internal) database or the possibility to import scenarios from private/public scenario databases.

### 5.5 Test execution

KANSM can launch the modules according to the test configuration with the help of a background application running on each node. Configuration parameters are sent to the simulation modules according to the specified test configuration. The stages of the test execution cycle is depicted in Figure 5-3 below:

**Figure 5-3: KANSM execution states**

Source: Author Generated (KAN Engineering)

The states the application goes through are:

- Launch: starts the modules
- Start: loads configuration and starts the simulation
- Stop: ends the simulation iteration
- Shut Down: the module is closed

## 5.6 KAN Scenario Engine (KANSE)

KAN Scenario Engine (KANSE) is an add-on module integrated with KANSM for running scripted traffic and weather elements in simulation through scripts based on ASAM OpenSCENARIO v1.1 and road logic based on ASAM OpenDRIVE 1.6. It is able to run parameter sweeps on scenarios with parameter declarations. More details about KANSE are provided in the next section of this report.

## 5.7 Data recording

The data recording module is an add-on module integrated with KANSM for recording data in the data bus (locally or in the cloud) that are originating from different modules at different frame

rates with timestamps. The recorded data can then be exported into different formats for post-simulation analysis (test oracle, for example).

## 5.8 Data visualisation

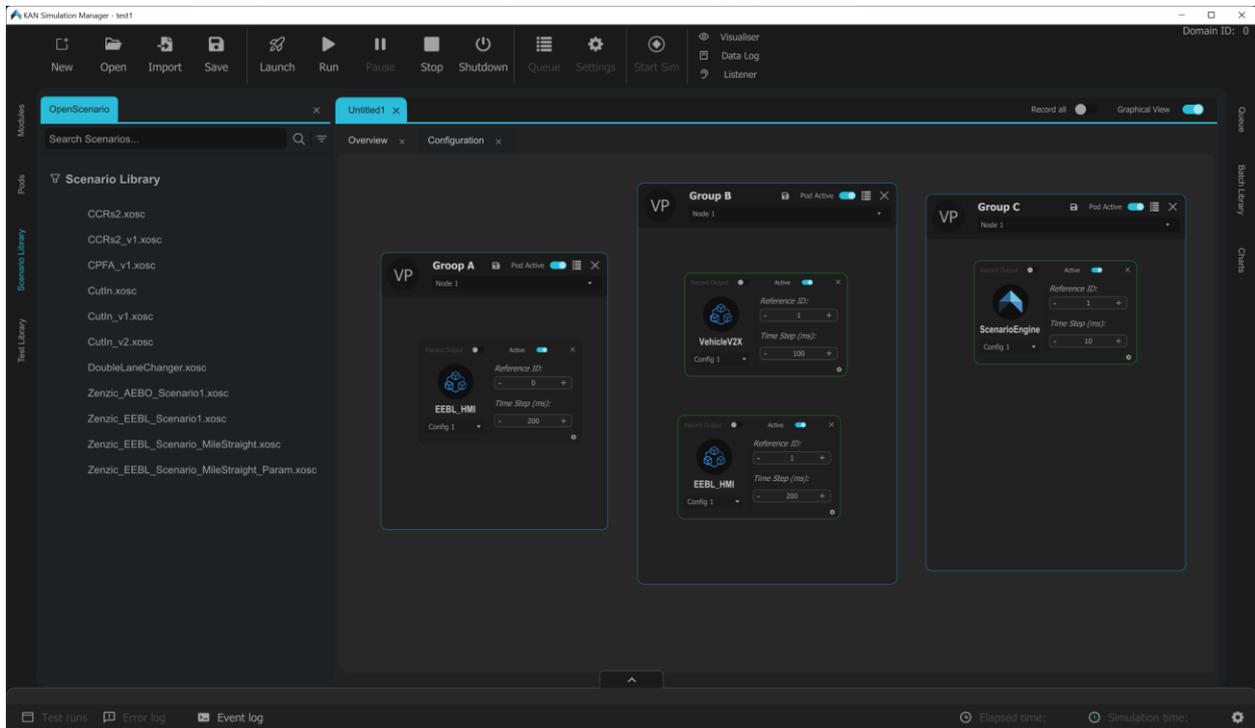
The data visualisation module of KANSM is a set of graphs, 2D maps and other charts that can be used to view the data in the data bus in real-time. This can be useful to view the positions of actors during a scenario execution or to view any signals in the data bus that can be useful to debug or evaluate the simulation execution.

## 5.9 User Interface (UI)

For easy and quick setup and operation of the distributed simulation system, KAN Engineering has designed a comprehensive graphical user interface module as part of the KANSM Software package. The KANSM's core and modules (back end), as explained above, are fully integrated with the user interface (front end). Figure 5-4 represents the first page of the KANSM's user interface. There are four primary elements on the left part of the screen, which represent the Modules, Pods, Scenario Library and Test Library. On the right side, the user can find information about the active Queue, Batch Library and Charts. The user can interact with the basic functionalities of the application at the top, such as:

- Create a new test case
- Open an existing test
- Import an existing test
- Save the current test

To control the application, the user can interact with the following options: Launch, Run, Pause, Stop and Shutdown. The user has the option to visualise the simulation using the built-in Visualiser to analyse the Data Log and access the Listener.

**Figure 5-4: KAN Simulation Manager User Interface (UI)**

Source: Author Generated (© KAN Engineering)

## 5.10 Integrated simulation capabilities

As part of this phase (phase 3) and the previous phase (phase 2) projects, a comprehensive range of interfaces and APIs to various simulation software and tools have been developed and implemented. This powerful feature of KANSM enables easy and quick integration of different simulation tools to form a distributed simulation setup in a flexible manner. The integrated simulation tools and their interfaces can be classified into three main categories, as outlined below:

### Commercial Off The Shelf (COTS)

Integrations to COTS simulation software/tools are based on the provided API from the vendor and KAN-API. According to the available vendor API, KAN-API is integrated by developing a plugin, external interface module or (in some cases) directly into the executables to enable the integration to KANSM.

### Open source

For the integration of open source simulation software/tools, KAN-API can be directly incorporated into the source code if the (software/tool) source code is available, or the associated interface can also be included with KAN-API if the APIs for the simulation software/tool exists.

### Bespoke XiL interfaces

For XiL simulator (properties of the testbeds) integration, bespoke interfaces should be developed in collaboration with testbed partners or system providers. Depending on the existing simulation software/tools used with the XiL Simulators and their provided APIs, the interface to KANSM can be made by developing either a plug-in or an interface module using KAN-API.

### Scenario database

The scenario files can be imported from Safety Pool<sup>8</sup> or other (internal or external) scenario databases using the provided API Schema. The API is integrated with the User Interface so KANSM can retrieve scenarios, in XOSC format, from the cloud.

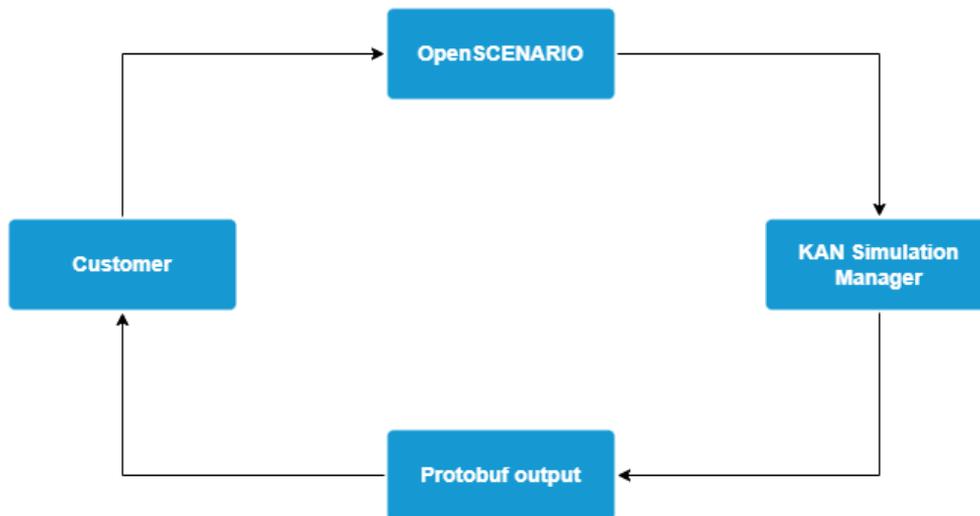
### Test oracle

KANSM supplies the output so that test oracles can determine the test outcome. The software contains a built-in recorder that synchronises recordings from several sources. It has the ability to export the recorded information into a data format that is suited for various clients based on the proto schema that they supply. We currently export data to HORIBA MIRA and WMG.

## 5.11 Closed-loop simulation (V&V)

KANSM allows the user to update the scenario parameters in the scenario file (declared parameters) iteratively from the User Interface for parameter sweeps. As seen in Figure 5-5 below, the programme can also include a closed-loop simulation in which clients can directly communicate with KANSM and send scenarios with various parameters to be simulated. They receive the simulation results, analyse them, and forward the next iteration to the manager.

**Figure 5-5: The V&V closed-loop setup**



Source: Author Generated (KAN Engineering)

<sup>8</sup> [Safety Pool - Powered by Deepen AI and WMG University of Warwick](#)

A (non-exhaustive) list of the simulation capabilities that have been integrated during the phase 2 and phase 3 of this project, are summarised in Table 5-1 below:

**Table 5-1: The (non-exhaustive) list of the integrated capabilities**

Testbeds	Integrated simulation capabilities	Type
<b>UTAC</b>	VI-Grade static driving simulator (DiL)	Bespoke
	UTAC’s full-scale driving Simulator	Bespoke
	Vehicle telemetry (ViL) via private 4G/5G network	Bespoke
	rFpro	COTS
	IPG Carmaker	COTS
	Aimsun	COTS
	PTV VISSIM	COTS
	SUMO	Open Source
<b>University of Leeds</b>	HIKER	Bespoke
<b>Applus IDIADA</b>	V2X IDAPT	Bespoke
<b>SMLL</b>	VISSIM	COTS
	SCOOT (Via VISSIM)	COTS
	Scenario generation (OpenSCENARIO)	Open Source
<b>HORIBA MIRA</b>	Test oracle	Bespoke
<b>WMG</b>	V&V (Closed-loop simulation)	Bespoke
	Safety Pool	Bespoke

Source: Author Generated (© KAN Engineering)

## 6 | KAN Scenario Engine (KANSE)

### 6.1 Introduction

KAN Scenario Engine (KANSE) is an add-on feature of KANSM (KANSM) responsible for the execution of scripted scenarios based on the ASAM OpenSCENARIO v1.1 standard (ASAM, 2021). It can execute the actor motion and set the environment in the simulation and runs on simplified kinematics models to enable real-time motion calculation for multiple actors in a scenario. Runtime behaviour is as specified in the OpenSCENARIO user guide (ASAM, 2021). In addition, it can also handle parameter variations in the scenario, currently by changing parameters declared in the OpenSCENARIO file iteratively. A brief literature survey was conducted to study the existing OpenSCENARIO based scenario engines, and the results have been tabularised in Annex B.

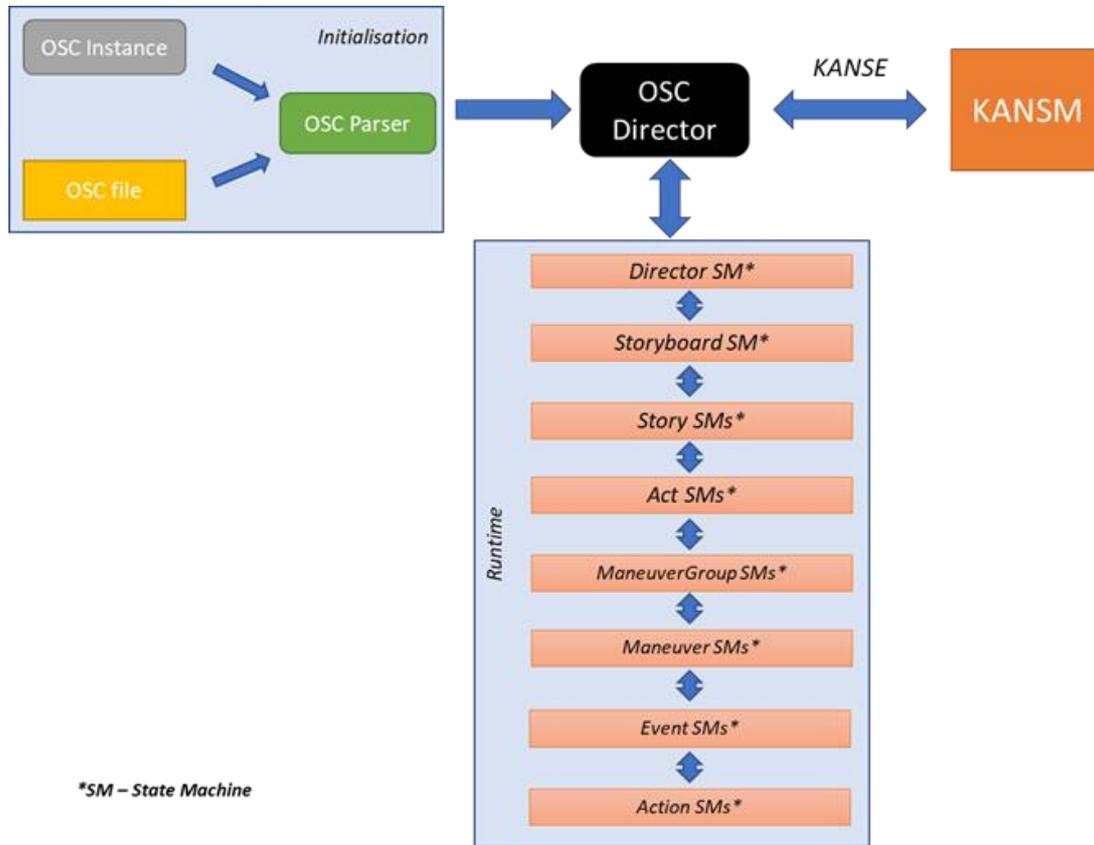
### 6.2 ASAM OpenSCENARIO standard and runtime behaviour

ASAM OpenSCENARIO (v1.x) is an XML-based scenario scripting language to define traffic and environment behaviour for scenario-based testing. The main components of the definition are *ParameterDeclarations*, *CatalogLocations*, *RoadNetwork*, *Entities* and *Storyboard*. *ParameterDeclarations* element is used to declare scenario parameter values to be used in the scenario. *RoadNetwork* defines the road logic used to define actor positions, usually based on ASAM OpenDRIVE. *Entities* element is used to specify the actor properties in the scenario. *Storyboard* element contains the definition of the expected runtime behaviour of the scenario as defined in the OpenSCENARIO User Guide (ASAM, 2021).

### 6.3 KANSE high level architecture

KANSE consists of nested state machines whose start, execution and termination are as defined in the OpenSCENARIO V1.1 standard (ASAM, 2021) which are coupled at the top level with the OpenSCENARIO (OSC) Director. These state machines communicate with KANSM through the KAN API (Figure 6-1). First, the scenario file specified in the file path in KANSM is parsed, and the elements of the scenario are initialised in terms of actor positions, orientations, velocities, and weather elements as defined in the scenario file. On giving the run command from KANSM, KANSE updates the scenario elements at each time step, and publishes actor data to the DDS data bus. Scenario execution is terminated on fulfilling the defined stop trigger in the storyboard or on issuing the stop command in KANSM. The scenario can then be re-run with new parameter values if required for seamless iterative execution.

**Figure 6-1: High level architecture of KANSE**



Source: Author Generated (© KAN Engineering)

## 6.4 Functions and features

The key functions and features of KANSE are summarised in Table 6-1:

**Table 6-1: Functions and features of KAN Scenario Engine**

Function	Feature(s) to accomplish function
OpenSCENARIO file validation and parsing	The KANSE OSC Parser can validate the given scenario file against a predefined XML schema based on the UML model. Any parameters declared in the scenario file are assigned where relevant.
OpenSCENARIO based scenario execution	As seen in Figure 6 1, KANSE consists of a nested state machine implementation with the OSC Director State Machine (SM) in the top-most level controlling the execution of the child state machines based on the runtime behaviour specified in the OpenSCENARIO v1.1 standard.
OpenSCENARIO based action execution	The KANSE OSC Action Runner is responsible for setting the dynamics of all the actors in the scenario as defined in the actions in the OSC file.
OpenDRIVE road logic-based positioning	The KANSE OpenDRIVE navigator is responsible for providing the world, road, and lane position data of each actor at each timestep.

Function	Feature(s) to accomplish function
Synchronous (real-time) or Asynchronous scenario execution	The KANSE Core is responsible for updating the dynamics of the actors and environment on each time step.
Parameter sweeping	The user can modify the values of parameters declared in the OpenSCENARIO file on each iteration manually or from a test oracle.
Feedback-based dynamic control of actors	The KANSE Active Driver feature can control traffic actors using feedback to keep a desired speed and lane offset, while respecting the performance constraints declared in the scenario file.

Source: Author Generated (© KAN Engineering)

## 6.5 Current limitations

At the time of writing this report, some of the limitations of KANSE are (these features are in the product roadmap and are in the development stage):

- No support for roads with superelevation in the OpenDRIVE file.
- Limited support for the execution of lateral actions.
- No support for the execution of synchronise and routing actions.
- Limited support for parameter value distributions.

## 7 | System integration

### 7.1 Introduction

This project involved the integration of various simulation capabilities from several partners, with their sites located at different geographical locations spread across the UK (Figure 7-1). AWS site-to-site VPN (AWS, 2022) was used to enable a data connectivity layer between the testbeds.

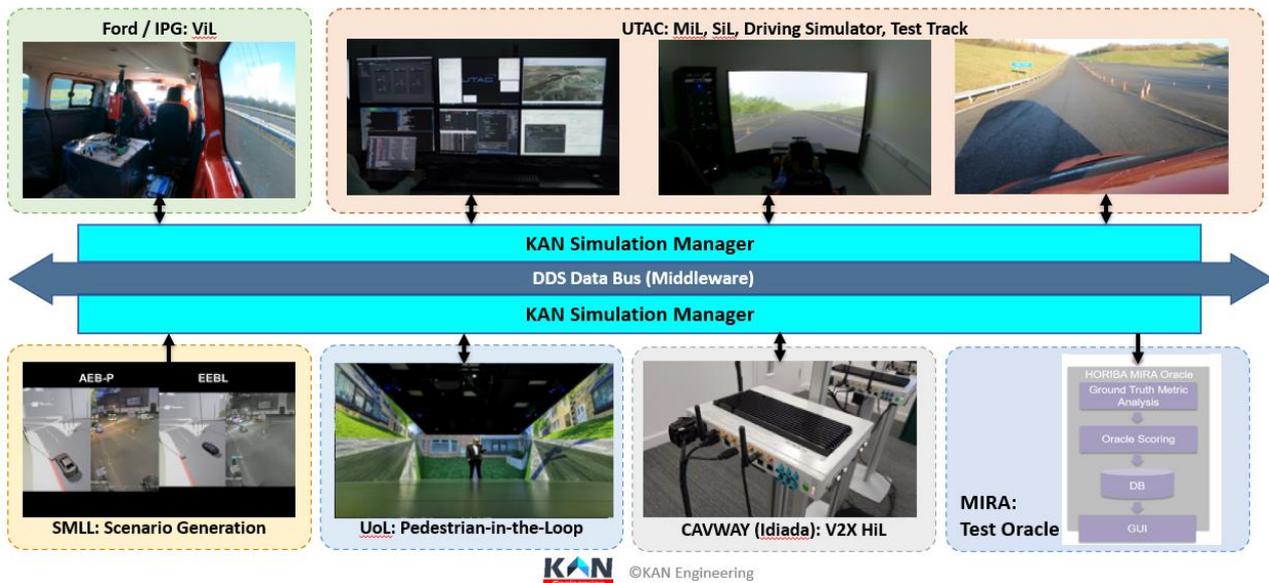
**Figure 7-1: Geographical locations of UK CAM testbeds**



Source: Google Earth

The high-level architecture of the integrated distributed simulation system is depicted in Figure 7-2. KANSM was used to orchestrate the different components of the simulation as per the given use case. KAN-API was used to integrate the testbeds that needed to access the DDS data bus in real-time with the relevant topics as required by the project use cases. For the testbeds that required post-processing data, Microsoft OneDrive was used as a platform for file sharing. Google Protobuf was used to serialise output data as per specifications decided by each testbed.

**Figure 7-2: The project's high-level architecture and the integrated capabilities involved**



Source: Author Generated (© KAN Engineering)

More specific to each project partner, the following integration activities were performed:

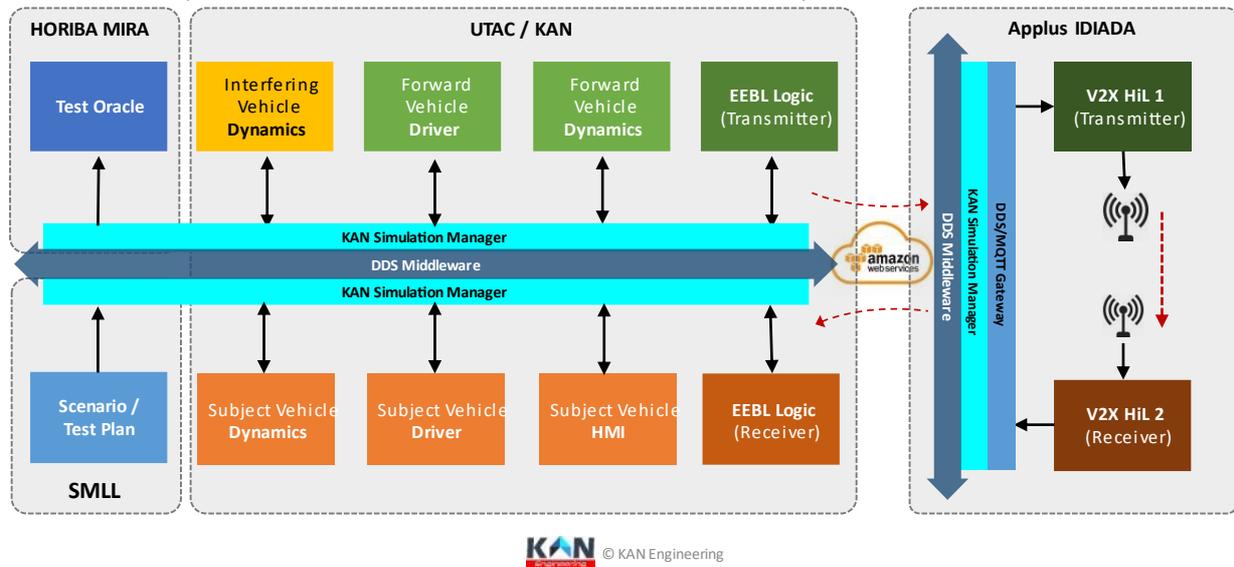
## 7.2 UTAC

The DDS-based simulation ecosystem at UTAC Millbrook was implemented in the previous phase of this project (phase 2) and was extended for this phase of the project. KAN-API was used to integrate rFPro for environment visualisation, IPG for vehicle dynamics, ADAS models (provided by Ford) and the driving simulator via Simulation WorkBench. The ViL simulator was integrated for publishing GPS/IMU positions and orientations to the data bus using 4G/5G connectivity in the test track. KANSM software was used to orchestrate the simulation nodes or modules as per the use case requirement. More details about UTAC simulation capabilities and their contributions to the project delivery are provided in Section 8.

## 7.3 Applus IDIADA CAVWAY

Applus IDIADA’s ecosystem was integrated with KAN-API via the CAVWAY Bridge to exchange V2X messages for the EEBL use case. The CAVWAY bridge handles the DDS to MQTT data mapping between the DDS data bus and IDIDA’s V2X transmitter-receiver setup. The high-level architecture of the distributed simulation setup between the UTAC, Applus IDIADA, SMLL and HORIBA MIRA that was used for the EEBL SiL and DiL tests is presented in Figure 7-3 below. More details about Applus IDIADA’s V2X HiL rig and their contributions to the project delivery are provided in Section 9.

**Figure 7-3: Electronic Emergency Brake Light (EEBL) System, Interoperable Distributed Simulation Setup**

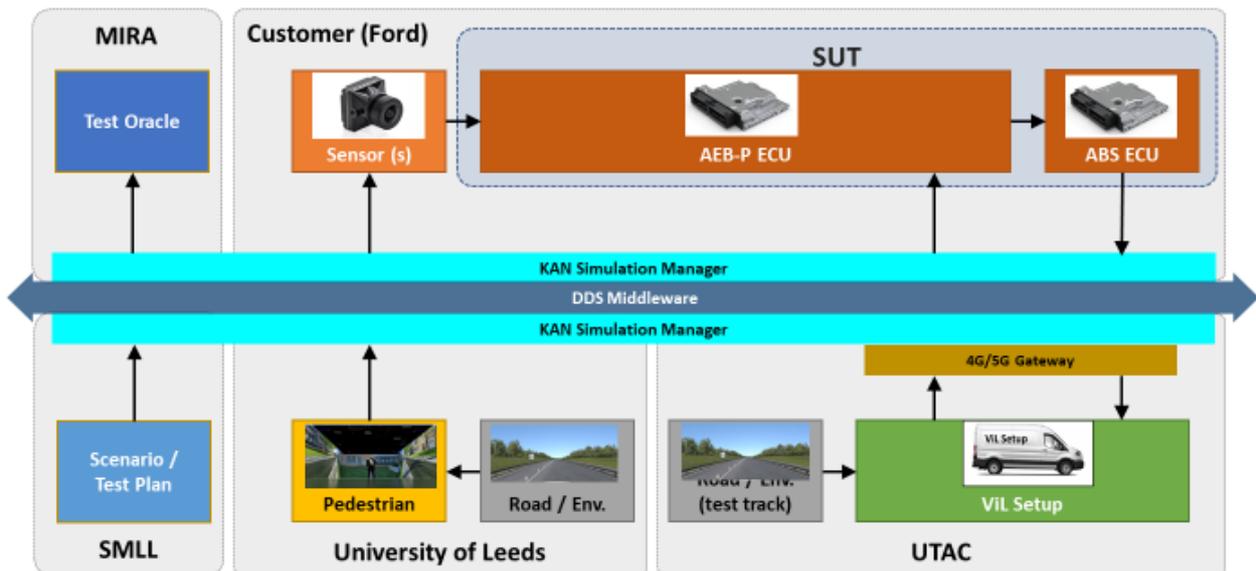


Source: Author Generated (© KAN Engineering)

## 7.4 University of Leeds

The University of Leeds' HIKER pedestrian simulator was integrated using KAN-API to achieve PiL functionality in the simulation system. This enabled HIKER to publish its pedestrian position and orientation to the DDS data bus and subscribe to the positions and orientation of other traffic actors in the simulation scene from the DDS data bus in real-time. The subscribed actor data was visualised in the simulator screens for the pedestrian to view the other traffic actors in the simulation. This integration setup was used for the AEB-P SiL and DiL tests. More details about the University of Leeds' HIKER and their contribution to the project delivery are provided in Section 10.

**Figure 7-4: Automatic Emergency Braking – Pedestrian (AEB-P) System, Interoperable Distributed Simulation Setup**



Source: Author Generated (© KAN Engineering)

## 7.5 SMLL

SMLL was involved in the creation of scenarios using real video captured from cameras in different locations. This data was transformed into a scenario script in OpenSCENARIO. The trajectory data was then transformed to match the road coordinates of the UTAC-Millbrook 'Mile Straight' track. More details about SMLL scenario generation and their contribution to the project delivery are provided in Section 11.

## 7.6 HORIBA MIRA

The output data of each simulation run for each use case were given to HORIBA MIRA's test oracle in their predefined protobuf format. The oracle generates scores for the performance of the SUT (AEB-P and AEB-O) on the scenario outcome based on the defined metrics. These scores can then be used to identify the critical scenarios to improve the performance of the SUT. More details about HORIBA MIRA's test oracle and their contribution to the project delivery are provided in Section 12.

## 7.7 WMG

The integration with WMG was done in two ways: first to obtain the scenario for testing, and then to send output data of the simulation on each iteration and receive a modified parameter set. The data format was predefined in a suitable protobuf serialisation. The scenario file with the necessary parameters declared in the OpenSCENARIO file for each use case was uploaded to the SafetyPool scenario database. KANSIM was integrated with SafetyPool through REST APIs to obtain the correct scenario file given the scenario ID. After running the scenario, the output data was serialised in the defined protobuf format and handed over so that new parameter values could be calculated and sent back, after which the scenario was re-run with the new

parameters. More details about WMG’s V&V pipeline and their contribution to the project delivery are provided in Section 13.

## 7.8 Simulation setup

The project use cases of AEB-O, AEB-P and EEBL were tested using several variations of MiL, SiL, DiL and ViL. The following sections outline the setup and features of each use case.

### Use case 1: AEB-O

The vehicle and ADAS models from the customer (Ford) implemented in MATLAB Simulink included functionality for both AEB-O and AEB-P. The MiL simulation was executed using IPG Carmaker and MATLAB Simulink (Figure 7-5). Original customer scenarios were tested, and new scenarios were created with the road as the Millbrook Mile Straight track (to match the specifications in EuroNCAP tests). The higher simulation time required was identified as one limitation in MiL testing, especially when running iterative simulations.

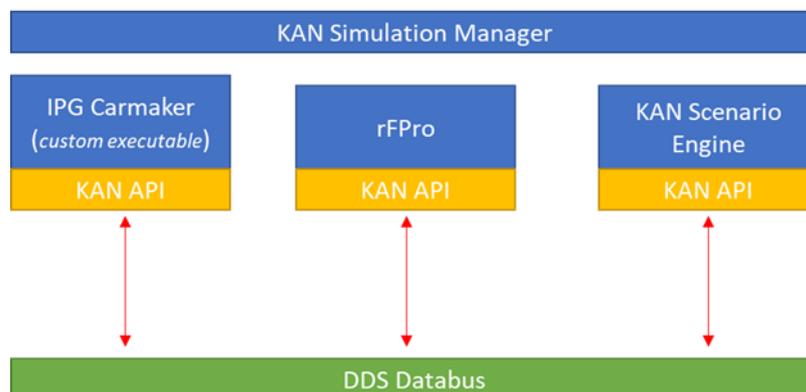
**Figure 7-5: MiL Automatic Emergency Braking – Oncoming (AEB-O) setup**



Source: Author Generated (© KAN Engineering)

The Simulink model containing the AEB functionality was then converted to C code and built as an executable to be run for SiL/DiL simulation. This enabled the simulation to run in real-time or (up to 10 times) faster than real-time, resulting in 30 times improvement on the simulation execution time. In the SiL setup (Figure 7-6), the created executable was run using IPG Carmaker. KAN Scenario Engine injects traffic into the Carmaker simulation environment. The capability of traffic injection enabled the use of OpenSCENARIO based scripts using KAN Scenario Engine. A parametrised OpenSCENARIO file was designed to describe an AEB scenario with one traffic vehicle. The parameter values can be varied on each simulation iteration manually or through a predefined protobuf data file.

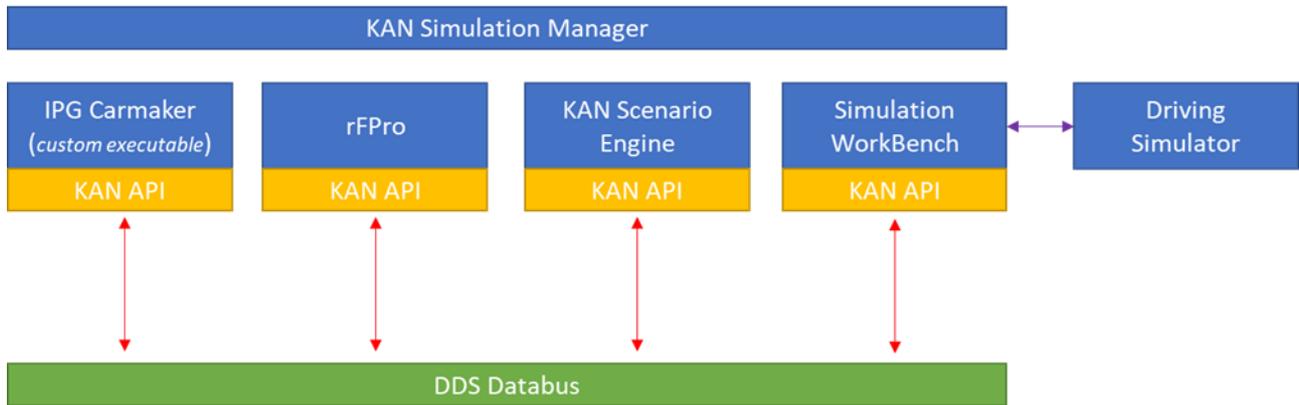
**Figure 7-6: Automatic Emergency Braking – Oncoming (AEB-O) SiL setup**



Source: Author Generated (© KAN Engineering)

The C code was further expanded with KAN-API to enable injection of driving simulator commands (throttle, brake, steering, gear) and traffic (positions, orientations, velocities, accelerations). This meant that the simulation was expanded to include DiL capability as well. For the DiL setup (Figure 7-7), the driving simulator data (available through SimWB) is integrated using KAN-API. This enabled the vehicle model running in IPG Carmaker to be driven by a real driver sitting at the simulator.

**Figure 7-7: Automatic Emergency Braking – Oncoming (AEB-O) DiL setup**



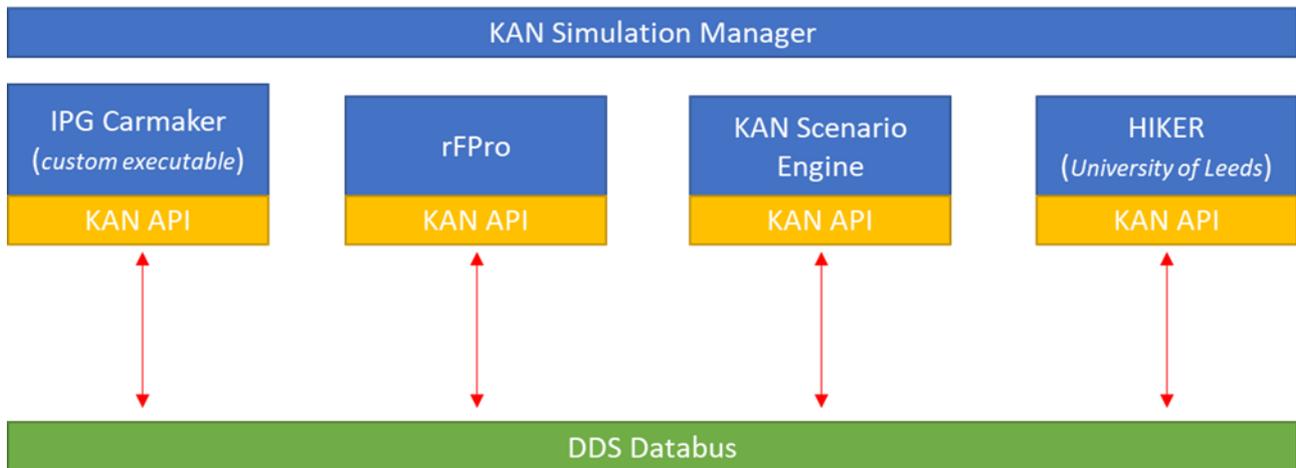
Source: Author Generated (© KAN Engineering)

**Use case 2: AEB-P**

The same compiled executable created for AEB-O was used for AEB-P tests since the Simulink model contains this functionality (provided by Ford). The MiL setup is essentially the same as in Figure 7-5 (for AEB-O). The only difference was that the scenarios tested in IPG contained pedestrian actors.

The flexibility and expandability of our simulation approach is clearly demonstrated through the SiL setup depicted in Figure 7-8. The integration with the University of Leeds’ pedestrian simulator (HIKER) was done using KAN API in addition to using scripted pedestrians from OpenSCENARIO through KAN Scenario Engine or IPG Carmaker. This enabled the simulator setup at UTAC Millbrook to publish traffic actor data from the ego vehicle and scenario to the DDS data bus and subscribe to the pedestrian data published by HIKER. In turn, the HIKER was able to visualise the vehicles for the real pedestrian in real-time.

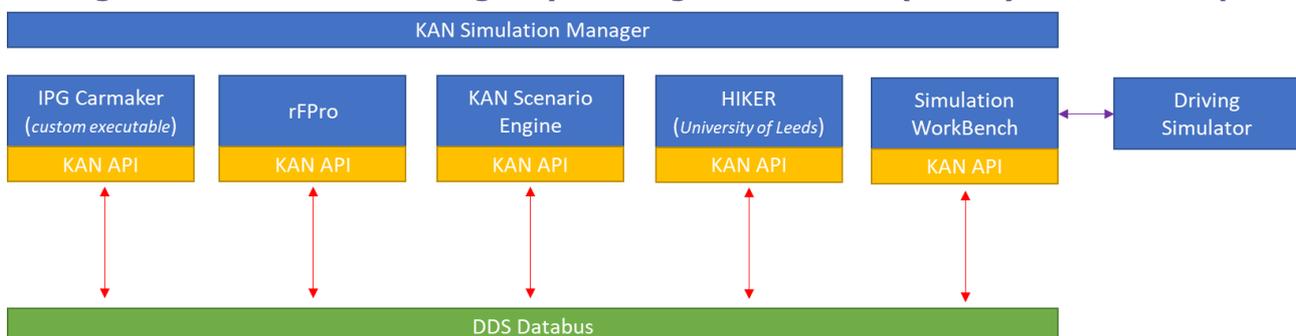
**Figure 7-8 Automatic Emergency Braking – Pedestrian (AEB-P) SiL/PiL setup**



Source: Author Generated (© KAN Engineering)

The DiL/PiL setup (Figure 7-9) could be seen as an addition to the setup in Figure 7-8 with the HIKER simulator. This enables even more sophisticated testing in the ADAS/CAV domain, where a real driver/pedestrian interaction can be facilitated in mixed real/virtual environments simultaneously at both testbeds (UTAC driving simulator and the University of Leeds HIKER).

**Figure 7-9 Automatic Emergency Braking – Pedestrian (AEB-P) DiL/PiL Setup**



Source: Author Generated (© KAN Engineering)

**Use case 3: EEBL**

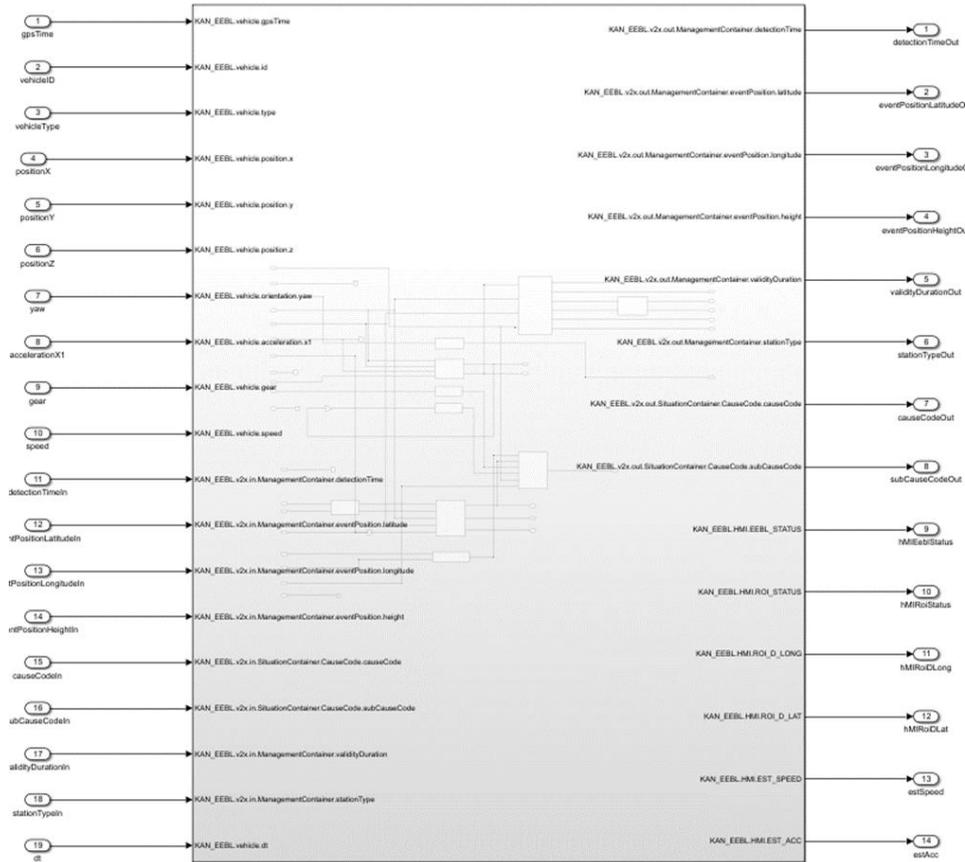
A simple EEBL algorithm (KAN EEBL) was developed by KAN Engineering based on ISO 20901:2020 standard (BSI, 2020a). The model was first designed in Simulink, as depicted in Figure 7-10 to achieve the following functions:

- The EEBL-T (transmitter) logic has been implemented such that an emergency braking flag is broadcast over V2X when the fitted vehicle has a deceleration greater than the threshold of 4 m/s<sup>2</sup>. In addition to the braking flag, it also broadcasts the geo position (latitude and longitude) where the brake event was triggered.
- The EEBL-R (receiver) logic has been implemented such that a warning message is displayed on the driver dashboard (through an HMI – designed by KAN Engineering) if the braking event was triggered inside the Region Of Interest (ROI). Both Forward Vehicle (FV) and

Subject Vehicle (SV) travel in the same direction, and the SV is in the SUT with the gear in forward or neutral positions.

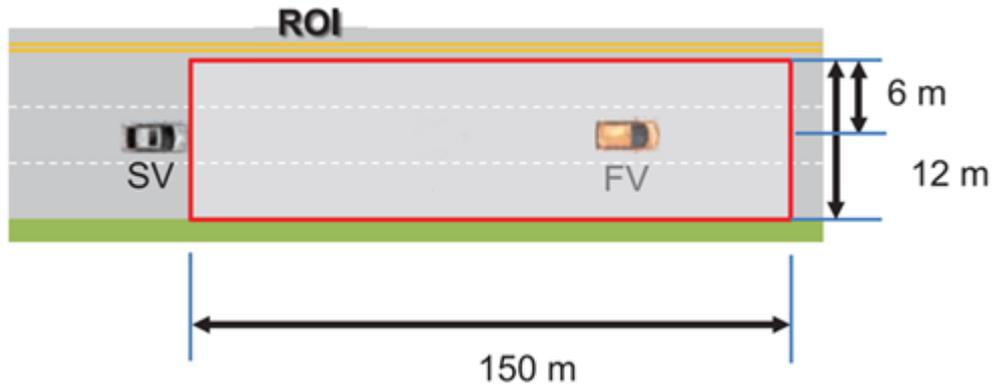
- The message structure has been mapped to comply with the Decentralized Environmental Notification Message (DENM) standard.(ETSI, 2014)

**Figure 7-10 Simulink model of Electronic Emergency Brake Light (EEBL) algorithm**



Source: Author Generated (© KAN Engineering)

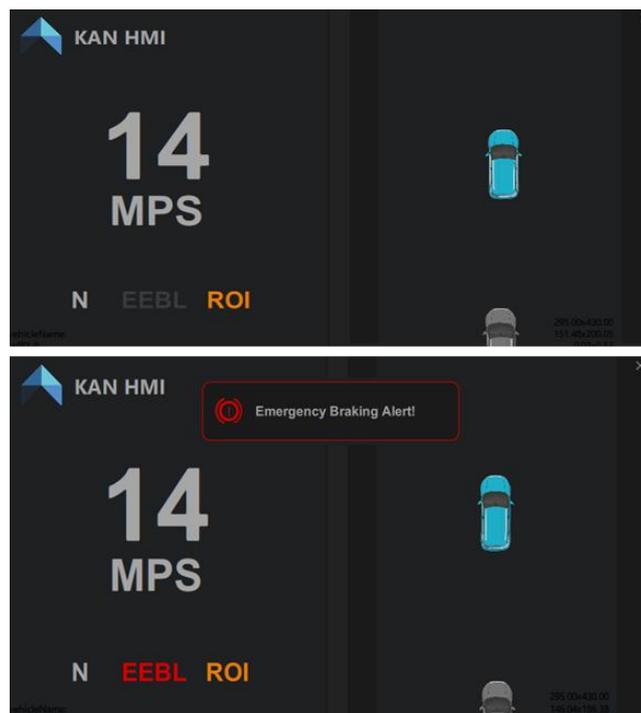
**Figure 7-11 Electronic Emergency Brake Light (EEBL) scenario**



Source:(BSI, 2020a), modified by Author(© KAN Engineering)

The Simulink model was then converted to C code using Simulink coder and integrated with KAN-API and the KAN HMI for displaying EEBL warnings (Figure 7-12). When the FV is in the ROI of the SV, the ROI indicator message is turned on. When the SV receives an emergency brake flag from the FV with the necessary conditions, the EEBL warning is switched on to alert the driver of a possible dangerous situation. The EEBL scenario was created according to the OpenSCENARIO v1.1 standard.

**Figure 7-12 KAN HMI user interface with Region-Of-Interest (ROI) (top) and Electronic Emergency Brake Light (EEBL) (bottom) indicators**



Source: Author Generated (©KAN Engineering)

In the MiL setup (Figure 7-13) the created EEBL Simulink model was integrated directly with IPG Carmaker using Carmaker4SL.

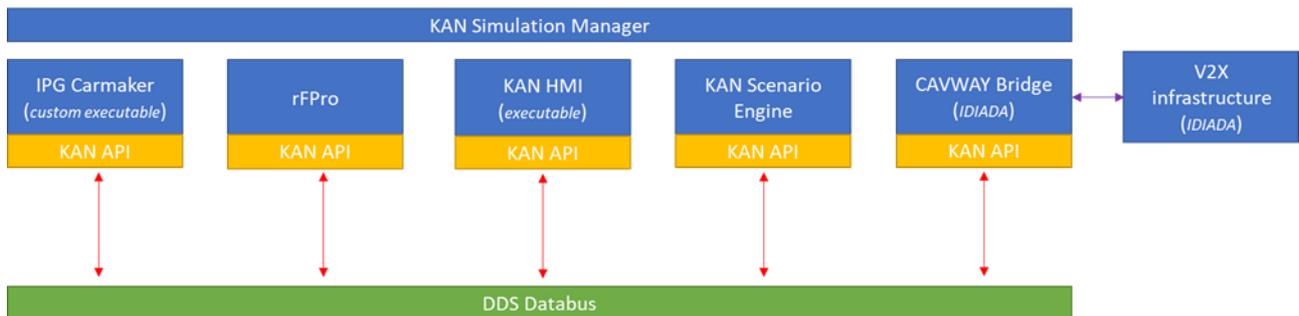
**Figure 7-13 Electronic Emergency Brake Light (EEBL) MiL setup**



Source: Author Generated (© KAN Engineering)

In the SiL setup (Figure 7-14), the CAVWAY bridge from Applus IDIADA was integrated with KAN API, and acts as a gateway to the V2X infrastructure. KAN HMI is used to visualise the EEBL and ROI warnings to the driver of the SV.

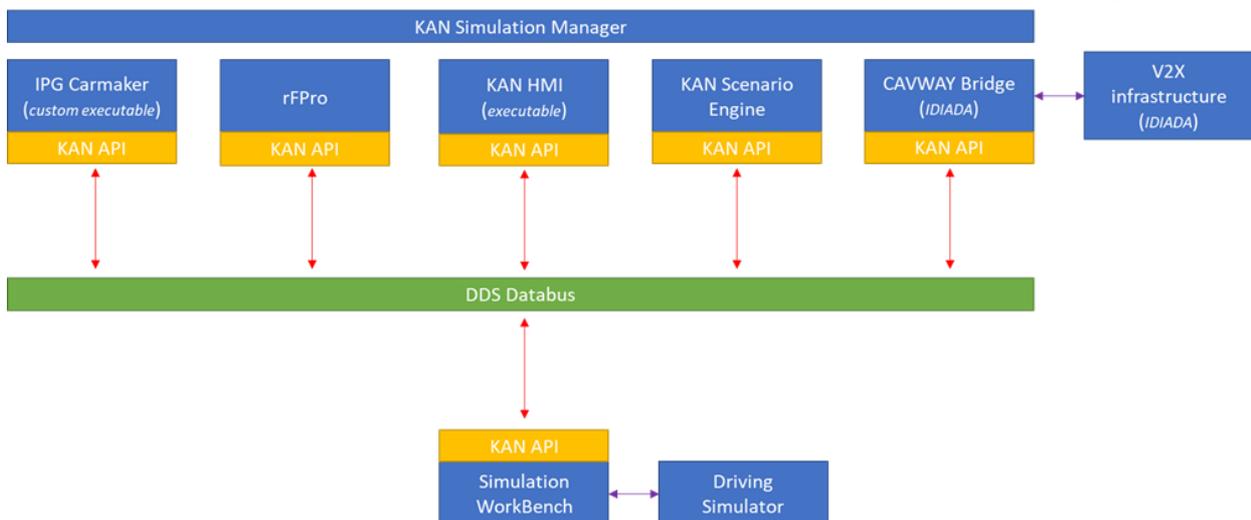
**Figure 7-14 Electronic Emergency Brake Light (EEBL) SiL setup**



Source: Author Generated (© KAN Engineering)

In the DiL setup (Figure 7-15), the SiL setup is extended to include the driving simulator integration.

**Figure 7-15 Electronic Emergency Brake Light (EEBL) DiL setup**



Source: Author Generated (© KAN Engineering)

## 8 | UTAC: MiL, SiL, ViL simulation and test track

By: Peter Stoker

### 8.1 Interoperable simulation role

UTAC Millbrook has specified, built and commissioned a unique suite of simulation as part of the CAV testbed. The UTAC Millbrook Simulation Environment is a comprehensive integrated X-in-the-Loop (XiL) ecosystem, including a wide range of the latest software, hardware, toolchains and infrastructure. As well as support to conventional vehicle programme development, it is capable of modelling, design, development, test and validation of CAVs and ADAS systems.

**Figure 8-1: UTAC Driving Simulator (DiL)**



Source: VI-Grade<sup>9</sup>

Utilising UTAC Millbrook's high-speed fibre and 5G private network, and a modular simulation architecture, UTAC Millbrook provides a flexible connected environment, to cover offline, real-time, and HiL simulations across the business, including real-world correlation, test and validation of sub-systems and full vehicles.

---

<sup>9</sup> [Compact Simulator, driving simulator | VI-grade](#)

**Figure 8-2: UTAC virtual test track**



Source: Author Generated (UTAC)

The suite offers a bespoke capability to help our customers develop, test and validate in a virtual and blended controlled, secure environment.

The design and implementation of this integrated simulation environment has evolved beyond the central DiL simulator as the first element to be installed in 2018. This is now working with a wide range of professional software, including IPG, VI-Grade products, rFpro, traffic modelling (SUMO and AIMSUN), sensor modelling, and communications and their integration, via complex and internally-developed interfaces, APIs (C++, C#, Python, Matlab, Simulink), and middleware. The architecture is now reaching out across the business to other test laboratories at UTAC Millbrook, such as those in the propulsion centre, starting to realise a much wider “X-in-the-loop” capability in the transition from internal combustion to electrical vehicles of the future.

These capabilities were put at the disposal of the project consortium and the following activities were completed as part of the project delivery

- Working with the System Integrator and other partners on defining and delivering the PoC demonstrator
- Capability readiness - Ensure readiness of the required simulation capability. Here, the lessons for phase 2 were taken into account, and the specification of the ecosystem defined in advance of project start, enabling a fast ramp up of capability
- Demonstrator delivery - Working in co-ordination with the System Integrator and the project customer (Ford) in defining and delivering the demonstrator, and other relevant supporting assets, such as track availability, network connection to the 4G / 5G testbed and additional positional equipment to accurately reflect the real position of the vehicle into the simulation.
- Stakeholder workshops - Support in delivery of stakeholder workshops both physical and held remotely
- Customer engagement - Engagement with potential customers for the demonstrator, and maintaining contact throughout the development.
- UTAC Millbrook were instrumental in proposing the partnership with the University of Leeds Institute for Transport Studies. Not only have they been a key partner in the CCAV funded

VeriCAV project, but also brought their extensive research expertise to bear in the field of mixed reality. One of the advances that UTAC Millbrook proposed was the mixing of virtual and real experiences. This looks to the future roadmap of certification, where the long term view is that certification to new regulations can and will be done with some level of virtual experience. As stated many times, driving the required mileage physically is not possible to cover all eventualities.

## 8.2 UTAC Millbrook Proving Ground

Figure 8-3: UTAC Millbrook Proving Ground



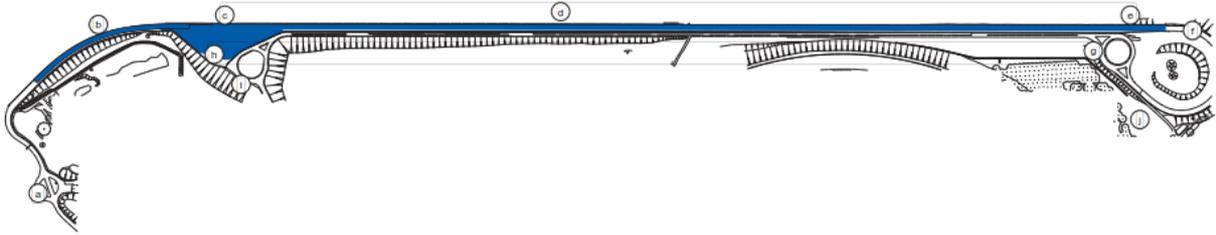
Source: Author Generated (UTAC)

UTAC Millbrook has been operating a secure, confidential vehicle proving ground for over 50 years. Custom-built to simulate the world's toughest terrains, they provide an invaluable test and development resource at a single, secure and well supported site. The on-road tracks and features make up a major proving ground, with more than 70km of varied surfaces and a range of technical features. UTAC Millbrook supports its test track customers with:

- Specialist durability correlation team
- Whole vehicle general quality, corrosion and structural durability
- Objective and subjective measurement
- Corrosion chambers
- Climatic chambers
- Type approval advice, guidance, application and certification

For the purposes of the demonstration, a known section of test track was chosen – the Mile Straight.

**Figure 8-4: Mile straight (marked for start, 1/2, 3/4, 1km and start, 1/2, 3/4, 1 mile)**



Source: Author Generated (UTAC)

It is a long, precisely levelled surface with fast approach and departure, ideal for the evaluation of vehicle acceleration and braking performance. It is an asphalt surface laid level end to end (1 mile) with 1 in 80 crossfall.

In order to generate a common reference across all of the testbeds, a specific OpenDRIVE model of the track was created and then supplied to the partners, which precisely reflected the

real track. In the UTAC Millbrook simulator, the digital terrain version was used, as modelled for the CAV testbed, which gives an overall accuracy of 1mm in Z-Axis.

**Figure 8-5: Actual test track**



Source: Author Generated (UTAC)

**Figure 8-6: Virtualisation of Mile Straight model**



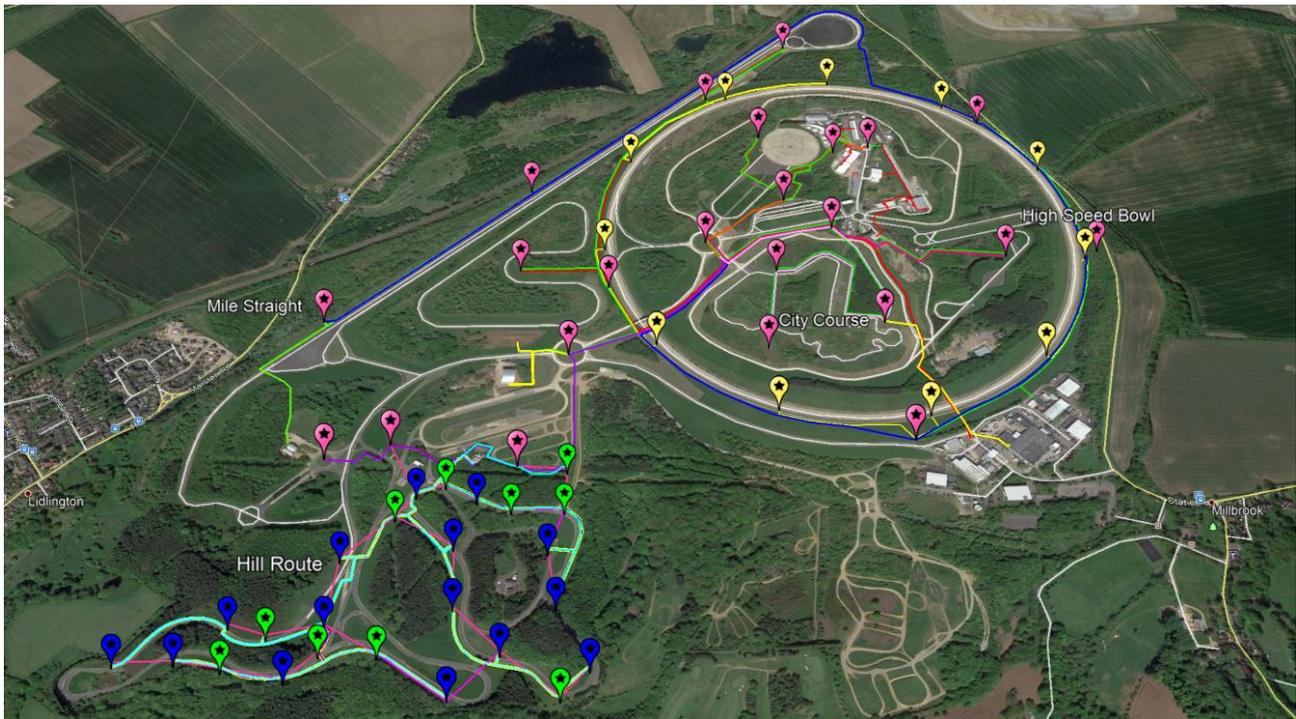
Source: Author Generated (UTAC)

### 8.3 UTAC Millbrook private network

UTAC Millbrook has a private hyper dense small cell network supported by Dense Air that can provide cellular connectivity across the site. The network consists of 4G and 5G Cores, 4G and 5G Stand Alone RAN, mmWave 70GHz, high speed internet connectivity, pervasive coverage around 70km of test tracks and deep indoor coverage into the majority of buildings. This allows customers to stream live data from a vehicle to an engineer or test a huge range of solutions and applications, as an initial controlled test environment or as a final location for application testing. Each mast has a cabinet which has power and spare ports on the switches to access the

network if required. UTAC Millbrook has a range of Mi-Wi routers SIM cards and handsets to connect to the network.

**Figure 8-7: UTAC 4G/5G private network**



Source: Author Generated (UTAC)

Some example applications are as follows:

- Real-time data analysis from test vehicles
- Smart infrastructure development and testing
- Connected vehicle testing
- Autonomous vehicle testing
- High speed data transfer testing
- Connected events and demonstrations

The network was used in the conduction of real-time data between the vehicle on the test track, back to the simulation in the CAV Village on site. A gateway module was utilised from KAN Engineering, and a modem was deployed to the test vehicle. The simulation suite was connected to the network directly, allowing real time communication between real and virtual environments. This allowed, for example, the ability to have the position of the real vehicle transferred in real-time to the University of Leeds environment, and also the real pedestrian to be represented onto the test track at UTAC Millbrook.

## 8.4 Integration with the Interoperable Simulation project

Re-use of the connected simulation environment from Phase 2 was crucial to the framing of the Phase 3 programme, and the architecture expanded to cover connections to the University of

Leeds, IDIADA and HORIBA MIRA. SMLL is already historically linked (as part of the previous phase of this project – Phase 2). Extensive use was made of the software and hardware tools at UTAC’s simulation laboratory, allowing the System Integrator to best model the complex environment needed for the Phase 3 programme. The UTAC simulation lab sat at the centre of the programme, and most of the practical work was conducted on site, as described in 7.2.

### 8.5 Summary

UTAC Millbrook was proud to play a role in the creation of this complex but unique architecture on site, as it has made the best use of the facilities co-invested in by Government, bringing out even more potential for the future. It is critical that to move into V&V by simulation as a part of future regulatory approvals that the simulation is trusted. Bringing real-time actors into the simulation was a challenge and stretch at times, but paves the way for further development and work in this field. This matches both the company strategy, but also those for regulators and consumer bodies, as scenarios and vehicle systems grow ever more complex. We look forward to further developments in this interoperable space.

## 9 | Applus IDIADA: V2V HiL Simulation

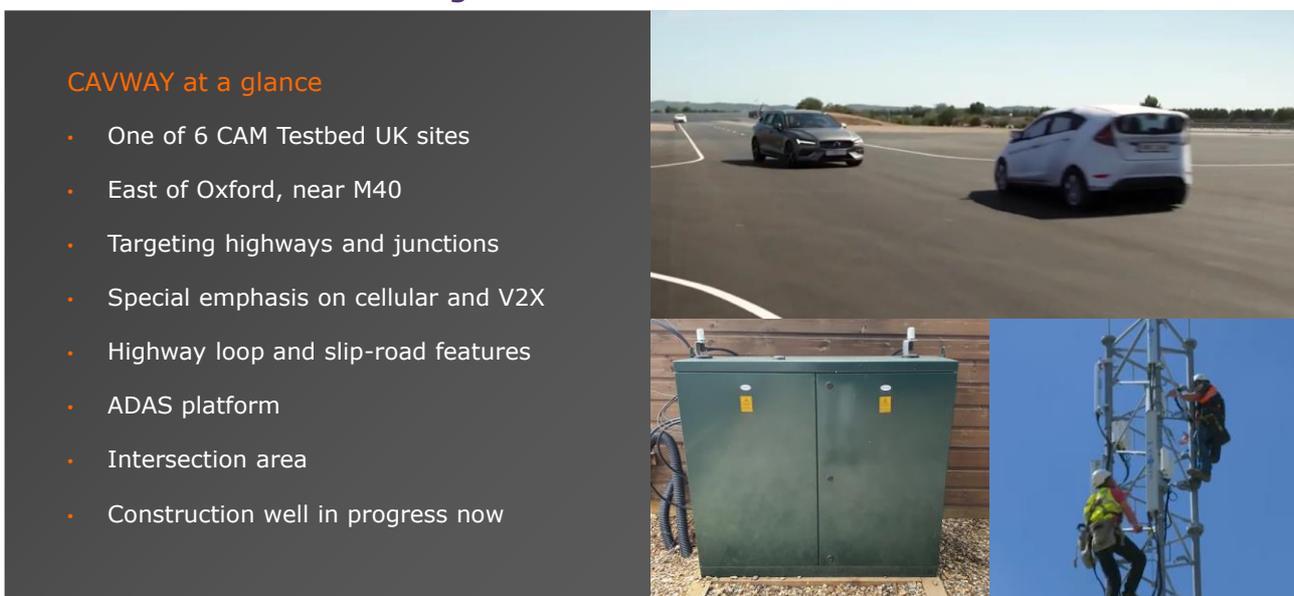
By: Charlie Wartnaby

### 9.1 Interoperable Simulation role

Applus IDIADA's particular contribution to this PoC project concerned Vehicle-To-Everything (V2X) messaging, bringing live radio communications into the interoperable ecosystem in Hardware-in-the-Loop (HiL) fashion. This leveraged existing hardware and software components we already have available, intended for CAVWAY and other applications.

### 9.2 CAVWAY Proving Ground

**Figure 9-1: CAVWAY overview**



Source: Author Generated (Applus IDIADA)

CAVWAY is designed from scratch for CAV testing. It has a 3 km highway loop of two lanes and upwards, and areas for slip road merging and departure. A 100 metre diameter tarmac area with acceleration roads allows for customer-specific junction layouts to be set up. There is also a general ADAS testing area of tarmac including custom road edge types.

For testing the connected aspect of CAVs, we have already installed three cell towers with our own 5G network, including a local network core. This both allows high-bandwidth, low-latency network services across the site when desired (e.g. for teleoperation), but also the opportunity to deliberately degrade access on demand for relevant test cases.

Complementing this cellular infrastructure we have twelve roadside units (RSUs) whose main function is to send and receive direct short-range (802.11p or C-V2X) messaging. These RSUs are connected via fibre to the site network, allowing fine-grained control of their behaviour and real-time data capture.

### 9.3 IDAPT V2X units

Each RSU contains our own IDAPT, a highly integrated computing unit aimed at CAV applications. These units have also seen in-vehicle service in our MuCCA<sup>10</sup> and CAVRide<sup>11</sup> projects as well as commercial customer deployments.

Each IDAPT combines a high-performance NVIDIA GPU-accelerated processing unit with on-board precision GNSS positioning, an inertial measurement unit, camera inputs, discrete I/O and a cellular modem. Crucially here though, it also includes a V2X modem, for short-range communications directly between vehicles and infrastructure units. Our software running on the IDAPT sends V2X messages (one-off or periodically) on command, generating real radio traffic which may represent many moving vehicles or static entities.

**Figure 9-2: IDAPT CAV-oriented control unit with integrated V2X**



Source: Author Generated (Applus IDIADA)

In the execution of this PoC, five such units, all "hearing" each other's V2X radio messages through the air, were run in parallel to give a true HiL representation of a cluster of real-world V2X stations. Every message was subject to ITS-G5 packaging (including data quantisation), radio transmission and reception (involving some latency and possible dropouts), and message unpacking. This therefore gave a realistic simulation of what would happen to message data passed between real vehicles on the road.

<sup>10</sup> As can be seen in <https://youtu.be/NHHyxm8Cdnc>

<sup>11</sup> As can be seen in <https://youtu.be/W7YtNiN0bT8>

**Figure 9-3: five IDAPT units running in parallel in bench tests**



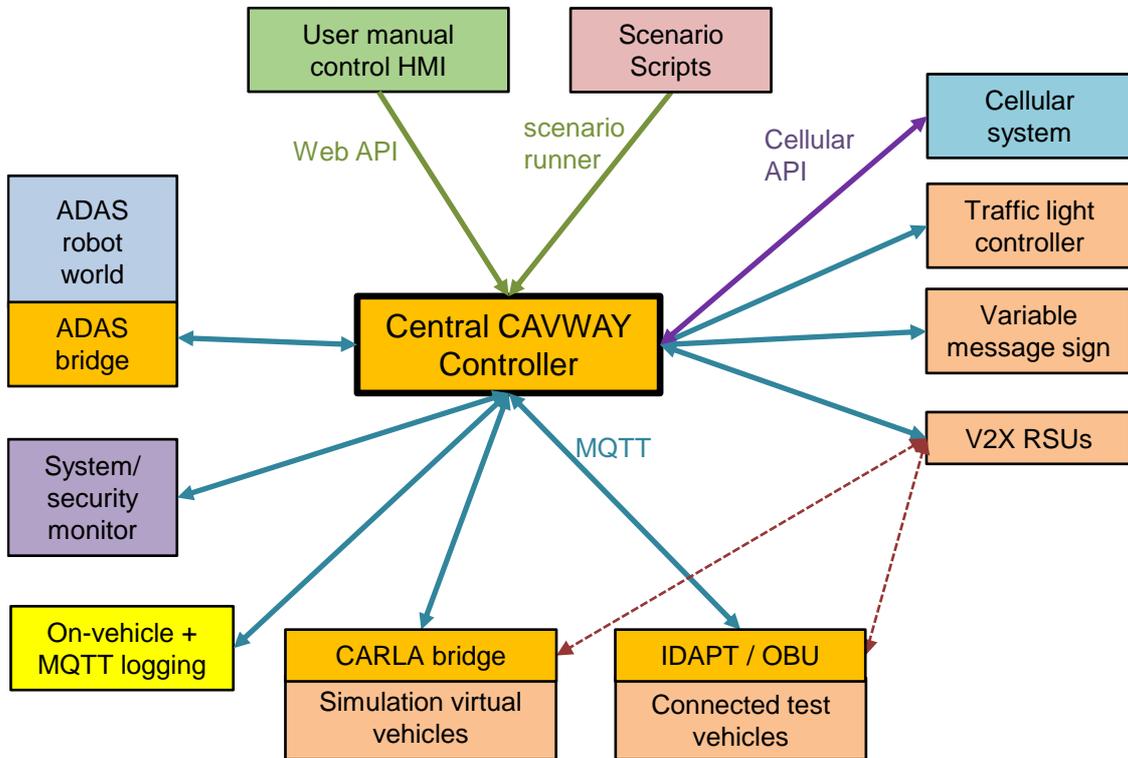
Source: Author Generated (Applus IDIADA)

## 9.4 MQTT-based control infrastructure

The various aspects of the CAVWAY electronic ecosystem are linked via a common MQTT data bus, a popular message-based middleware protocol. This flexible system allows individual devices to dynamically subscribe to topics of interest (e.g. vehicle positions, or control commands).

This system allows diverse systems to be interlinked in real time: web-based management and mapping tools, scenario test programs, ADAS test robots and "digital twin" simulations. In particular, our RSUs respond to MQTT commands to initiate and tailor their transmissions, and report back receive messages. In this way one RSU can be told to send messages on behalf of multiple moving vehicles, or interleave messages belonging to entirely different tests (using different IDs), all under central control.

**Figure 9-4: CAVWAY control system**



Source: Author Generated (Applus IDIADA)

## 9.5 Environment model

At our HQ in Spain, Applus IDIADA have partnered with rFpro and VI-grade to implement a high-fidelity immersive simulation of the proving ground. This has sufficient resolution to simulate the surface irregularities which bump and jolt car occupants via the suspension system, allowing (for example) the virtual design of chassis systems by trying different permutations of mathematically modelled components to feel their effect on the simulated track.

For CAVWAY, the priority was a visual representation of the site which could be used to plan and execute CAV scenario tests (e.g. merging and overtaking manoeuvres), even before construction is complete. For this a low-fidelity model based on design drawings was appropriate, but with proper road and lane definitions. This was built in-house by Applus IDIADA using the *RoadRunner* tool, which supports export to a variety of simulation tools including both the ASAM standard OpenDRIVE road definitions and visual elements represented as graphical mesh files.

**Figure 9-5: CAVWAY model in CARLA simulator and (top right) RoadRunner**



Source: Author Generated (Applus IDIADA)

We have imported this model into the popular CARLA simulator, which supports the rendering of different weather and lighting conditions. Being open source, we have been able to customise its content and behaviour, e.g. to introduce variable message signs. It is possible to virtually drive in the simulation (using the keyboard, or a steering wheel and pedals). By bridging to our MQTT ecosystem, we can also automatically spawn virtual representations of real vehicles on the track, or conversely ask RSUs or other entities on the real track to respond to the movements of virtual vehicles in the simulation, e.g. to spoof the moving V2X transmissions that should be "heard" from them, as broadcast by real RSUs distributed round the highway loop.

For this PoC, Millbrook shared their OpenDRIVE model of their "mile straight" track. Applus IDIADA added visual elements to this in RoadRunner, and exported it to Unity and CARLA formats to share among the other partners. This was a clear demonstration of the interoperability of environmental model data amongst the CAM Testbed UK organisations.

**Figure 9-6: Millbrook Mile Straight model built on OpenSCENARIO definition in RoadRunner design tool**



Source: Author Generated (Applus IDIADA)

## 9.6 OpenSCENARIO scripting

OpenSCENARIO is the ASAM industry standard language for defining simulation test scenarios, as used in this PoC to coordinate the EEBL scenario to exercise the subject vehicle as the System Under Test (SUT).

At Applus IDIADA, we have built on OpenSCENARIO with permitted language extensions so that we can, for example, send commands to entities on the proving ground in response to events such as vehicles closing distance or reaching a predefined point. This gives us a mechanism for coordinating reproducible tests not just in simulation, but in physical tests, or in combined "mixed reality" testing.

**Figure 9-7: OpenSCENARIO with custom command to transmit V2X for real**

```

.....<Event name="AmbulanceStartsDenms" priority="overwrite">
.....<Action name="AmbulanceStartsDenms">
.....<UserDefinedAction>
.....<CustomCommandAction type="SiteAction">
{"meta":{"destination":"rsul1","action_id":3,"fb":1,"period":200,"latest_time":"now+60000"},"denm":{"management":
{"eventPosition":{"device":"s.vehicle.volkswagen.t2"}}}}
.....</CustomCommandAction>
.....</UserDefinedAction>
.....</Action>
.....<StartTrigger>
.....<ConditionGroup>
.....<Condition name="WhenEgoWithinRange" delay="0" conditionEdge="rising">
.....<ByEntityCondition>
.....<TriggeringEntities triggeringEntitiesRule="any">
.....<EntityRef entityRef="s.vehicle.tesla.model3"/>
.....</TriggeringEntities>
.....<EntityCondition>
.....<RelativeDistanceCondition entityRef="s.vehicle.volkswagen.t2" relativeDistance

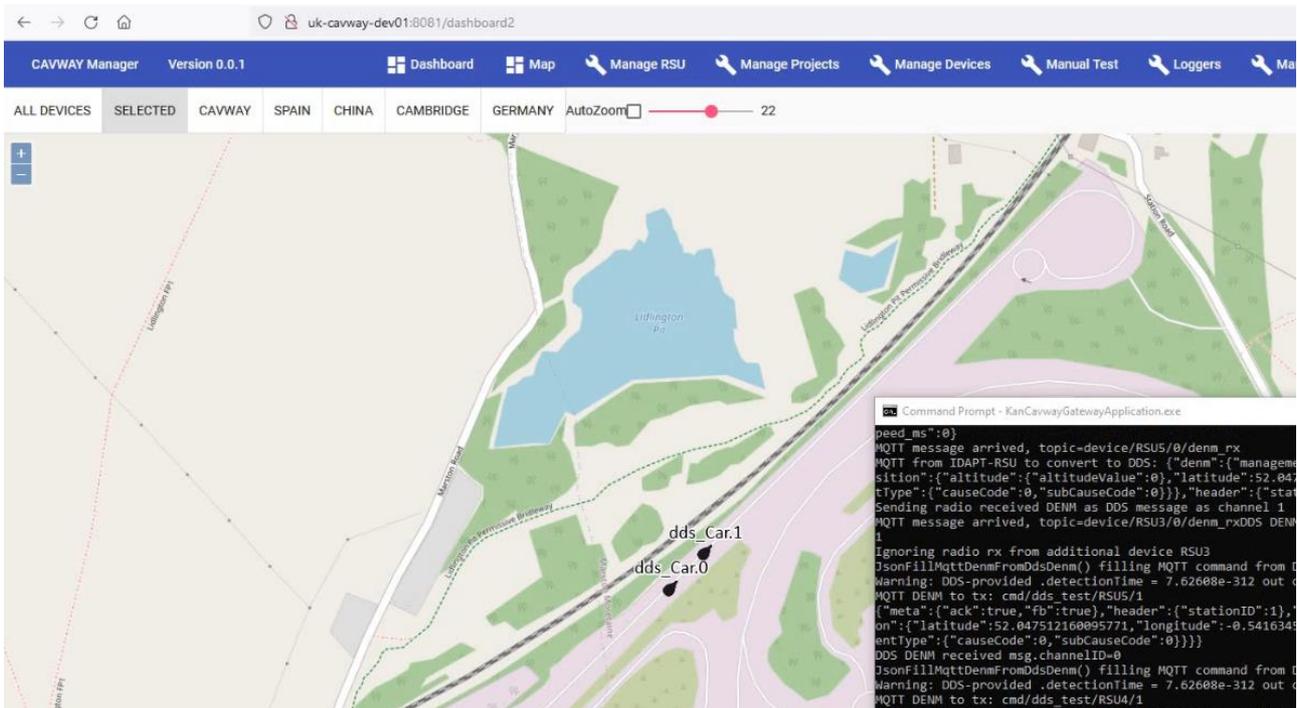
```

Source: Author Generated (Applus IDIADA)

## 9.7 Web-based management and mapping

The management of tests at CAVWAY is handled via a bespoke web-based interface, allowing access by users around the track and beyond. This includes mapping functionality, which displays any vehicles or other actors currently advertising themselves on the MQTT bus.

**Figure 9-8: CAVWAY web dashboard map showing simulation vehicles at Millbrook**

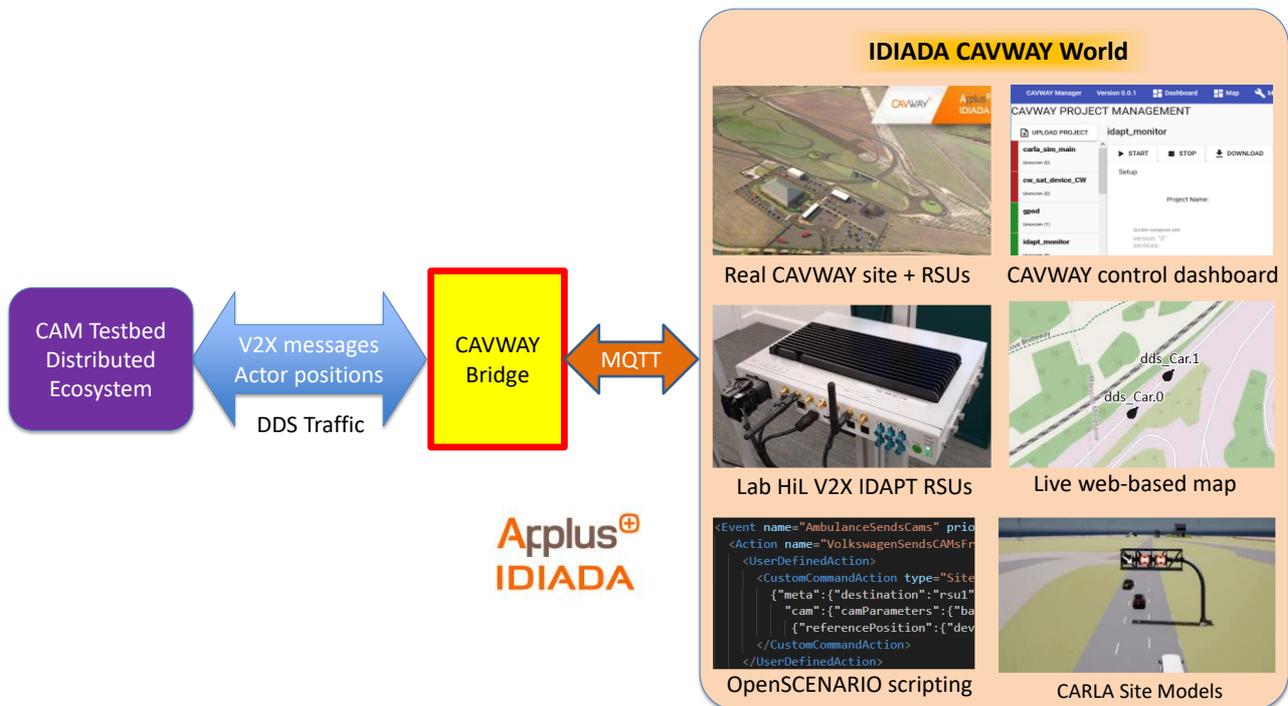


Source: Author Generated (Applus IDIADA)

## 9.8 Integration with Interoperable Simulation

While the CAM Testbed UK Interoperable Simulation used DDS as the middleware "glue" network messaging protocol, our Applus IDIADA systems use MQTT. Therefore to link the two ecosystems, we merely needed a "bridge" or gateway component to translate information between the two worlds. This fairly modest piece of work then opened up all of the Applus IDIADA ecosystem functions to data from the CAM Testbed UK world, adding "free" features on top of the actual requirements of the PoC.

**Figure 9-9: CAVWAY bridge links CAM Testbed UK DDS to MQTT ecosystem**



Source: Author Generated (Applus IDIADA)

For the PoC, the following types of data traffic were crucial in this integration:

- Commands to send V2X messages on behalf of the Forward Vehicle in the EEBL scenario
- V2X receive messages from another IDAPT, sent back to the EEBL Subject Vehicle
- Actor position and velocity streams, allowing local display/simulation of vehicles
- The messaging can easily be changed for future projects, as we have full control of the software source code.

The CAVWAY bridge component was implemented as a C++ Windows application, using a bespoke API from KAN Engineering to access a custom library to perform the DDS communications. The program employed an open-source MQTT stack to exchange messages with other devices in the CAVWAY ecosystem, acting as a bidirectional event-based gateway overall to minimise any additional latency. Applus IDIADA's LAN linked the CAVWAY components, while our firewall passed DDS traffic via the project cloud connection.

## 9.9 Summary

Applus IDIADA have been delighted to cooperate with other members of CAM Testbed UK in this PoC. It has shown that we are ready and able to tackle new concepts in simulation testing, and work with partners to jointly solve bespoke engineering challenges. We have leveraged work already in place for our CAVWAY proving ground to achieve this, with a particular focus on connectivity.

## 10 | University of Leeds: Pedestrian-in-the-Loop

By: Albert Solernou Crusat, Anthony Horrobin

### 10.1 Introduction

This section is a detailed report of the work that has been done on the University of Leeds side, starting with a description of the HIKER, following with the details of the software integration, a discussion about its technical limitations and potential solutions, and finishing with a series of conclusions.

### 10.2 The HIKER

The HIKER lab, shown in Figure 10-1, consists of a 9 by 4 metre walking space surrounded by 2.5 metres high solid glass walls with rear projection coming from an array of eight 4K projectors that provide a resolution of nearly 1 pixel per millimetre, and Hi-Fi speakers providing stereo sound.

Using a series of cameras, the HIKER tracks the position and orientation of the participant's head and adjusts the projection to match their point of view, reproducing the VR effect without the need to wear a headset, and thus giving a more natural and therefore more highly immersive experience. In Figure 10-1 demonstrated the HIKER cave lab as seen from outside with a participant ready to cross the street. The glass walls show the street from the point of view of the participant, and on top of them one can see six of the cameras tracking the pedestrian movements and one of the 4K projectors, projecting the image to the floor

The fact that the participants can see their own body and interact with the scene as well as with tracked objects, together with the fact that they can use their own mobile phones or any other tool, results in a highly immersive experience. Furthermore, it also results in a more accurate, split second resolution that is required when dealing with fine tolerances around vehicle-pedestrian impacts. This makes it a unique research asset especially suited for the study of pedestrian - vehicle interactions, while being widely flexible to accommodate other imaginative research applications.

### 10.3 Integration

In order to implement a co-simulation experiment we linked the driving simulator at UTAC Millbrook with the University of Leeds HIKER lab through the AWS infrastructure via a secure VPN tunnel.

The KANSM was used as the communication layer over Amazon Web Services (AWS), and also internally by the driving simulator at UTAC, and the HIKER connected to it through an API interface developed by KAN. In Leeds, the HIKER is powered by SimulatorD, the software that underpins most of the projects on the Virtuocity simulators, and which has been developed on-site over the last 25 years. While a standalone piece of software could have been developed specifically for this project, doing the integration through SimulatorD opens the door not only to

the HIKER but also to the rest of the simulators in Virtuosity including the largest, most advanced academic driving simulator in the UK.

**Figure 10-1: The HIKER cave lab**



Source: University of Leeds website

## 10.4 SimulatorD

Originally developed on-site for the University of Leeds Driving Simulator, SimulatorD now has now an architecture consisting of a set of loosely coupled simulation components that can be used flexibly and also replaced with third party software.

At its core, SimulatorD implements a networked message-bus together with a distributed database which acts as a real-time repository for a set of data objects, and facilitates the sharing of these objects across the simulation. Connecting to the message-bus, different nodes implement the different functional elements of the full simulator, publishing and updating objects to the central database and receiving and responding to changes published by other nodes. At the same time, each node may host additional services within the context of the node itself, without contributing to the communications overhead. This architecture allows the implementation of linkages between different systems, thereby allowing distributed simulation systems.

Hence, the integration was done with a newly implemented node, MillbrookBridge, that links the SimulatorD bus together with the DDS messaging system. This node, derives both from KANCore and from a SimulatorD messaging abstract class and is therefore capable of receiving and posting updates on the KANSM network as well as receiving and updating the shared SimulatorD objects "car" and "pedestrian". Thus, at every time-step, MillbrookBridge receives updates on the state of the vehicle at UTAC Millbrook and posts updates on the state of the participant in the HIKER.

## 10.5 Turner nodes

In order to display the graphical scene, SimulatorD runs as many rendering nodes as needed. In the HIKER, the eight projectors are connected to eight different computers with powerful GPU cards, each of them running a rendering or Turner node. Turner (Traffic in the Unity ReNderER) is a Unity3D-based project in which the University of Leeds implemented a network system that constantly updates the scene objects according to the SimulatorD shared objects. This network system delivers messages to the specific Unity objects, and thus it is used to update vehicle traffic, traffic lights, and in this case the state of the UTAC Millbrook vehicle.

While the same Turner rendering nodes are used in the University of Leeds Driving Simulator dome to display the scene from the point of view of the driver, it is more complicated in the HIKER since the participant (the pedestrian) is allowed to walk and approach the walls. In order to dynamically calculate the correct perspective, all the Turner nodes use Uni-CAVE (Tredinnick *et al.*, 2017) a Unity3D plugin for non-head mounted VR display systems. Thus, the set of VICON cameras feed the tracking system, and Turner acquires the data of the participant through Uni-CAVE, while the frustum is correctly updated in real time creating the immersive illusion.

The graphical UTAC Millbrook scene that is loaded by Turner was created by Applus IDIADA from a geospecific OpenDRIVE map using Matlab's RoadRunner, creating a digital twin of the UTAC Millbrook proving ground. Applus IDIADA's Filmbox (.fbx) file was loaded into Unity and converted into a self-contained Unity AssetBundle. Loading it into the right position was done manually, adjusting the position of a series of points in the scene. Automating and improving the accuracy of this process is an obvious missing step to achieve a quick and easy setup.

## 10.6 Discussion

In simulation experiments it is important to understand the key limitations of the system design. While the total walking space in the HIKER is finite, we believe that the most important factor in studying pedestrian-driver interactions is the latency of the messaging system, i.e., the one-way delay involved in a message travelling from the UTAC Millbrook driving simulator to the University of Leeds HIKER lab. Thus, in this section we analyse the latency that imposes the HIKER side, and discuss different optimisation approaches to mitigate it.

Latency of the different stages for a car message to reach the HIKER (top), and for a pedestrian, message to leave the HIKER (bottom) are indicated in Table 10-1. As discussed before, the greyed areas can be optimised out. Moreover, several of these operations happen concurrently, and therefore, the resulting latency is smaller than the sum of these values

**Table 10-1: Latency of the different stages.**

Incoming message path					
AWS → FortiGate	FortiGate → MillbrookBridge	MillbrookBridge → Turner	Unity Rendering	Projector refresh	
6 ms	KANCore doStep()	SimulatorD 2 x update()	Unity update()	8 ms	

Outgoing message path					
Cameras refresh	VICON Tracker	Unity Rendering	Turner → MillbrookBridge	MillbrookBridge → FortiGate	FortiGate → AWS
3 ms	1.5 ms	Unity update()	SimulatorD 2 x update()	KANCore doStep()	6 ms

Source: (Tredinnick *et al.*, 2017)

Regarding the physical network, internally, the HIKER uses gigabit ethernet switches, typically adding tens of microseconds to the latency, and a Fortinet FortiGate 40F firewall router to connect it to the University of Leeds campus network. From a machine connected to the router to AWS we measured a reliable 6ms one-way latency.

As depicted in Table 10-1, when an incoming message on the state of the remote vehicle arrives to the MillbrookBridge node, it takes one SimulatorD bus update call to update the state of the shared object in the central database and a second call to reach Turner. In the following FixedUpdate call, which happens to be every Time.fixedDeltaTime (Unity Technologies, 2022), Unity will render it in the new state and it will be visible in the HIKER after another 8 ms, which is the refresh rate of the BARCO F90 projectors in the HIKER (120 Hz).

Regarding the traffic in the other direction, and again depicted in Table 10-1 (bottom), the state of the HIKER participant is captured by VICON Vero cameras working at 330 Hz (3 ms), processed by VICON Tracker in 1.5 ms (Vicon, 2019), and fed into Turner so that it reaches the MillbrookBridge node after one Unity FixedUpdate call and two SimulatorD bus update calls.

In order to improve the overall latency, improvements may come both from the physical and the software sides. Regarding the former, ensuring that traffic goes through the Janet network (Janet Network, 2022) could possibly improve the latency for the AWS to the Fortinet router by a few milliseconds (Jisc, 2021; Open Science Grid, 2022), but the bulk of the potential optimisation resides on the software side.

Indeed, one could start by removing one of the two hops that a message reaching the MillbrookBridge node needs to get to Turner since SimulatorD already supports direct messaging between nodes. While this would be good for the incoming message to arrive to the 8 different Turner nodes, the outgoing message with the pedestrian data could be further improved. Indeed, a new sender could read directly from the VICON Tracker, adding very little overhead to the 3ms of the cameras and the 1.5 ms of Tracker.

Finally, the overall latency is not the sum of all these steps, as several of them happen concurrently. Along the incoming route, the first three steps run sequentially, but this set runs in parallel to the Unity update() call and the projector refresh. On the contrary, once the outgoing

route has been optimised, the messaging route is sequential, and thus the overall latency is the sum of the times. Given the low values of the final figures in both directions, this analysis demonstrates that it is technologically possible to obtain meaningful results in distributed co-simulation experiments across geographically separated sites.

### 10.7 Conclusions

The HIKER lab at the University of Leeds is a cave based immersive environment that allows safe and controlled Human-in-the-Loop experiments under conditions that may not exist in the real world, predicting their potential impact. By connecting the HIKER to a driving simulator we open the door to a broad range of co-simulation experiments that range from studying pedestrian-driver interactions to Pedestrian-Machine-in-the-Loop interactions. Moreover, by connecting the HIKER to the Driving Simulator in UTAC Millbrook we also open the door to the much richer use of resources that CAM Testbed UK can offer, and doing so with SimulatorD, we make the rest of the VirtuosityCity simulators available too.

Finally, using a careful analysis of the network and the messaging system, we demonstrated that the system can run at a very low latency, which is essential to co-simulation experiments. Future work includes sending more detailed pedestrian data. By using a series of body trackers, a more accurate representation of the pedestrian (an avatar) could be seen on the Driving Simulator side. Moreover, we could also capture and send video streams to include the facial expressions of the Humans-in-the-Loop, pedestrian and driver. These improvements would enhance the immersive experience and increase the accuracy of the interactions.

# 11 | SMLL: Scenario generation

By: Jonathan Clark

## 11.1 Introduction

As a member of CAM Testbed UK, interoperability is a key requirement to enabling SMLL customers to test across the full suite of testbeds in the UK and allow consistent and repeatable simulation results as well as ease of access to the facilities.

The phase 3 simulation project builds heavily upon the work that SMLL conducted in the phase 2 project and seeks to incorporate additional capabilities into the demonstrator, namely, the scenario generation data.

## 11.2 Objectives and role in the project

SMLL's role was to utilise its real-world scenario generation capability to provide scenarios to the demonstrator to meet the customer needs of AEB-P and EEBL testing. SMLL has developed a pipeline for taking real-world observable events from the SMLL roadside cameras and generating an executable OpenSCENARIO file that can be run within CARLA at the same location that it was detected. This methodology instantly allows SMLL to identify and run meaningful concrete scenarios in simulation.

The objective of the project was to be able to detect the two scenarios on the SMLL testbed and generate an OpenSCENARIO v1.0 file using agreed structures that can be executed at both the SMLL and also within the simulation suite at UTAC Millbrook. The simulation was run on a virtual model of the UTAC Millbrook "Mile Straight" and was the basis for the testing the customers AEB capabilities using ViL and V2X messaging using HiL from Applus IDIADA. The scenarios were then converted from Woolwich coordinates to UTAC Millbrook coordinates so that it could execute there.

## 11.3 The importance of real-world scenarios

Knowing which scenarios to test and validate against is crucial to virtually validating a system, this poses the following questions:

- How do we ensure that the scenarios are applicable and appropriate for the environment or ODD?
- Will the scenario occur in reality?
- How do we ensure that all scenarios have been identified and tested against?
- How do we scale scenario generation?

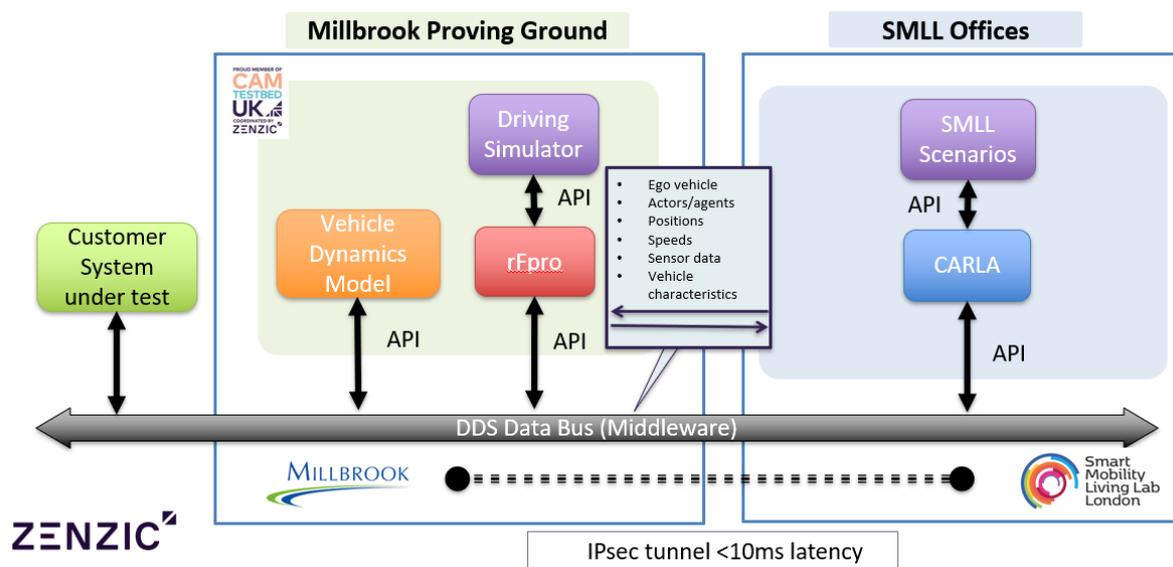
Identifying scenarios from real-world data can begin to answer these questions as it ensures that any scenario is representative including the manoeuvres and also the parameters that the scenario can operate within. It also broadens the diversity of the dataset by identifying scenarios

using camera infrastructure, thus, near miss and other scenarios that are not captured in collision or incident data can be detected and used to validate against.

## 11.4 Connectivity

In this phase we built further on the connectivity built in phase 2 which consisted of a site-to-site VPN between SMLL and UTAC Millbrook to share data between the two simulation environments. In phase 3, an IPsec tunnel was successfully established and tested between SMLL and the project AWS to allow for central communications for the DDS between all sites at CAM Testbed UK.

**Figure 11-1 - Original connectivity set up in phase 2 with the phase 3 inputs**



Source: Author Generated (SMLL)

## 11.5 Utilised capabilities at SMLL

SMLL has numerous simulation capabilities to offer the project:

- 1- Representative digital model
- 2- BIM data
- 3- Simulation environments (OpenDRIVE files)
- 4- Textured environment for VR
- 5- Crash data from iMAAP
- 6- TRL Driving Simulator (Digi-car)
- 7- TRL mobile Digi-truck and mini-simulators
- 8- Urban Traffic Control Instance (TRL UTC)
- 9- Representative Signal Phase and Timing data
- 10- Traffic model of routes (VISSIM)
- 11- Autoware / Aslan CAV software stack
- 12- Fixed camera infrastructure (300 cameras)
- 13- Object detection data from SMLL machine learning software
- 14- CARLA simulator (SMLL modified left hand drive)

The bottom 4 (11-14) capabilities were proposed for the project with 12-14 identified for further development and integration with the demonstrator building on phase 2 which incorporated 1,3,8,9,10.

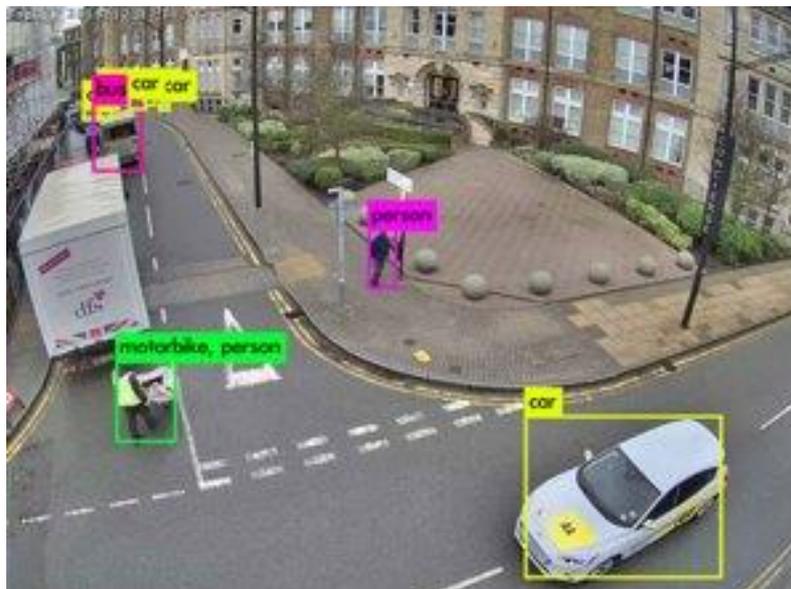
### Camera infrastructure and routes

The SMLL has 300 cameras (and growing!) across its 24km of routes. SMLL can use these cameras to monitor the specific behaviors of either test vehicles or of public road users. The cameras were installed in particular locations to ensure optimised coverage of road features was achieved e.g. bus stops, junctions, cycle lanes, mini roundabouts; across a wide range of road types including: urban, inter-urban, residential, dual-carriageway and semi-private roads.

### Object detection

SMLL has deployed real-time object detection on its cameras to capture the positions, speeds and classifications of road users across its routes. The object detection is built and developed by SMLL using open-source software YOLOv4 and darknet. This is a neural network designed to detect and classify objects from real-time video. The software has been subsequently trained to improve the reliability of classification with over 49,000 images. Each object is then tracked across the image and optimised for each class type. After the detection and tracking, the information is pushed to the localisation module. The object centroid is transformed to real world coordinates which enables video to be recorded based on triggers such as position, class type, acceleration and speed. These triggers are defined based on the type of scenario we are trying to capture. Currently these are pre-determined and can be set strictly or loosely to open or close the net for the scenarios that we capture. We could capture every hard braking event that is seen on the testbed, which would generate a large dataset of videos that would include both AEB scenarios and EEBL along with other less relevant scenarios such as sudden light changes, cut-ins etc.

**Figure 11-2: Example object detection on SMLL cameras (Royal Arsenal, Woolwich)**



Source: Author Generated (SMLL)

### **CARLA, OpenDRIVE, maps and left-hand drive**

SMLL has modified the source code of CARLA so that all road users can operate on the left-hand side of the road. This is vital to executing the scenarios that are from real routes within the UK.

The simulation uses the SMLL OpenDRIVE files that map the routes in SMLL and define the lane widths, stop lines and junctions rules, providing the basis for running and executing the OpenSCENARIO files.

Within simulation, SMLL has 9km of accurate geometry enabling a simulation of complex areas of SMLL. Figure 11-3 depicts a visual representation of the OpenDRIVE file for a complex junction outside the SMLL Woolwich office.

**Figure 11-3: Visualised OpenDRIVE file on SMLL 3D model**



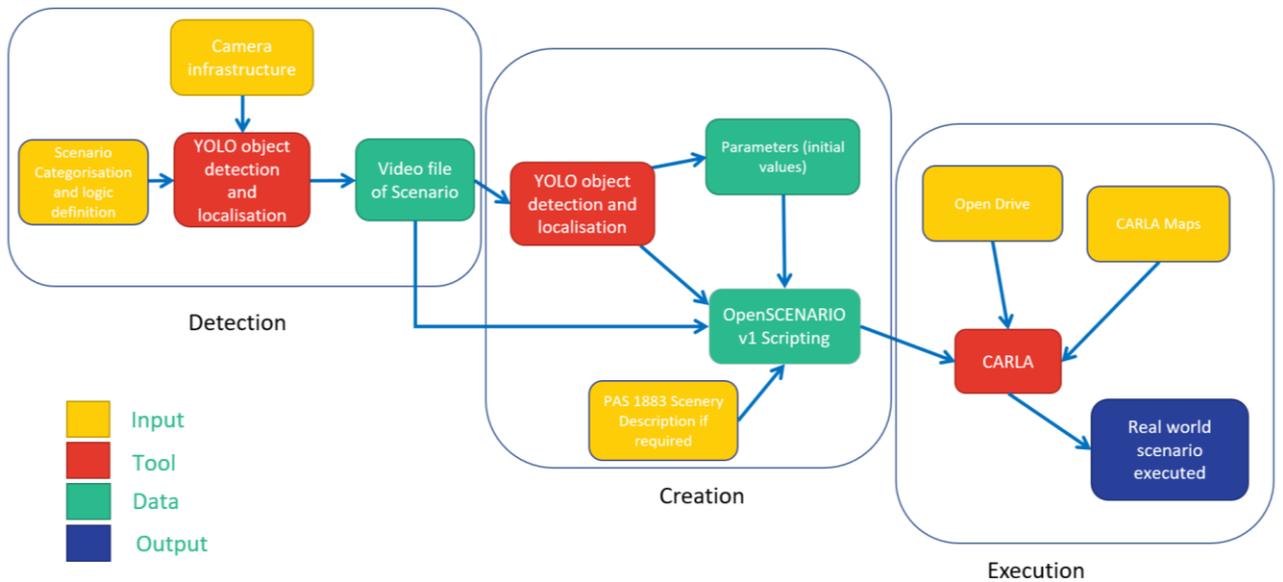
Source: Author Generated (SMLL)

## **11.6 Scenario generation pipeline**

SMLL have developed a PoC scenario generation pipeline that can be used to detect, create and execute real-world scenarios in simulation. Currently this method creates the individual test cases or concrete scenarios that are then run in the same 3D representation of the environment.

Figure 11-4 depicts the high-level workflow for SMLL’s scenario generation pipeline:

**Figure 11-4: High-level data flow of scenario generation**



Source: Author Generated (SMLL)

**Logic for scenario capture (Detection)**

Using basic logic, scenarios are captured based on what is occurring in the scene. This stores 1 minute video files that are then used for processing in the creation stage. The logic can be dependent on the vehicle information:

- Speed in approach, acceleration or deceleration to a location
- Proximity to points in the road
- Class type i.e. pedestrian

Using combinations of this logic, and the boundaries of acceptable values, a library of scenarios can be collected.

**11.7 Detecting EEBL**

The standard ISO 20901:2020 stipulates test requirements for EEBL. These are shown in Figure 11-5.

**Figure 11-5: The ISO test requirements for EEBL**

	Test case 1 FV transmission and delay	Test case 2 False positive	Test case 3 True positive
Vehicle speed* (V <sub>1</sub> )	60 km/h	60 km/h 80 km/h	60 km/h 80 km/h
FV deceleration	2 to 3 m/s <sup>2</sup> and over 5 m/s <sup>2</sup>	2 to 3 m/s <sup>2</sup>	Over 5 m/s <sup>2</sup>
IV involved	None	None	None
Situation description			

NOTE The test vehicle speed indicated has a tolerance of 5 km/h.

Source: (BSI, 2020a)

To detect a real scenario applicable for EEBL on the SMLL testbed, the parameters for test case 3 were intended to be used as the inputs for the recording trigger. However, this didn't result in many videos being captured as the speed limit in most areas is 30mph. The decision was made to use the deceleration parameter only which resulted in the capture of a Forward Vehicle (FV) braking for a pedestrian crossing and the Subject Vehicle (SV) braking hard behind it.

### 11.8 Detecting AEB-P

To capture this scenario, basic logic about the vehicle movements and classification were used. Firstly there needed to be a pedestrian in the scene along with a vehicle, this was applied with a hard deceleration of >5m/s<sup>2</sup> to capture a car braking hard for a pedestrian stepping out at pedestrian crossing shown in Figure 11-6.

**Figure 11-6: AEB-P scenario**



Source: Author Generated (SMLL)

### **Scenario creation**

Once the scenario has been detected the video can be run through the object detection algorithm once more to generate information about each object at each frame along with an annotated video. The information is stored in CSV file containing the speeds, positions and, crucially, time.

Through a combination of using the data in the CSV file and the annotated video, the following is identified:

- The Ego Vehicle
- Key agents in the scene
- Manoeuvres

This information is then used to construct an OpenSCENARIO v1.0 file using actions only. A library of manoeuvres is called upon to build the scenario with the initial conditions and triggers. The timing data and positional data allows for triggers in the OpenSCENARIO file to be scripted appropriately so that it can be executed with a vehicle under test. Data about the lighting conditions and weather can be input based on the data in the video.

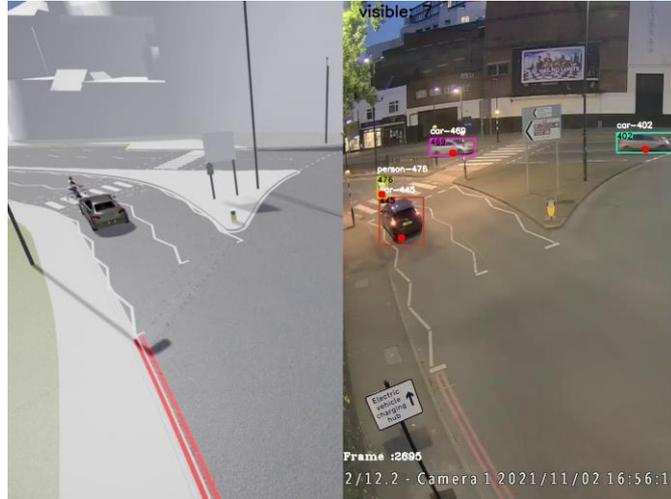
Parts of the scenario can then be parameterised to further extend the testing of the system, for example varying the speed of the Ego Vehicle on approach or different environmental conditions.

### **Execution (testing)**

The developed scenarios were executed in SMLL's CARLA simulator to test and debug. ScenarioRunner was used to execute the scenario, the automatic\_driving.py was also used as the Ego Vehicle in this instance. This operates the car with basic CARLA driving rules as per the OpenDRIVE file. Each scenario is executed in the exact same location in the simulation as where it was detected. This can become an extremely powerful tool to extend testing, de-risk real-world trials and even virtually certify any vehicles in the future.

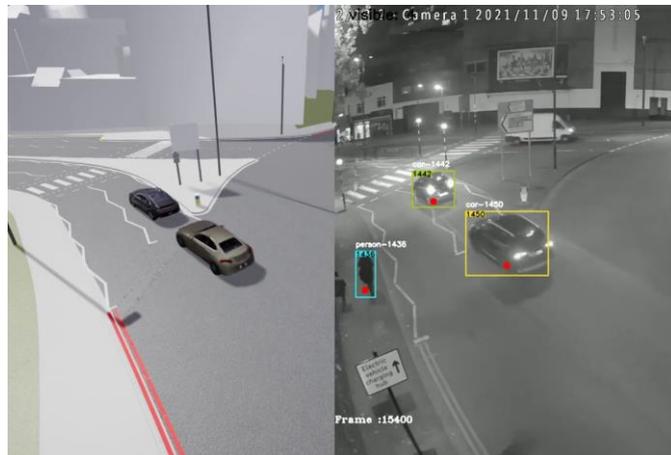
Figure 11-7 and Figure 11-8 show the two scenarios running in CARLA vs. the real scenario. These images showcase the strong correlation between the simulation and the real-world video captured.

**Figure 11-7: AEB-P scenario CARLA (left) Real world (right)**



Source: Author Generated (SMLL)

**Figure 11-8: EEBL scenario CARLA (left) Real world (right)**



Source: Author Generated (SMLL)

## 11.9 Interoperability

OpenSCENARIO was selected as the format for scripting the scenarios due to its prevalence in the ADS and ADAS industry and for its ease of execution within CARLA. To demonstrate that the scenarios are interoperable, after verification within SMLL CARLA, they were subsequently executed in simulation at the UTAC Millbrook simulation facilities on UTAC Millbrook’s “Mile Straight”. The coordinates for the objects were translated to be able to be executed there for testing with Ford and for testing with CAVWAY’s V2X HiL. To ensure that it could be executed, the consortium parties agreed the structure of OpenSCENARIO file to use as actions only for all OpenSCENARIO files created. This meant that the same execution could be done at all sites with only the translation of coordinates required.

## 12 | HORIBA MIRA: Oracle module

By: William Latham

### 12.1 Introduction

In simulation it is necessary to assess the behaviour exhibited when a SUT (System Under Test) is confronted by a given driving situation. Metrics and oracles can be used to facilitate the majority of scenario analysis. After performing analysis, oracles produce a final SUT score for each test case. Rather than relating to an arbitrary good/bad threshold, this score can be used to identify those test cases most pertinent for further review.

The challenges of building oracles, the approach taken by the HORIBA MIRA Oracle module and ultimately its capabilities are discussed in 12.2 and 12.3. How these capabilities enrich CAM Testbed UK and how being part of the testbed is accelerating the Oracle module's capabilities is described in 12.4. Integration of the HORIBA MIRA Oracle into CAM Testbed UK is discussed in 12.5. Finally, oracle evaluation results and a closing discussion are outlined in 12.6 and 12.7, respectively.

### 12.2 Oracle challenges and background

It should be noted that multiple distinct categories of oracles can exist within the HORIBA MIRA Oracle module, each addressing an overarching property to be assessed, all of which will benefit from high-quality CAM Testbed UK data. These might include comfort, progress, etiquette, legality, etc. Such a diverse set of oracles are ultimately necessary as, like a human driver, a SUT must trade-off competing priorities and thus different oracles are required to capture these different aspects of performance. However, this section mainly focuses on HORIBA MIRA's safety oracle.

The sheer mechanics of generating a "final score" are not that challenging on paper: combining a series of peak values, averages and other statistics can yield a result. However, to achieve meaningful, correlated scores that are correctly oriented and indicative of driving outcomes, sophisticated solutions must be employed to overcome the inherent challenges of working with large datasets. Those challenges and solutions are discussed in this section.

There is a lot of information hidden within raw ground-truth data. Test data may be available not only across all scenario times but for all road users (ego, other vehicles and pedestrians), each of which has its own extensive range of tracked and recorded parameters. This presents three main challenges:

1. Temporal nature of test data: data quantity scales linearly with scenario duration. This may result in an SUT appearing to perform worse in longer scenarios (longer scenarios = more time for negative events to occur). In contrast, a SUT could be falsely rewarded/punished less for completing some scenarios more quickly than others (shorter scenarios = less time for negative events to occur)
2. "Tailedness" of scoring: should a SUT be rewarded for positive outcomes **and** punished for negative outcomes (two-tailed)? Or, should the SUT only be subject to the latter (one-

tailed)? The possible weakness of a two-tailed approach is that a mixture of good and bad scores may appear mediocre when averaged; for this reason, the HORIBA MIRA Oracle module takes a one-tailed approach.

3. Scoring calibration/correlation: processing vast amounts of ground truth data into a few simple scalars results in information loss and blurring. This presents the risk of oracle scoring being inconsistent with human observations. Therefore, the quantitative reduction in data must be reconciled through a sophisticated, multi-tiered workflow.

One facet of the HORIBA MIRA Oracle module is to reduce the data in two distinct steps:

1. Churning ground truth data into metrics.
2. Processing metrics into oracles.

The focus and mitigations through the design of each of these steps are different and deliberate. Metrics aim to extract key indicators of performance and context, which make the task of scoring an oracle more tractable since the oracles can be fed with a directed subset of all the test data whilst also enabling parameter variation and trade-offs to be incorporated.

## **Context**

Construction of performance metrics and oracles alone is largely devoid of meaning without the ability to contextualise and possibly “explain away” those performance measures under certain conditions. Viewed through the lens of oracle scores, context presents extra information about the circumstances of a scenario, where scores/metrics would be challenging to interpret alone. E.g. “was that a good score for the SUT, given what happened in that scenario?”.

Performance metrics focus on an ego’s dynamics, proximity and interactions with other road users, context describes elements other than the ego: other actors, pedestrians, weather, and even more macroscopic aspects such as traffic density and competing performance outcomes (e.g. pragmatic considerations regarding the safest outcome vs. a safe but more natural, progressive outcome).

The HORIBA MIRA Oracle module combines event/feature identification with some insight into the intentions of all the vehicles in a scenario. Specifically, the module decomposes scenarios into a finite set of discrete driving manoeuvres. A basis of manoeuvres offers a convenient view of driving, falling between features/events and intentions. Unlike performance metrics, contextual information is inherently qualitative. Therefore, the challenge of context is in meaningfully combining, correlating and overlaying qualitative and quantitative data.

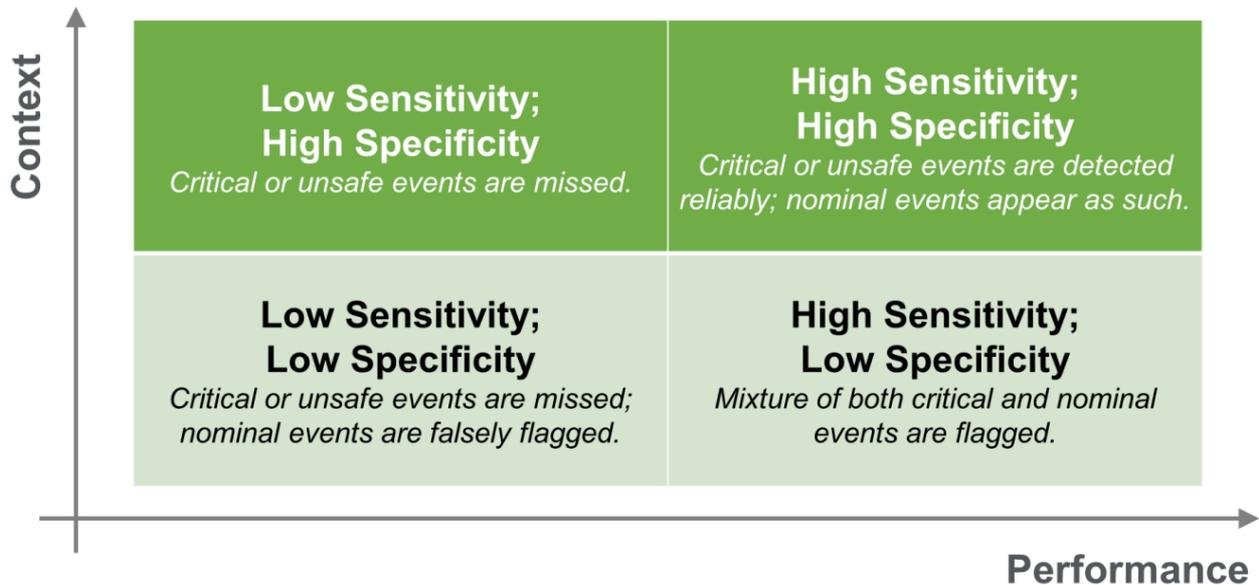
Ultimately, context does not replace human input; however, it enables intelligent labelling of ground truth data and thus the interrogation of large data sets inside a wider test programme, positively impacting coverage and completeness arguments.

## **Sensitivity and Specificity**

No “test” is perfect; that is, no test is wholly reliable at all times. A key challenge in assessing the performance of the SUT is assuring both the Sensitivity and Specificity of that assessment. These borrowed terms are common in medicine and epidemiology yet are totally synonymous with the notions of true/false positives/negatives.

A safety oracle must be sensitive to unsafe scenarios and specific about those which are safe. Figure 12-1 indicates the outcomes with which good/poor sensitivity and specificity are associated.

**Figure 12-1: Language common in medicine to describe the required properties of its holistic assessments of the SUT.**



Source: Author Generated (HORIBA MIRA)

Sensitivity is the ability to detect those scenarios, and moments within scenarios, which a “perfect” test would identify; in other words, to flag what should be flagged, and to capture all “positives” as true positives. Specificity is the opposite: the ability to rule out (ignore) those scenarios and moments in scenarios that are safe, uncritical, and uninteresting; in other words, the reliability of the assessment to not flag false failure modes.

Incorporation of a large set of relatively unsophisticated performance metrics is seen as the port to sensitivity; incorporation of a set of distinct, context metrics is seen as the port to specificity when combined with performance metrics data.

### 12.3 HORIBA MIRA Oracle workflow

This section provides a high-level description of the sophisticated multi-tiered approach required to solve the “scoring calibration/correlation” challenge discussed in 12.2.

1. A ground truth data record is received containing all the **raw** ground truth (RGT).
2. The RGT is re-interpreted, in part, to generate **processed** ground truth (PGT). The PGT contains information that is not present in the RGT, such as the causal and diagnostic evidence needed to apply context metrics.
3. The PGT is analysed, and performance/context metrics are calculated.
4. The metrics inform algorithms, which determine the global, scalar, oracle scores for each scenario.

Figure 12-2 displays a high-level overview of the test analysis workflow, wherein data is channelled and “cascaded downwards” into the outcome from this procedure: the oracle score(s).

**Figure 12-2: Visualisation of how the vast and difficult-to-interpret ground truth data are “cascaded downwards” into oracles via the intermediary calculation of many metrics.**



Source: Author Generated (HORIBA MIRA)

### Ground Truth

The first step in the workflow is to extract the RGT from the simulation environment, which is then stored in the HORIBA MIRA Oracle module’s database before being used to generate the PGT.

### Performance Metrics

A range of performance metrics are calculated as timeseries traces after each scenario has yielded its PGT. Headline performance metrics include but are not limited to: Brake Threat Number (BTN), Steering Threat Number (STN) (Sjöberg *et al.*, 2010) and PEGASUS Criticality Score (Junietz *et al.*, 2018)

Each oracle score is a determination made by combining many forms of information and evidence in the form of metrics.

### Scenario Decomposition

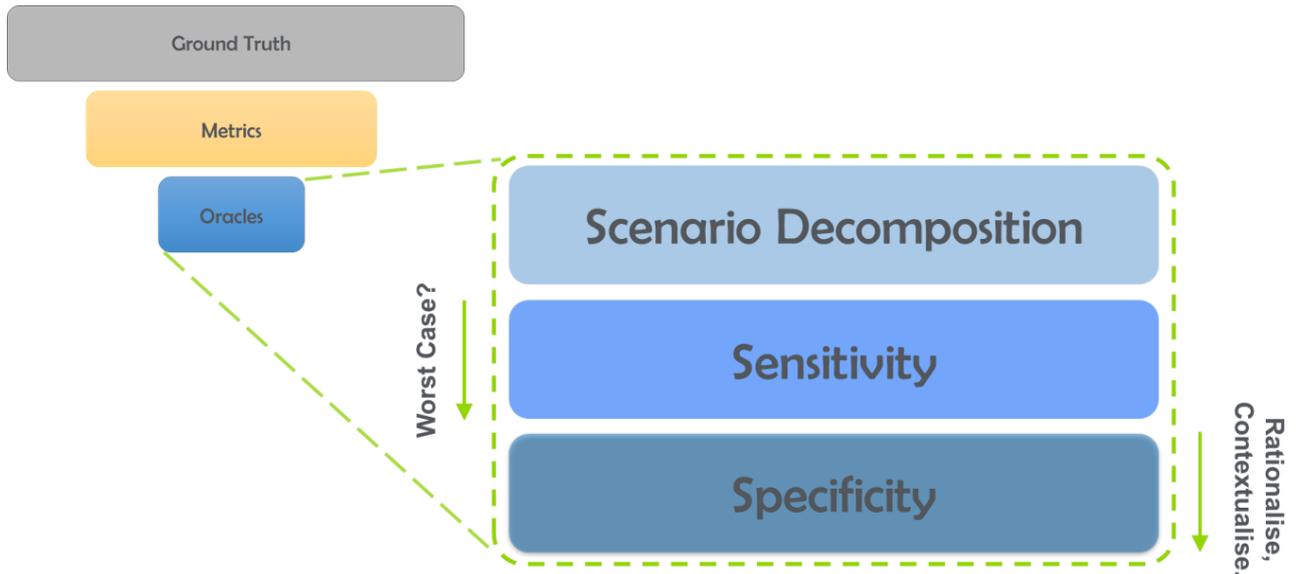
Scenario Decomposition is the process of extracting the scenario “storyboard”; that is, a sequence of high-level features and events that, when taken chronologically, define the scenario as a whole journey/story. The scenario “storyboarding” forms a large part of any context evidence that shall be considered when assessing overall SUT performance; manoeuvre detection is one method by which to decompose scenarios.

### Processing Metrics into Oracles

The multi-tiered approach described in Performance Metrics, Scenario Decomposition and Sensitivity and Specificity is described in Figure 12-3.

Scenario Decomposition is a scenario verification and/or event/feature extraction process, which informs the applicability of certain metrics, and conditions the subsequent oracle evaluation according to the particular nature of the scenario in question. The sensitivity stage targets identification of critical and unsafe moments in the metrics’ traces. The specificity stage is designed to rationalise, contextualise and explain away any artefacts in the metric data and ensure that the final score is as representative of the performance by the SUT as is feasible.

**Figure 12-3: Visualisation of how an oracle score evolves.**



Source: Author Generated (HORIBA MIRA)

## 12.4 Integration benefits for HORIBA MIRA and CAM Testbed UK

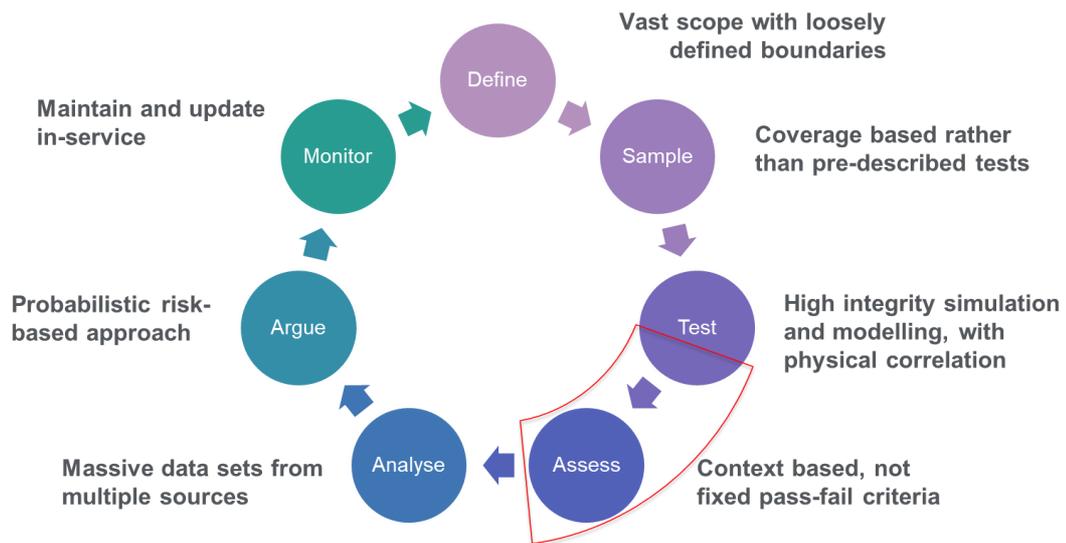
Developments in simulation and test automation allow large volumes of scenarios to be executed, but the resulting test data must be analysed accurately and efficiently to assess system performance and inform overall test coverage. This need is recognised within the Zenzic concept architecture, but no data post-processing was implemented within the Phase 2 PoC development. Without satisfying this need, it is less likely that Zenzic will succeed in presenting the UK as offering a comprehensive simulation capability.

HORIBA MIRA aimed to address the need for efficiency in assessing system performance and informing overall test coverage by integrating key functions from the VeriCAV project with the KANSM – namely the HORIBA MIRA Oracle module. The result of which is twofold: Increased focus on safety through detailed virtual testing and validation when using this testbed in its current form, and also paving the way for the UK CAM industry as a whole. This integration is made possible by the oracle module being independent of any test method, meaning it can be reused many times.

Figure 12-4 illustrates a complete scenario-based testing workflow developed by HORIBA MIRA with AESIN and Automotive Council UK. Each circle represents a critical step of the process but also encompasses a set of technical challenges still to be resolved in order to robustly validate new CAV technologies. The focus of HORIBA MIRA’s contribution is highlighted with a red

boundary: the assessment of test data through test Oracles to evaluate system performance with respect to the context of the specific test scenario.

**Figure 12-4: Scenario-based testing workflow developed by HORIBA MIRA with AESIN and Automotive Council UK. Elements outlined in red represent the primary focus of HORIBA MIRA’s contribution.**



Source: Author Generate (HORIBA MIRA)

HORIBA MIRA aimed to extend CAM Testbed UK’s capability with the ability to post-process and evaluate test data, thus providing end-users with insight into a given SUT performance, directing their attention towards the most critical areas for review. By identifying those areas of poor performance, engineers are able to understand and improve upon the limitations of the SUT more efficiently, meaning customers will benefit from faster time-to-market product development, thus bringing capability value.

During this phase, HORIBA MIRA has been able to feed more high-quality data into its Oracle module. This data is enabling HORIBA MIRA to refine its Oracle module further, leading to more accurate and efficient highlighting of those areas of poor CAV performance that require focused engineering development effort. Additionally, this data will be invaluable in future, as HORIBA MIRA looks to address further performance aspects, such as progress and roadcraft/etiquette. By improving and developing new Oracles, CAM Testbed UK will become more powerful and robust, creating a better toolchain that brings the UK closer to offering simplified access to comprehensive simulation capability.

A key innovation of VeriCAV (Levermore and Peters, 2020) was the capability to take feedback from the Oracle module and input this data to HORIBA MIRA’s Test Generation module. The Test Generation module would then be informed by the Oracle module’s learning and use that knowledge to intelligently generate new scenarios. Whilst out of scope for in this phase, looking forward, the enabler to integrate this functionality at a future date would be the further

development of the Oracles, which, again, the high-quality data from the testbed is contributing towards.

## 12.5 Integration

### Interface background and scope

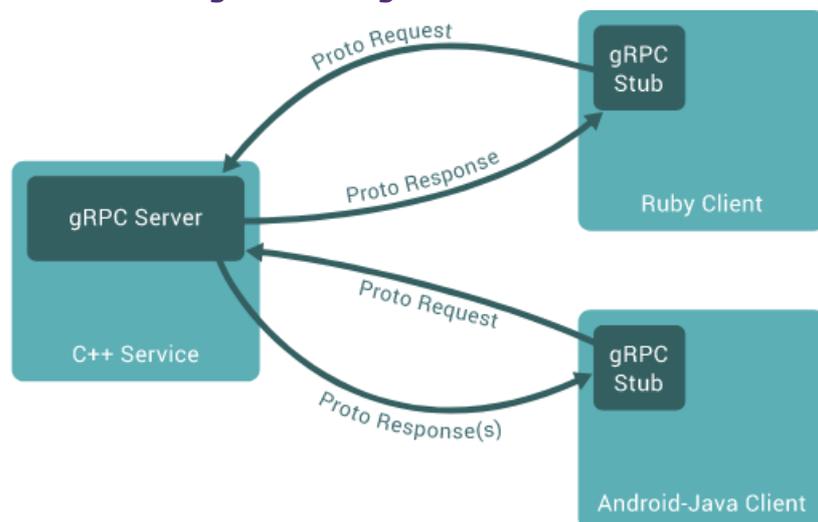
Prior to Phase 3, HORIBA MIRA developed its oracle and scenario generation capabilities as part of the VeriCAV project. Therefore, the HORIBA MIRA Oracle and Scenario Generation modules existed within and were coupled to the VeriCAV framework. Test programmes were communicated via a Scenario Interface and test data via a Ground Truth Interface. Ultimately, the framework would handle the distribution of ground truth and test program data to and from the Oracle and Scenario generator, respectively.

It was decided that the scope of HORIBA MIRA’s work in this phase would be limited to the integration of the Oracle module and, therefore, the Ground Truth Interface. However, as discussed in Section 12-4, refinement of the Oracle module and its oracle algorithms will act as the enabler to potentially integrate the Scenario Generation module in future.

### gRPC/proto3 common interface

In the VeriCAV framework, the Ground-truth Interface was built using gRPC (google Remote Procedure Call). In gRPC, services are defined and implemented on the server-side, whilst clients call these services. Each service specifies required input parameters and return types, which are defined using Google Protocol Buffers 3 (proto3), which can be thought of as clients sending “Proto Requests” and receiving “Proto Responses”. This is visualised in Figure 12-5 Protocol Buffers are a platform-agnostic solution that allows structured data to be serialised, subject to predefined schema files, ensuring consistency and confidence for both partners, either side of an interface.

**Figure 12-5: gRPC visualisation**



Source: (gRPC Authors, 2022)

Various APIs and data formats would be appropriate for communicating ground truth data from the testbed to the HORIBA MIRA Oracle module. However, as discussed, because gRPC and proto3 were successfully deployed in the VeriCAV framework, HORIBA MIRA and the System Integrator (KAN Engineering) agreed that the most efficient solution in this phase would be to adapt and deploy a similar interface between the testbed and the HORIBA MIRA Oracle module.

The HORIBA MIRA Oracle module does not require ground truth data to be sent in real-time. Instead, the module ingests an entire ground truth dataset once a scenario has been executed. Therefore, at this stage, a mechanism to automate data exchange in real-time with the testbed was not strictly necessary in the context of a PoC. For this reason, the HORIBA MIRA Oracle module does not interface with the testbed's cloud solution to receive data from the testbed. Instead, binary data is manually shared. However, closer integration remains a possibility in future as the testbed continues to mature. In effect, this saw the System Integrator manually sharing ground truth data in the form of protocol buffer binary messages, with HORIBA MIRA operating both the server and client side of the gRPC interface, using the shared binary messages as input.

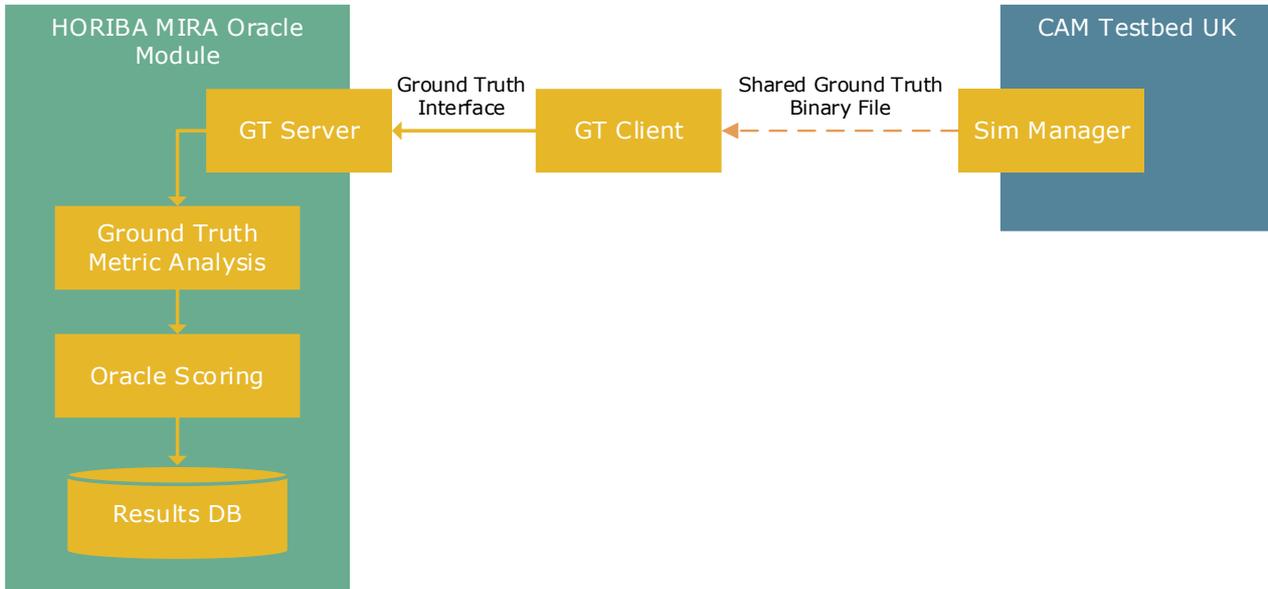
### **Integration work**

Integration efforts for the System Integrator focused on supplying HORIBA MIRA with the ground truth binaries. The System Integrator first took scenario ground truth data generated by the testbed. They then converted and encoded that data into Ground Truth Interface proto3 binary messages, thus complying with the agreed proto3 schema.

As mentioned at the beginning of this section, the HORIBA MIRA Oracle module was heavily coupled to the VeriCAV framework, which it relied upon to receive and distribute ground truth messages. In order to integrate the Oracle module with the testbed, HORIBA MIRA's efforts focused on decoupling the module from the broader VeriCAV framework. To achieve this, a new gRPC-based Ground Truth Interface was developed.

Like the VeriCAV Ground Truth Interface, the new Oracle module interface communicates ground truth binaries. However, it differs in allowing external platforms to communicate ground truth data directly to, and interact with, the Oracle module. This is visualised in Figure 12-6: ground truth binary files are shared with HORIBA MIRA, who then use the GT (Ground Truth) client to send Ground Truth data to the Oracle module, which in turn performs metric analysis, oracle scoring and finally stores results in a database.

**Figure 12-6: Visualisation of interface architecture**



Source: Author Generated (HORIBA MIRA)

## 12.6 Results

Example simulation data was provided for AEB-O and AEB-P functions to validate the integration of the HORIBA MIRA oracle. In the AEB-O scenario, the 'Ego Vehicle' progressed along a straight road before colliding with a slow moving 'Traffic Vehicle'. There were 6 test case variations with different speed profiles for the Ego and Traffic Vehicle, which in turn affected the response of the AEB system and therefore the deceleration and impact speed of the Ego Vehicle. The variation parameters and oracle results can be seen in Table 12-1.

**Table 12-1: AEB scenario variation results**

Variation	Ego Vehicle speed (km/h)	Traffic Vehicle speed (km/h)	Safety Oracle score
1	60	2	-27.3
2	70	2	-15.1
3	85	4	-29.9
4	90	4	-27.1
5	100	5	-144.9
6	105	5	-63.3

Source: Author Generated (HORIBA MIRA)

From the above table, it can be inferred that the safety of the Ego Vehicle response is heavily dependent on the speed profiles of the two vehicles. Whilst the general trend indicates that

higher vehicle speeds are associated with worse safety performance, it can be observed that the optimal case for the AEB system was 70km/h and 2km/h for Ego Vehicle speed and Traffic Vehicle speed respectively. In contrast, the worst case was 100km/h and 5km/h for Ego Vehicle speed and Traffic Vehicle speed respectively.

Clearly any vehicle collision is not good. However, when observing the Safety Oracle results, it is important to appreciate that these scores do not focus on indicating absolute performance but rather act as a tool to enable engineers to more efficiently direct their test and analysis efforts to those areas with particularly poor or unpredictable performance. This would otherwise become an increasingly difficult and inefficient task as datasets expand to include thousands of tests.

Two variations of an AEB-P scenario, and the associated safety oracle results, can be seen in the table below. Whilst a collision occurred in both variations, the Ego's safety performance is worse when a pedestrian walks out in front of the Ego, rather than when the pedestrian is already in front of the Ego vehicle and is walking in-line with traffic.

**Table 12-2: AEB-P scenario variation results**

Variation	Pedestrian behaviour	Oracle score
1	Pedestrian walks along road, in-line with traffic.	-20.9
2	Pedestrian crosses road, walking out in front of Ego.	-55.3

Source: Author Generated (HORIBA MIRA)

## 12.7 Discussion

The HORIBA MIRA Oracle Module has been successfully integrated into the wider simulation framework. This has added to the capability of CAM Testbed UK, improving the accuracy and efficiency at which system performance can be assessed, providing a tool to isolate and interrogate areas of poor performance for a given SUT. In particular, this has been demonstrated in the evaluation of example AEB scenarios to prove the concept and validate the integration.

It is important to stress that the field of Oracles in general, and the HORIBA MIRA Oracle module specifically, is continuously evolving. However, the rate of evolution can be limited by data availability and quality. Integration with the CAM Testbed UK simulation framework opens up new channels for collecting and analysing test data, further accelerating the development of the HORIBA MIRA Oracle and the benefits it can bring. This will not only improve the Safety Oracle's accuracy but support the development of new oracles, contributing to a broader, more diverse evaluation toolchain.

## 13 | WMG: V&V pipeline

By: Jason Zhang for WMG Verification & Validation team

### 13.1 Motivation of the V&V project

The recent advancement in Automated Driving Systems (ADSs) and Advanced Driver Assistance Systems (ADASs) technologies is driven by the many benefits they offer, such as increased road safety (Fagnant and Kockelman, 2015), increased traffic throughput (le Vine *et al.*, 2016), reduced emission levels (Fagnant and Kockelman, 2014), and decreased driver workload (Balfe, Sharples and Wilson, 2015). However, their safety assurance poses challenges due to the complexity of the systems. Industry and academia have shifted the safety assurance approach for ADSs to scenario-based testing and simulation-based testing in the recent years. Due to its novelty, there lacks a unified approach and methodology for the testing process across the industry. Utilising the existing knowledge and capabilities within the WMG V&V team, the key focuses for the Zenic V&V work are: 1) the introduction of the major components along a scenario-based evaluation process, 2) the implementation and integration of such a V&V framework with the rest of the Zenic Interoperable Simulation ecosystem, 3) the establishment of a simulation validation approach for both the dynamic and static elements.

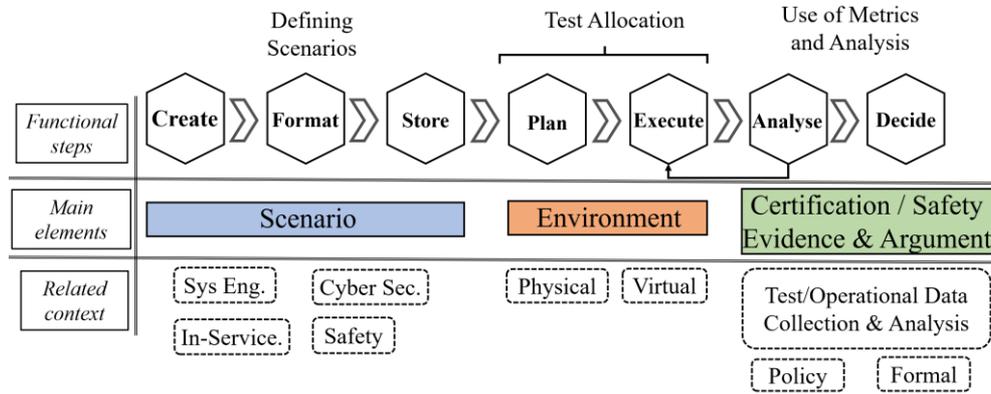
### 13.2 Scenario-based evaluation continuum overview

Figure 13-1 illustrates the key components of a scenario-based evaluation continuum; it subdivides the content into functional steps, main elements, and the related context. As the main elements, the whole process only needs **scenario**, **environment**, and **certification**. Every main element is then mapped to their corresponding functional steps at the top, and their related contexts are referred to at the bottom. The workflow is independent from the test execution environment, and is applicable for simulation execution, real-world execution as well as X-in-the-Loop (XiL) testing. The central information carried along this workflow is the scenario; relevant scenario content is created, processed, and assessed throughout the whole process, which forms the overall scenario-based V&V framework. The following three sections further detail each of these main elements to provide more insight.

### 13.3 Scenario

The **scenario** within the main elements row in Figure 13-1 is located at the upstream of the workflow; from it the structured scenario artefacts, together with pass/fail criteria, are created.

**Figure 13-1: Elements within the scenario-based evaluation continuum**



Source: Author Generated (WMG)

The scenario element includes three sub-processes: create, format and store. The create sub-process represents the creation of scenario content; scenarios can be created using two different approaches – knowledge-driven and data-driven (Menzel *et al.*, 2019; Zhang, Khastgir and Jennings, 2020; Zhang *et al.*, 2021). Furthermore, the scenario creation can be tailored towards different related contexts as shown in the bottom row, for example system engineering, safety, cyber security and in-service testing. After the scenario generation, the format sub-process will then convert the raw scenario output content into a human and machine-readable format. Subsequently, the formalised scenario content will then be stored in a scenario database for storage, sharing, analysis and query purposes.

### 13.4 Environment

The **environment** within the main elements row in Figure 13-1 contains different options of the execution environment. Test allocation is a key step within the environment element. This step entails the allocation of test scenarios to be executed in different environments. Once the allocation or the test plan has been created, the next step is to execute and analyse the scenario.

### 13.5 Certification/safety evidence & argument

The **certification/safety evidence & argument** within the main elements row in Figure 13-1 contains analyse and decide. Analyse can be further divided into 3 separate stages:

1. Correct execution - whether the intended test case has been executed.
2. Pass/fail assessment – monitoring the execution of the scenario and assessing the runtime output against a set of pre-defined pass/fail criteria/metrics.
3. Scenario parameter space exploration – based on the current and past concrete parameters (e.g., speed, acceleration) and the pass/fail criteria, a test case generator, such as an optimisation algorithm, can be applied to introduce a new set of test case parameters with the aim of violating the scenario pass criteria.

The output from the test case generator will result in the creation of new test cases which can then be fed back into the execution module. This is indicated by the arrow going from analyse to execute in Figure 13-1, and it forms a closed feedback loop within this workflow. This allows the increase of scenario coverage, the decrease of the 'unknown unsafe' region and the addition

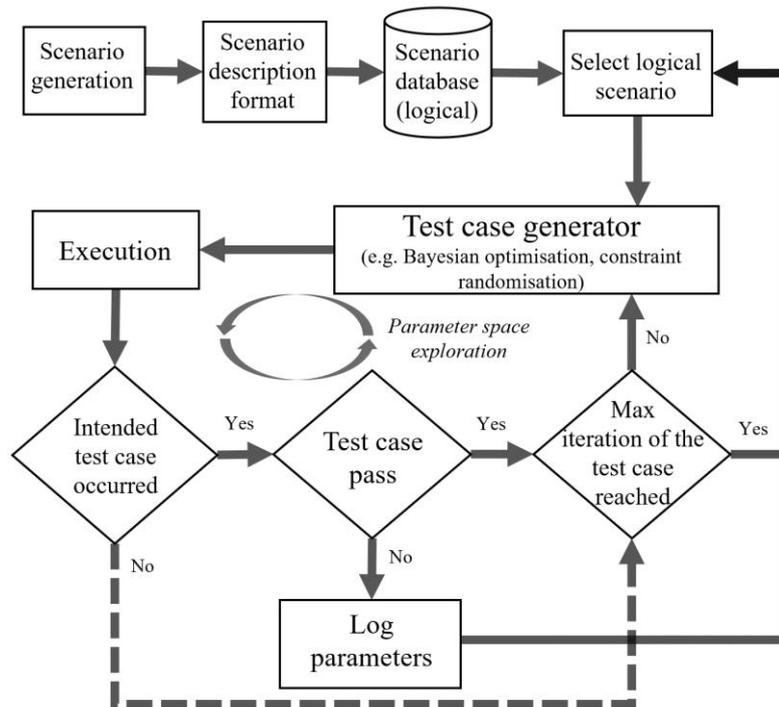
of new test cases into the database. The final stage is the decide stage, based on whether the intended test cases have occurred, the assessment on the pass/fail criteria and whether the scenario coverage was achieved. This stage will determine the output of the whole V&V process.

### 13.6 Simulation validation vs system testing

It is important to differentiate between simulation validation and system testing. As illustrated, the execution of scenarios can be achieved in a virtual environment, the real world, or a hybrid of the two. As the quantity of scenarios increases, being able to decide how they are distributed across the available environment settings for their execution becomes crucial. Simulation based testing can be used in conjunction with the more 'expensive' real world testing (Zhang *et al.*, 2021) to form a comprehensive testing strategy. In this case it acts as an initial phase of the testing to identify a small set of high interest scenarios from a large number of input scenarios (which might be impossible to execute in the real world), this small set of high interest scenarios can then be executed in a real world environment. However, to enable such a workflow, or rather to utilise any simulation environment for testing, a fundamental assumption is that simulation and real-world environments are comparable or correlated. Without this assumption, simulation-based activities carry little meaning. For the simulation validation, the same scenarios will be executed both in simulation and in the real world; subsequently their execution output will be compared to derive comparison metrics.

### 13.7 Scenario-based evaluation at the functional level

Building on the scenario-based evaluation continuum, each of the blocks can be expanded further to result into a workflow at the logical level. Figure 13-2 illustrates the functional level workflow of the scenario-based evaluation framework. Scenarios are generated and described using a human and machine-readable format at the logical scenario level, which are then stored in the Safety Pool™ scenario database (Safety Pool, 2022) ready for query or testing via API. Optionally, other desired executable formats such as ASAM OpenDRIVE (Dupuis, Hekele and Biehn, 2019) and ASAM OpenSCENARIO 1.x (ASAM e.V, 2022a) can also be generated within the 'scenario description format' box; they are then attached to the scenario database. A conversion toolchain has been developed by the WMG V&V team, to automatically convert WMG's two level abstraction Scenario Description Language (SDL) (Zhang, Khastgir and Jennings, 2020) to OpenX format. A scenario selector is implemented for performing the API calls; it iterates within a specific scenario library and retrieves individual logical scenarios and the OpenX files attached to it. The test case generator is then used to generate concrete test case parameters. Upon execution, the test case data is processed and checked against three decision modules. The first one is whether the intended test case situation occurred. If yes, then the test case pass/fail criteria are checked. If no, then the test case run is checked against a pre-defined maximum iteration number of test cases.

**Figure 13-2: Scenario-based testing workflow at the functional level**

Source: Author Generated (WMG)

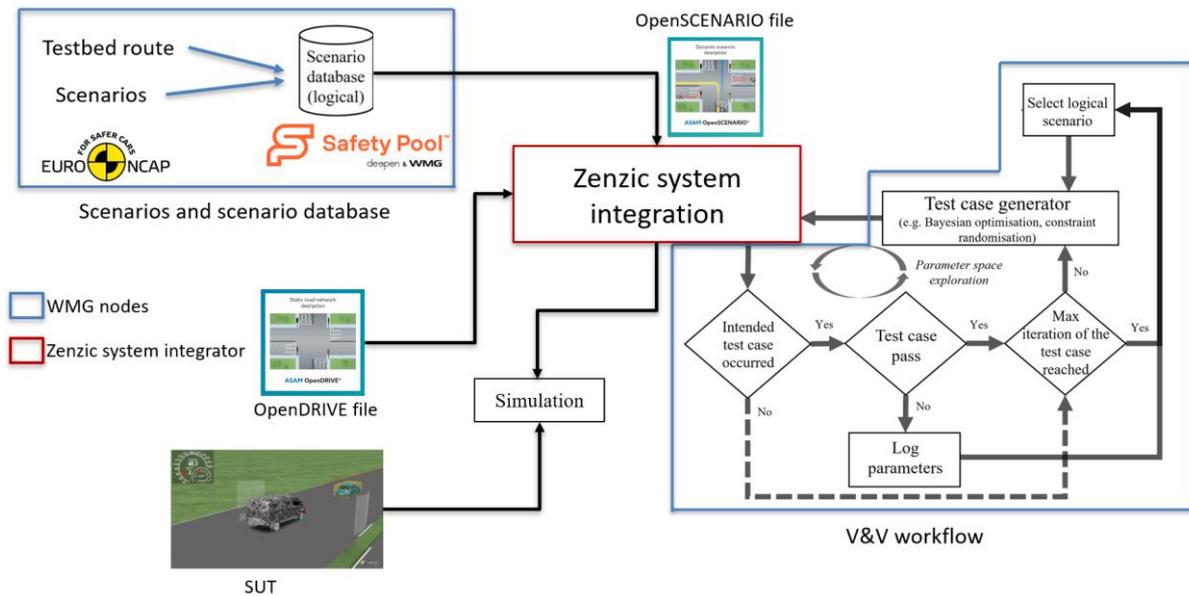
The test case pass/fail criteria module consists of multiple types of criteria sources. If a test case fails the criteria, then its parameter combination is recorded, and the current logical scenario testing is terminated. If a test case passes, then it will be checked against the maximum iteration limit. In the last step, if the maximum iteration is not reached, the current test case parameters together with the pass criteria will be fed into the test case generator where algorithms such as Bayesian optimisation (Gangopadhyay *et al.*, 2019) can be applied as a “concretiser” to introduce new parameters with the goal of driving the SUT to violate the pass criteria. The closed loop formed by the test case generator, the test execution, and the three test case checks enables the exploration of the parameter space set out within the logical scenarios, while increasing the test coverage and reducing the ‘unknown unsafe’ territories.

### 13.8 Integration with the Zenic Phase 3 ecosystem

As shown in Figure 13-3, the Zenic system integration layer sits at the centre of the Zenic V&V integration work. The integration utilises a common network to enable communication between different operational nodes that are connected to this network. At the top left, it connects to the Safety Pool™ scenario database via Safety Pool’s API. In addition, it also connects to the SUT node, and the simulation execution node. The majority part of the WMG V&V node is seen at the bottom right, which sends the scenario ID to the Zenic system integration layer and receives execution output from it. In this case, based on the original WMG V&V framework, the following modifications were made: 1) the database API call was made by the Zenic system integration layer rather than the WMG V&V module, 2) the simulation execution and the SUT were not hosted within the V&V framework, but were connected as a separate node to the Zenic system integration layer. The rest of the V&V framework largely remained the same. The following

sections illustrate the scenario preparation and retrieval process, the construction of the data transmission schema between the Zenic system integration layer and the V&V module, and the use cases performed during the Zenic V&V testing process.

**Figure 13-3: Zenic V&V integration functional diagram**



Source: Author Generated (WMG)

### 13.9 Scenario preparation and retrieval

The preparation of scenarios used in the Zenic V&V project contains several stages: 1) construction of the SDL scenarios, 2) conversion from SDL to OpenSCENARIO/OpenDRIVE, 3) custom modifications to the OpenSCENARIO file to fit the UTAC testbed OpenDRIVE specification, 4) confirmation of suitable parameters for updating with the Zenic system integration layer and the execution node, 5) upload onto Safety Pool™ database and generation of the corresponding scenario ID. Since the scenarios used in this V&V project were agreed to be Euro NCAP (Euro NCAP, 2017a) scenarios (described in section 3.5), the creation of SDL scenarios involved manual translation, based on the tables and diagrams found in the relevant Euro NCAP documentations (Euro NCAP, 2017b; NCAP, 2020), to SDL level 1 and level 2.

Upon the generation, a toolchain is available to convert from SDL level 2 description into the equivalent OpenSCENARIO 1.1 and OpenDRIVE 1.6 format. The building blocks of this conversion toolchain include an SDL parser based on the pre-defined grammar of the SDL level 2. The parsed information is then sent to a generator based on the OpenSCENARIO/OpenDRIVE specification, which outputs the corresponding files. Please note that since the scenery description within the SDL is location-independent, the generated OpenDRIVE file is also location-independent and is generic. The term 'generic' indicates that the toolchain will constantly generate the corresponding OpenDRIVE file purely based on the SDL description of the scenery; the road and lane IDs used within the file are generated by a default setting within the toolchain. For the Zenic V&V project, the virtual 'Mile Straight' of the UTAC testbed was used. To adapt the converted OpenSCENARIO file to the UTAC-based OpenDRIVE file, a manual

modification was made on each of the OpenSCENARIO files to select the suitable road and lane IDs for the entity actions. As described previously, during the V&V process, the scenario parameters need to be constantly updated, while the scenario storyline is kept the same. In the Zenic V&V process, to easily update the corresponding scenario parameters, they are written inside the 'parameter declaration' section of the OpenSCENARIO file. The last step for the scenario preparation includes the upload of the scenario to the Safety Pool™ database and gathering of the corresponding reference ID.

### 13.10 Simulation data and test case parameters

During the integration between the V&V framework and the Zenic system integration layer, the data format used for communication was Google Protocol Buffers (Protobuf in short) (Google Developers, no date). It is a language-neutral, platform-neutral, extensible mechanism for serialising structured data; it is smaller, faster, and simpler than the xml format. In order to use the Protobuf format, the process is divided into two steps: 1) the definition of the message template, including all the possible scenario parameters together with their data type, 2) sending the actual message during the V&V activities.

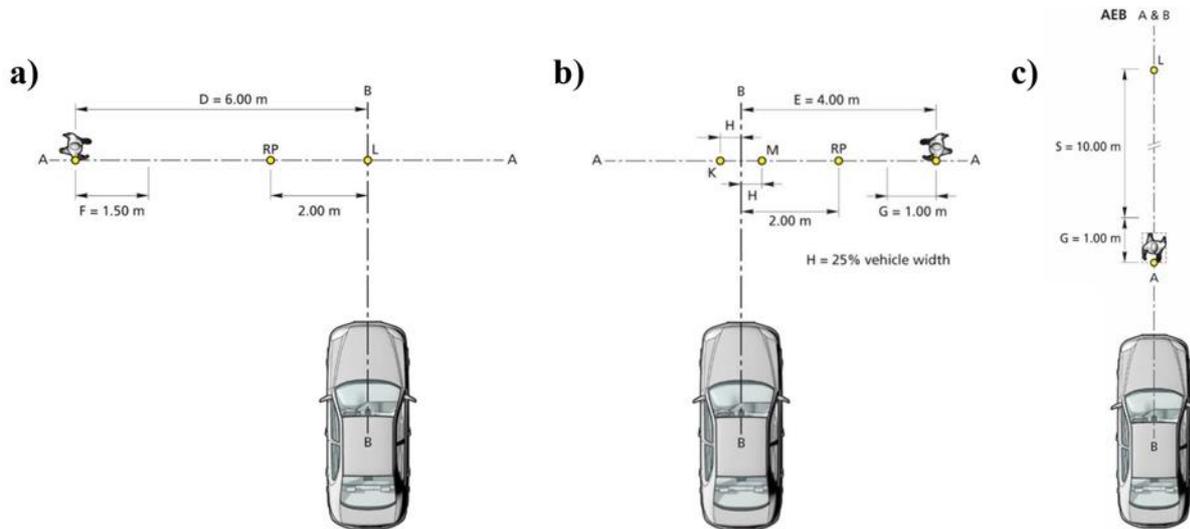
Two different message elements have been defined for the Zenic V&V integration: the message from the V&V module to the Zenic system integration layer, and the message from the system integration layer back to the V&V module. For the message sent from the V&V module to the integration layer, the required message contents are `simMode`, `scenarioID`, `testRunIteration`, and updated parameters. `simMode` indicates whether the iteration is the first, intermediate or the last iteration. The `scenarioID` is a unique identifier the database assigned to each scenario, the `testRunIteration` is used to indicate the current iteration number, and the updated parameters contain the available parameter names and the updated parameter values. For the message sent from the integration layer back to the V&V module, the `testGroundTruthRecord` of each execution is recorded and sent to the V&V module for post-run analysis. The information included within this ground truth record is `scenarioID`, `testRunIteration`, and `timeStampedGroundTruth`. The time stamped ground truth data further contains ego ground truth, environment ground truth, scripted vehicle ground truth, and VRU (vulnerable road user) ground truth data. In here the ground truth data is obtained directly from the simulation node. They contain information such as the vehicle's road lane position, lane type, junction type, left/right lane marking types, collision information, position in the world coordinates, velocity, orientation, acceleration etc.

### 13.11 Euro NCAP AEB use case

The SUT tested within the Zenic V&V project is an Autonomous Emergency Braking (AEB) system. Across different partners involved in the Zenic V&V work, the Euro NCAP scenarios were chosen. They include Car-to-Car scenarios, as well as AEB VRU (pedestrian) scenarios. Please note that although the Euro NCAP use case was chosen, the V&V framework is independent to any specific use case. To adapt the framework to different use cases, one would need to input the corresponding assessment criteria. The section below illustrates the AEB VRU scenarios.

Figure 13-4 displays the three Euro NCAP AEB VRU scenarios used for the testing. These are Car-to-Pedestrian Farside Adult (CPFA, in Figure 13-4a), Car-to-Pedestrian Nearside Adult (CPNA, in Figure 13-4b), and Car-to-Pedestrian Longitudinal Adult (CPLA, in Figure 13-4c). In CPFA, an adult pedestrian is walking towards the trajectory of the SUT from the farside. In CPNA, an adult pedestrian is moving toward the SUT from the nearside. And in the CPLA, an adult pedestrian is walking toward the SUT along the SUT’s trajectory from the front.

**Figure 13-4: Euro NCAP AEB VRU scenarios**



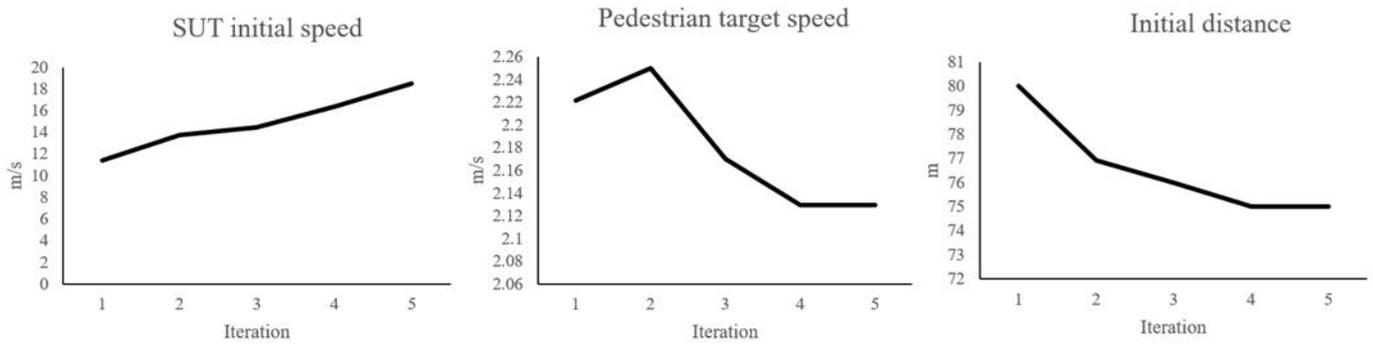
Source: (NCAP, 2020)

During both the car-to-car scenarios and the VRU scenarios, the V&V framework effectively converged towards a parameter combination which encourages hazardous situations (i.e., collision in this case). For example, during the initial iteration of the CPFA scenario, the pedestrian would reach the opposite side of the road even before the SUT reached the pedestrian’s trajectory. Since the role of the optimisation module is to identify hazardous situations, within the subsequent iterations, the SUT initial speed, the pedestrian target speed, and the initial distance between the vehicle and the pedestrian were updated by the algorithm. It can be seen in Figure 13-5, the initial SUT speed was constantly updated to higher values to encourage early intersection between SUT and pedestrian. In the meantime, the pedestrian target speed was updated to lower values, and the initial distance between the pedestrian and the SUT was also decreased, thus encouraging a potential collision to occur. Similar performance of the optimisation algorithm was observed across all the test scenarios. Figure 13-6 illustrates the same parameter updates but using test case layouts; the arrows indicate the speed and the length of them indicate the magnitude. Please note that these three parameters are not the only parameters that were updated during the testing, but they are the most visually representative.

### 13.12 Dynamic and static elements validation method

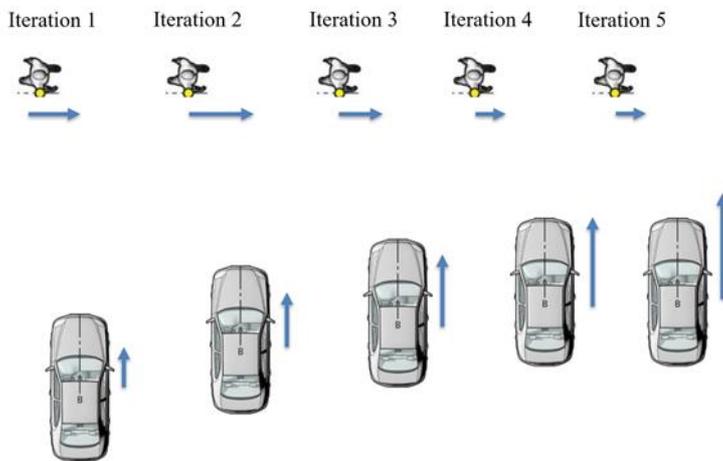
The V&V project has also developed a robust methodology to validate the static and dynamic elements of simulation. More information can be found in the full V&V report through [zenzic.io](http://zenzic.io).

**Figure 13-5: Examples of parameter optimisation**



Source: Author Generated (WMG)

**Figure 13-6: Visualisations of the test case parameter update**



Source: Author Generated (WMG)

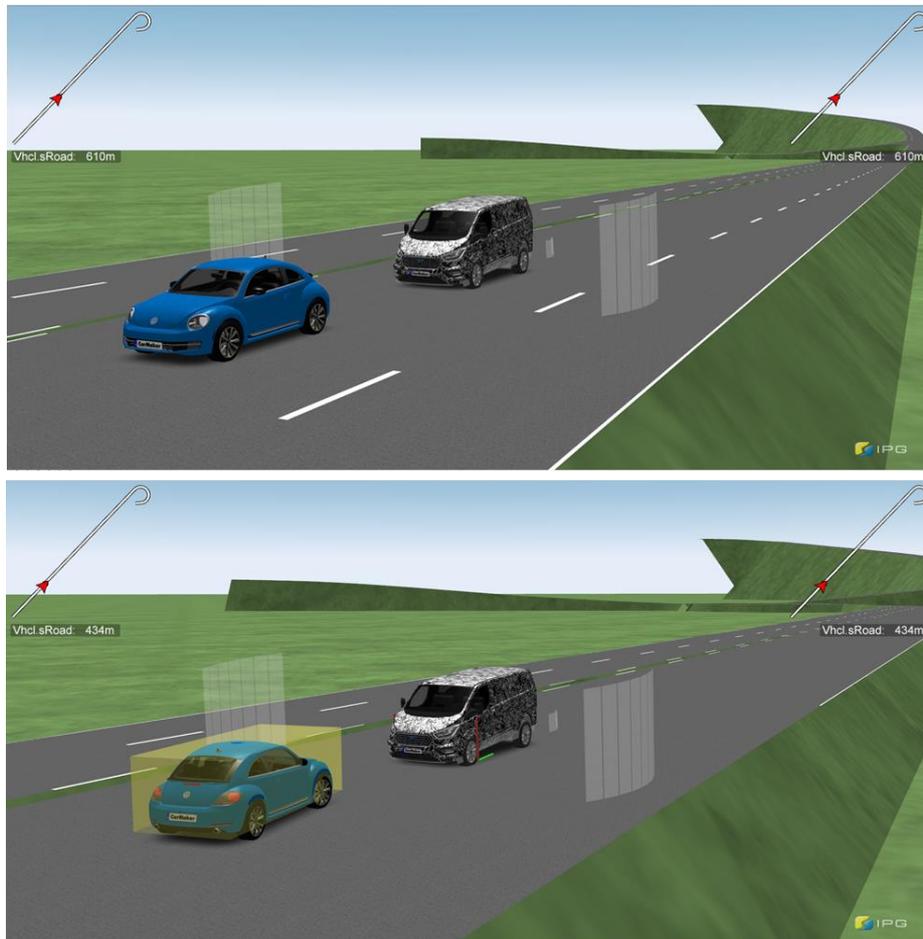
## 14 | Simulation results

The project use cases of AEB-O, AEB-P and EEBL were tested using several variations of MiL, SiL, DiL and ViL. The following sections outline the simulation results of each use case.

### 14.1 Use case 1: AEB-O

In addition to the real-time simulation capability, the advantage of the SiL model is that KANSM can be used to integrate the existing vehicle and AEB model to other components such as rFPro, which can be used to visualise the environment to a higher degree of photorealism. Further, if the sensor models, which are currently integrated into the Carmaker environment, are made universal using standards such as ASAM Open Simulation Interface (OSI), any external sensor plugin which uses such standards can be easily integrated into the model. This would enable more in-depth testing according to the levels in the V Model. With the DiL capability integrated, the driving simulator was used to drive the simulated Mile Straight model. The AEB system was observed to be activated when a traffic vehicle was in the path according to the functionality of the model. Figure 14-1 shows the visualisation in IPG carmaker environment.

**Figure 14-1 Visualisation of Ego Vehicle and Traffic Vehicle in IPG Carmaker**

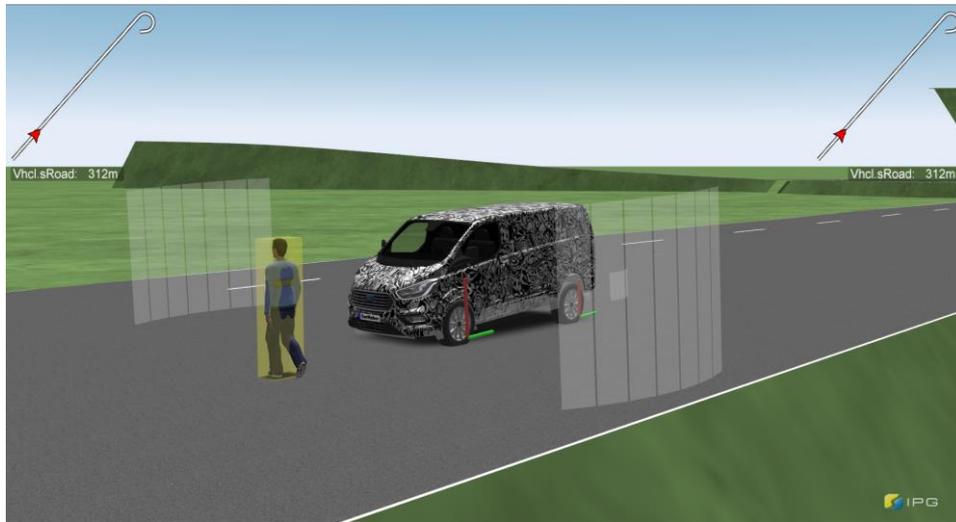


Source: Author Generated (© KAN Engineering)

## 14.2 Use case 2: AEB-P

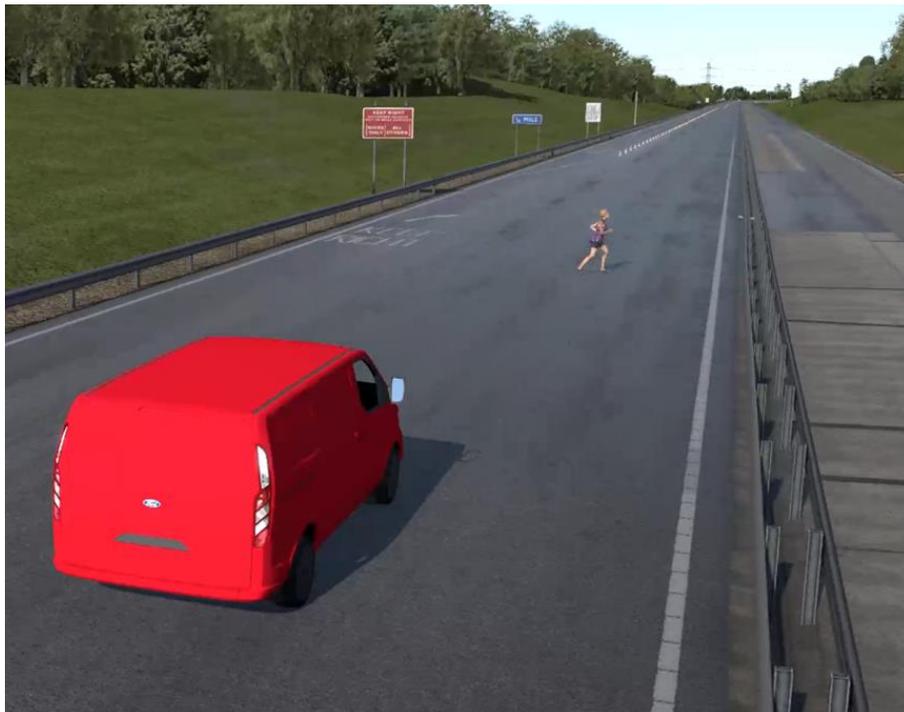
The approach followed has great potential for CAM testing due to the flexibility of the simulation system. Scripted pedestrian traffic can be used for testing the CAV systems. However, the HIKER pedestrian may also be considered the SUT, in which case the simulation system can be used for the analysis of the interaction of real humans with CAVs. Figure 14-2 shows the visualisation of AEB-P scenario in IPG Carmaker environment and Figure 14-3 shows the visualisation in rFPro environment.

**Figure 14-2 Visualisation of Ego Vehicle and Pedestrian in IPG Carmaker**



Source: Author Generated (© KAN Engineering)

**Figure 14-3 Visualisation of Ego Vehicle and Pedestrian in rFPro environment**



Source: Author Generated (© KAN Engineering)

### 14.3 Use case 3: EEBL

MiL testing for EEBL was carried out using IPG Carmaker and Simulink. An IPG scenario was created with an FV and SV in the Mile Straight, and parameters such as the distances and accelerations of the FV were varied to observe the behaviour of the EEBL algorithm. For SiL and DiL tests, an OpenSCENARIO based scenario script was used for the simulation. KAN HMI was used to visualise the ROI and EEBL warnings to the SUT. Figure 14-4 shows the EEBL SV being driven using UTAC Millbrook’s driving simulator, displaying the HMI to the driver and also shows the V2X setup at Applus IDIADA relaying the messages between the FV and SV.

**Figure 14-4: EEBL test setup at UTAC and Applus IDIADA, and the HMI warning message**



Source: Author Generated (UTAC, Applus IDIADA, KAN Engineering)

## 15 | Discussions and conclusions

The Zentric Interoperable Simulation Projects series (phases 2 & 3) exhibited collaborative efforts across CAM Testbed UK and beyond toward a common goal of building a technically robust and financially cost-effective solution for distributed real-time co-simulation capabilities. The project successfully integrated various simulation tools across the UK to demonstrate a seamless end-to-end process for development, virtual testing, Verification and Validation (V&V), and regulatory assurance for CAV applications. A unique offer has been developed by leveraging the UK's simulation strength while establishing a common, flexible integration framework. The offer is aligned with customer needs and their development processes.

In addition to what has been achieved as part of the previous phase of this project (phase 2), a wide range of additional simulation capabilities have been integrated in this phase. This formed a comprehensive, integrated, distributed real-time co-simulation environment across CAM Testbed UK and beyond. More specifically, the required interfaces and APIs for the following simulation tools have been developed, implemented, tested and integrated with the DDS Middleware via KANSM software:

- **UTAC:** IPG, rFpro, Driving Simulator, Vehicle Simulator, test track
- **SMLL:** VISSIM (and SCOOT through VISSIM), scenario generation
- **Applus IDIADA:** V2X HiL
- **HORIBA MIRA:** Test Oracle
- **WMG:** Scenario generation, closed-loop V&V pipeline
- **University of Leeds:** HIKER (PiL)

Engagement with Ford, a leading global OEM, as the project customer was another great achievement in this project. Ford provided the required technical data, information, simulation models and technical support to the team in a timely and effective manner. In addition, Ford's engineers have worked closely with the project team to implement the proposed technical solutions to make them compatible with, and useful for, Ford's product development processes. The main outcome of this joint effort can be summarised as below:

- Increasing the simulation execution time up to 30 times faster than the original Ford simulation processes (MiL vs SiL),
- Converting Ford's ADAS scenarios and test plans into OpenSCENARIO format and providing the possibility of executing the scenarios with a wide range of simulation tools (by using KAN Scenario Engine),
- Implementing an integrated simulation solution to provide a seamless development platform for the various stages of the V model, including MiL, SiL, HiL, DiL, ViL and test track (by using KANSM).
- Providing Ford Engineers with access to some unique simulation capabilities across the UK, including the University of Leeds' PiL rig (HIKER) integrated with state-of-the-art simulation facilities, simulators, and test track at UTAC, Applus IDIADA's V2X HiL, HORIBA MIRA's test oracle, SMLL's scenario generation, and WMG's V&V process.

The design, development and implementation of such a cutting-edge distributed co-simulation solution requires access to the latest technological tools and techniques, which could not happen without close collaboration with the world-leading technology provider companies. We have been privileged to get help and support from several global companies as project contributors. Especially thanks to IPG, Amazon (AWS), and RTI for providing free access to their software, products, and services and helping us develop and implement the distributed simulation architecture, DDS middleware, and cloud-based connectivity and infrastructure.

## 16 | Future work

The Zenzic Interoperable Simulation Projects series (phase 2 & 3) demonstrated the UK's world-class technical capabilities and asserted its global leadership in designing, developing, and deploying advanced simulation technologies. It strengthened its position as a global leader in research and development and its competitive advantage of testing and simulating CAV systems anywhere in the UK, as one of the UK priorities, highlighted in the UK Industrial Strategy Report (Clark and Stein, 2018).

Although the scope of this project was limited to the development and demonstration of a PoC, the outcome of this novel world-leading project confirmed the potential (technical and commercial) benefits of simulation interoperability to CAM Testbed UK and broader stakeholders.

Based on the project's outcome, some technical tasks for future work are proposed below. Note that the suggested topics are mainly about further technological improvements, enhancements and expansion of the current technical achievements.

The proposed topics for future (technical) work could be classified into three main categories:

- Advancement from a PoC to a minimum viable product
- Further enhancements and expansions
- More application coverage

### 16.1 Advancement from a PoC to a minimum viable product

The current distributed co-simulation system and its associated product (KANSM) and infrastructure (cloud-based connectivity through AWS) have been developed as a PoC, to demonstrate the state-of-the-art technological capabilities for an interoperable distributed simulation system across CAM Testbed UK and beyond. The project's outcome confirmed the technical validity and robustness of the proposed solution, but further works are required to upgrade the developed solution and its functionalities and features from PoC stage to a higher technology readiness level to make it a minimum viable product.

### 16.2 Further expansion

The recently developed distributed co-simulation setup has the capability and capacity to be further enhanced and expanded to include additional simulation capabilities and tools, simulation service providers and customers from globally dispersed geographical locations. The areas that could be considered as future work include:

- Develop and implement more advanced connectivity solutions such as MPLS, as explained in this report, to improve the robustness, quality, security, latency, and bandwidth of the distributed simulation setup.
- Further improvements on the distributed simulation capacities and/or quality by adding new features or improving the already developed ones. Some examples of potential enhancements or additional features to be included in the current system are:

- further integration of WMG's V&V system for real-time closed-loop simulation,
  - including HORIBA MIRA's scenario generation to the simulation architecture, and also enabling real-time closed-loop simulation,
  - further integration of SMLL's scenario generation for real-time closed-loop simulation,
  - Improving the quality of the created avatar for pedestrians or other objects at the University of Leeds HIKER, by increasing the numbers and positions of body trackers to be sent to the DDS data bus.
- The capability of the developed distributed simulation system could be further expanded by integrating more simulation tools (including open source, COTS and/or bespoke simulation tools, simulators and HiL rigs) from the existing partners, and other institutions and universities in the UK or globally, into the current architecture.
  - The usability of the current distributed simulation setup could be further expanded by inviting more customers or simulation providers from various geographical locations (within the UK or globally) to join the distributed simulation ecosystem and share their capabilities or utilise the simulation capabilities of the other partners.

### 16.3 More applications

The objective of the current project was to develop a PoC, to push the technological boundaries of distributed co-simulation in the CAV domain and demonstrate some possible use cases and example applications. The validity and effectiveness of the developed solution on reducing the time and cost to market for CAV products and systems should be further confirmed. The developed distributed simulation setup should be further utilised and examined in real-world development applications. Some possible applications are using the distributed simulation tool for a complete V development cycle (MiL, SiL, HiL, DiL, ViL and test track) for an automated vehicle or system (AEB, for example). The other route would be to examine more SUTs with various TRL levels and/or various autonomy levels (Level 3,4,5, as defined in Appendix A1)).

Using simulation for certification and assurance of CAV is a challenging yet important domain that needs serious consideration and careful investigation. The robustness and reliability of the proposed distributed simulation solution as a certification tool and the accuracy and validity of the simulation results need to be thoroughly studied and confirmed.

An appetite for demonstrating and enabling more "portable" customer journeys between the testbeds was expressed by project partners and stakeholders during the project. This portability focuses on enabling customers to seamlessly move between testbeds to perform the same or similar tests, while taking advantage of the different test environments each testbed offers. Portability describes the customer experience, removing friction and improving efficiency as customers move testing from one testbed to another.

Future work could explore how this PoC solution could further accommodate this outcome (demonstrated in phase 2), with comparative evaluations of alternative technical solutions. During the project, it was agreed that the adoption of interoperable models and implementation of common interfaces and data formats (inputs and outputs) would be core to further realising portability.

# 17 | Appendix A – Taxonomy definition for UK simulation capabilities

**Table 17-1 Taxonomy definition for UK simulation capabilities**

No.	Term / Concept / Standard	Definition / Description	Source
<b>Modelling and Simulation</b>			
1	<b>Model</b>	An approximation, representation, or idealization of selected aspects of the structure, behaviour, operation, or other characteristics of a real-world process, concept, or system.	(IEEE, 1989)
		Or	
		Application of a standard, rigorous, structured methodology to create and validate a physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process.	(Department of Defense, 2011)

2	<b>Simulation</b>	A model that behaves or operates like a given system when provided a set of controlled inputs.	(IEEE, 1989)
		Or	
		A method for implementing a model over time.	(Department of Defense, 2011)
		Or	
		A simulation is the imitation of the operation of a real-world process or system over time.	(Banks <i>et al.</i> , 2010)
Or			

		Computer generated environments used to test components, systems or human behaviours.	(BSI, 2020b)
3	<b>Modelling and Simulation</b>	The discipline that comprises the development and/or use of models and simulations.	(Department of Defense, 2011)
		Or	
		The use of models, including emulators, prototypes, simulators, and stimulators, either statically or over time, to develop data as a basis for making managerial or technical decisions. The terms 'modelling' and 'simulation' are often used interchangeably.	(Department of Defense, 2011)
4	<b>Discrete Model</b>	A mathematical or computational model whose output variables take on only discrete values; that is, in changing from one value to another, they do not take on the intermediate values.	(IEEE, 1989)

5	<b>Model Fidelity</b>	The degree of similarity between a model and the system properties being modelled.	(IEEE, 1989)
		Or	
		The degree to which the representation within a simulation is similar to a real-world object, feature, or condition in a measurable or perceived manner.	(Department of Defense, 2011)
		Or	
		The accuracy of the representation when compared to the real world.	(Department of Defense, 2011)
6	<b>Cyber Physical Systems (ACPS)</b>	A cyber-physical system (CPS) is an integration of computation with physical processes whose behaviour is defined by both cyber and physical parts of the system	(Lee and Seshia, 2015)

7	<b>System Under Test (SUT)</b>	A System Under Test (SUT) is a complete system that comprises hardware, software, and connectivity components; that is the object or target of a performance measurement test or benchmark.	(Raab, 2019)
8	<b>Environment Model</b>	Formal description of all environmental aspects that allow for calculating the impact of the SUT's behaviour (output) on the environment and feeding back results to the SUT (minus the model). If the SUT is component of a larger system, relevant aspects of the remaining parts of that system are also modelled.	(Nickovic and Herzner, 2019)
9	<b>Vehicle Dynamics Model</b>	The representation of the dynamic behaviour of real vehicles as realistically as possible.	(Schramm, Bardini and Hiller, 2017)
10	<b>Traffic flow Model</b>	Modelling the dynamics of traffic flows to simulate their temporal propagation through traffic networks.	(Barceló, 2010)

11	<b>Road Model</b>	Road Model describes the geometry of roads, lanes and objects, such as roadmarks on the road, as well as features along the roads, like signals.	(ASAM e.V, 2022a)
12	<b>Sensor Model</b>	A simulation unit that represents sensor behaviour.	(Nickovic and Herzner, 2019)
13	<b>Offline Simulation</b>	Offline simulation refers to running the controller in a separate offline PC to see the manipulated variable (MV) to control variable (CV) dynamic responses of the process. One major task of offline simulation is to set the different tuning parameters of the controller.	(Lahiri, 2017)
14	<b>Real time Simulation</b>	In modeling and simulation, simulated time with the property that a given period of actual time represents the same period of time in the system being modelled.	(IEEE, 1989)
		Or	

		In modelling and simulation, simulated time advances at the same rate as actual time; for example, running the simulation for one second results in the model advancing time by one second. Contrast with: fast time; slow time.	(Department of Defense, 2011)
15	<b>Soft Real time System</b>	A real-time task is said to be soft if producing the results after its deadline has still some utility for the system, although causing a performance degradation. A real-time operating system that is able to handle soft real-time tasks is called a soft real time system.	(Buttazzo, 2011)
16	<b>Hard Real time System</b>	A real-time task is said to be hard if producing the results after its deadline may cause catastrophic consequences on the system under control. A real time operating system that is able to handle hard real time tasks is called a hard real time system.	(Buttazzo, 2011)
18	<b>Model Based Development</b>	Model Based Design is a model-centric approach to the development of control, signal processing, communications, and other dynamic systems. Rather than relying on physical prototypes and textual specifications, Model-Based Design uses a model throughout development.	(Aarenstrup, 2015)

19	<b>Model-in-the-loop (MiL)</b>	Execution of the SUT model against the environment model, purely virtual, not real-time.	(Nickovic and Herzner, 2019)
20	<b>Software-in-the-loop (SiL)</b>	Simulation and simulators that employ one or more pieces of operational equipment (to include computer hardware) within the simulation/simulator system.	(Department of Defense, 2008)
		Or	
		Execution of the target SUT against the environment model, at least of SUT's software. Not necessarily real-time.	(Nickovic and Herzner, 2019)
21	<b>Hardware-in-the-loop (HiL)</b>	Simulation and simulators that employ one or more elements of operational software (computer programming code) within the simulation/simulator system.	(Department of Defense, 2008)
		Or	

		Execution of the target SUT in a hardware-based environment that allows real-time.	(Nickovic and Herzner, 2019)
22	<b>Object-in-the-loop (XiL)</b>	The stages of development and testing process includes most prominently model-in-the-loop (MiL), software-in-the-loop (SiL) and hardware-in-the-loop (HiL). The notation 'XiL' indicates that the standard can be used for all 'in-the-loop' systems.	(ASAM e.V, 2022b)
23	<b>Vehicle-in-the-loop (ViL)</b>	ViL represents the state when the vehicle is moving on the real surface or on a test bench, and the full environment is simulated, the actuators of the vehicle get their input signal directly from the traffic simulation software, i.e. sensor spoofing is applied.	(Horváth <i>et al.</i> , 2019)
24	<b>Driving Simulator</b>	Driving simulators place the driver in an artificial environment believed to be a valid substitute for one or more aspects of the actual driving experience.	(Chang, 2015)
<b>Simulation Interoperability</b>			

1	<b>Co- Simulation</b>	Co-simulation (cosim) is a technique that enables execution of an integrated system by simulating independent, distributed sub-systems which can be either models or hardware, individually with coupled boundary connections.	(Raczkowski <i>et al.</i> , 2020)
		Or	
		Co-simulation consists of the theory and techniques to enable global simulation of a coupled system via the composition of simulators. Each simulator is broadly defined as a black box capable of exhibiting behavior, consuming inputs, and producing outputs.	(Thule <i>et al.</i> , 2018)
2	<b>Distributed Simulation</b>	Distributed simulation refers to technologies that enable a simulation program to execute on a computing system containing multiple processors such as personal computers, interconnected by a communication network.	(Fujimoto, 2000)
		Or	
		A distributed simulation is one which uses more than one computer simultaneously, to guarantee access from/to different resources (e.g., multi-users operating different systems, or distributed data sets).	(Petty, 1995)

		Or	
		Distributed simulation is often seen as a sub-domain of distributed systems. Distributed computing systems deal with software that is executed in parallel on multiple autonomous computers.	(Tolk, 2012)
3	<b>Internet of Things (IoT)</b>	A network of items—each embedded with sensors—which are connected to the Internet.	(Minerva, Biru and Rotondi, 2015)
4	<b>Industrial Internet of Things (IIoT)</b>	A system comprising networked smart objects, cyber-physical assets, associated generic information technologies and optional cloud or edge computing platforms, which enable real-time, intelligent, and autonomous access, collection, analysis, communications, and exchange of process, product and/or service information, within the industrial environment, so as to optimise overall production value.	(Boyes <i>et al.</i> , 2018)

5	<b>Quality of Service (QoS)</b>	Quality of Service (QoS) is a set of technologies that work on a network to guarantee its ability to dependably run high-priority applications and traffic under limited network capacity.	(Paloalto Networks, 2021)
6	<b>System of Systems (SoS)</b>	A 'System of Systems' (SoS) is a system of interest (SOI) whose elements are managerially and/or operationally independent systems. These interoperating and/or integrated collections of constituent systems usually produce results unachievable by the individual systems alone.	(International Council on Systems Engineering (INCOSE), 2015)
7	<b>System architecture</b>	Fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution.	(IEEE Standard, 2011)
8	<b>Middleware</b>	Middleware is software that enables one or more kinds of communication or connectivity between two or more applications or application components in a distributed network. By making it easier to connect applications that weren't designed to connect with one another - and providing functionality to connect them in intelligent ways - middleware streamlines application development and speeds time to market.	(IBM, 2021)

9	<b>Data Distribution Service</b>	The OMG Data Distribution Service (DDS™) is a middleware protocol and API standard for data-centric connectivity from the Object Management Group® (OMG®). It integrates the components of a system together, providing low-latency data connectivity, extreme reliability, and a scalable architecture that business and mission-critical Internet of Things (IoT) applications need.	(DDS Foundation, 2021b)
10	<b>High Level Architecture (HLA)</b>	The HLA is a technical architecture developed to facilitate the reuse and interoperation of simulation systems and assets (e.g., federation managers, data collectors, passive viewers, real world ('live') systems, and other utilities).	(IEEE Standard, 2010)
11	<b>Distributed Interactive Simulation (DIS)</b>	A time and space coherent synthetic representation of world environments designed for linking the interactive, free-play activities of people in operational exercises. The synthetic environment is created through real-time exchange of data units between distributed, computationally autonomous simulation applications in the form of simulations, simulators, and instrumented equipment interconnected through standard computer communicative services. The computational simulation entities may be present in one location or may be distributed geographically.	(IEEE Standard, 2012)

12	<b>MQTT</b>	MQTT is an OASIS standard messaging protocol for the Internet of Things (IoT). It is designed as an extremely lightweight publish/subscribe messaging transport that is ideal for connecting remote devices with a small code footprint and minimal network bandwidth.	(MQTT, 2022)
13	<b>Interoperability</b>	The ability of a model or simulation to provide services to, and accept services from, other models and simulations, and to use the services so exchanged to enable them to operate effectively together.	(Department of Defense, 2008)
		Or	
		Degree to which two or more systems, products or components can exchange information and use the information that has been exchanged.	(ISO/IEC, 2011)
14	<b>Portability</b>	A software unit is portable (exhibits portability) across a class of environments to the degree that the cost to transport and adapt it to a new environment in the class is less than the cost of redevelopment.	(Mooney, 1997)

15	<b>Modularity</b>	Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components.	(ISO/IEC, 2011)
16	<b>Composability</b>	Composability is the ability to select and assemble, combine and recombine, configure and reconfigure simulations reusable components.	(Sivakumar <i>et al.</i> , 2015)
17	<b>Model interoperability</b>	Model interoperability entitles the capability of effortless exchange of simulation assets and models between the partners.	(Sivakumar <i>et al.</i> , 2015)
18	<b>Simulation interoperability</b>	The capability of effortless carrying out of similar tests in different environments or drawing down capabilities for different tests.	(Soltani and Reed, 2021)

19	<b>6 Levels of Conceptual Interoperability Model</b>	<p>Level 0 - No Interoperability</p> <p>Level 1 - Technical Interoperability</p> <p>Level 2 - Syntactic Interoperability</p> <p>Level 3 - Semantic Interoperability</p> <p>Level 4 - Pragmatic Interoperability</p> <p>Level 5 - Dynamic Interoperability</p> <p>Level 6 - Conceptual Interoperability</p>	(Wang, Tolk and Wang, 2009)
<b>OpenX Standards</b>			
1	<b>ASAM OpenDRIVE</b>	<p>The ASAM OpenDRIVE format provides a common base for describing road networks with extensible markup language (XML) syntax, using the file extension xodr. The data that is stored in an ASAM OpenDRIVE file describes the geometry of roads, lanes, and objects, such as roadmarks on the road, as well as features along the roads, like signals. The road networks that are described in the ASAM OpenDRIVE file can either be synthetic or based on real data.</p>	(ASAM e.V, 2021)
2	<b>ASAM OpenSCENARIO</b>	<p>ASAM OpenSCENARIO defines the dynamic content of the world, for example, behavior of traffic participants and how these are expected to interact with each other and the environment. Static components, such as the road network, are not part of ASAM OpenSCENARIO but can be referenced by the format.</p>	(ASAM e.V, 2021)

3	<b>ASAM OpenLABEL</b>	ASAM OpenLABEL is currently under development. The project aims to standardize the annotation format and labeling methods for objects and scenarios. ASAM OpenLABEL will provide a guideline on how the labeling methods and definitions should be used.	(ASAM e.V, 2021)
4	<b>ASAM OpenODD</b>	ASAM OpenODD (Operational Design Domain) is still a very young standardization initiative within the ASAM Simulation domain ASAM has started a concept project in Sep 2020, to create the base concept for a future ASAM OpenODD standard. The aim is to provide a format that is capable of representing a defined Operational Design Domain for connected automated vehicles (CAV).	(ASAM e.V, 2021)
5	<b>ASAM OpenXOntology</b>	The OpenX standards describe road networks, driving maneuvers and test scenarios for driving and traffic simulation. The standards share concepts from the area of road traffic, such as roads, lanes, and traffic participants. ASAM OpenXOntology provides an ontology based architecture for these concepts and thus a common definition of the domain model for the ASAM OpenX standards.	(ASAM e.V, 2021)
6	<b>ASAM OSI</b>	The ASAM OSI® (Open Simulation Interface) is a specification for interfaces between models and components of a distributed simulation. OSI is strongly focused on the environmental perception of automated driving functions.	(ASAM e.V, 2021)

7	<b>ASAM XiL</b>	ASAM XIL is an API standard for the communication between test automation tools and test benches. The standard supports test benches at all stages of the development and testing process – most prominently model-in-the-loop (MiL), software-in-the-loop (SiL) and hardware-in-the-loop (HiL). The notation 'XiL' indicates that the standard can be used for all 'in-the-loop' systems.	(ASAM e.V, 2021)
8	<b>ASAM XiL -MA</b>	ASAM XIL-MA is an extract from ASAM XIL and contains the model access port specification as well as some common functionality from this standard.	(ASAM e.V, 2021)
<b>CAV Vocabulary</b>			
1	<b>SAE Levels of driving automation</b>	Level or Category 0 - No Driving Automation -The performance by the driver of the entire Dynamic Driving Task (DDT), even when enhanced by active safety systems.	(SAE, 2021)

		<p>Level or Category 1 - Driver Assistance - The sustained and ODD-specific execution by a driving automation system of either the lateral or the longitudinal vehicle motion control subtask of the DDT (but not both simultaneously) with the expectation that the driver performs the remainder of the DDT.</p>	<p>(SAE, 2021)</p>
		<p>Level or Category 2 - Partial Driving Automation - The sustained and ODD-specific execution by a driving automation system of both the lateral and longitudinal vehicle motion control subtasks of the DDT with the expectation that the driver completes the Object and Event Detection and Response (OEDR) subtask and supervises the driving automation system.</p>	<p>(SAE, 2021)</p>
		<p>Level or Category 3 - Conditional Driving Automation - The sustained and ODD-specific performance by an ADS of the entire DDT with the expectation that the DDT fallback-ready user is receptive to ADS-issued requests to intervene, as well as to DDT performance-relevant system failures in other vehicle systems, and will respond appropriately.</p>	<p>(SAE, 2021)</p>
		<p>Level or Category 4 - High Driving Automation - The sustained and ODD-specific performance by an ADS of the entire DDT with the expectation that the DDT fallback-ready user is receptive to ADS-issued requests to intervene, as well as to DDT performance-relevant system failures in other vehicle systems and will respond appropriately.</p>	<p>(SAE, 2021)</p>

		Level or Category 5 - Full Driving Automation - The sustained and unconditional (i.e., not ODD-specific) performance by an ADS of the entire DDT and DDT fallback without any expectation that a user will need to intervene.	(SAE, 2021)
2	<b>Advanced driver assistance system (ADAS)</b>	entity consisting of interdependent components that supports human drivers by performing a part of the dynamic driving task or providing safety relevant information.	(BSI, 2020b)
3	<b>Automated driving system (ADS)</b>	hardware and software that are collectively capable of performing the dynamic driving task on a sustained basis, regardless of whether it is limited to a specific operational design domain.	(BSI, 2020b)
4	<b>Automated vehicle (AV)</b>	vehicle designed or adapted to be capable, in at least some circumstances or situations, of safely driving itself that may lawfully be used when driving itself, in at least some circumstances or situations, on roads or other public places in Great Britain.	(BSI, 2020b)

5	<b>Connected vehicle (CV)</b>	Vehicle equipped with wireless communications technology that enables data transfer with other vehicles, infrastructure, or other networks.	(BSI, 2020b)
6	<b>Vehicle-to-everything (V2X)</b>	unidirectional or bidirectional sharing of data between vehicles and other vehicles, infrastructure, other road users or any other communications system.	(BSI, 2020b)
7	<b>Vehicle-to-vehicle (V2V)</b>	vehicles sharing data with other vehicles.	(BSI, 2020b)
8	<b>Connected and Automated Mobility (CAM)</b>	The Connected and Automated Mobility (CAM) sector is an entire ecosystem of services, operations and infrastructures comprised of a variety of actors and stakeholders.	(ENISA, 2020)
		Or	

		Connected and Automated Mobility (CAM) refers to autonomous/connected vehicles or self-driving cars (vehicles that can guide themselves without human intervention).	(European Commission, 2022)
9	<b>Dynamic driving task (DDT)</b>	All of the Real-time operational and tactical functions required to operate a vehicle safely in on-road traffic.	(BSI, 2020b) (SAE, 2021) (Khastgir, 2020)
10	<b>Operational design domain (ODD)</b>	Operating conditions under which a given driving automation system or feature thereof is specifically designed to function.	(BSI, 2020b) (SAE, 2021) (Khastgir, 2020)
		Or	
		Operating conditions under which a given driving automation system or feature thereof is specifically designed to function, including, but not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain traffic or roadway characteristics.	(SAE, 2021)

11	<b>Scenario based testing</b>	The scenario-based test methodology enables test cases to be defined independently of the function and to be used across different test sites.	(Bertraundt, no date)
		Or	
		A methodology to purposefully vary and validate the operating scenarios of the automated vehicle. Therefore, the systematic derivation of scenarios and further assumptions have to be documented along the development process to ensure a traceable scenario generation.	(Menzel, Bagschik and Maurer, 2018)
12	<b>Functional scenarios</b>	Functional scenarios include operating scenarios on a semantic level. The entities of the domain and the relations of those entities are described via a linguistic scenario notation. The scenarios are consistent. The vocabulary used for the description of functional scenarios is specific for the use case and the domain and can feature different levels of detail.	(Zhang, Khastgir and Jennings, 2020)  (Menzel, Bagschik and Maurer, 2018)
12	<b>Logical scenarios</b>	Logical scenarios include operating scenarios on a state space level. Logical scenarios represent the entities and the relations of those entities with the help of parameter ranges in the state space. The parameter ranges can optionally be specified with probability distributions. Additionally, the relations of the parameter ranges can optionally be specified with the help of correlations or	(Zhang, Khastgir and Jennings, 2020)  (Menzel, Bagschik and Maurer, 2018)

		numeric conditions. A logical scenario includes a formal notation of the scenario.	
12	<b>Concrete scenarios</b>	Concrete scenarios distinctly depict operating scenarios on a state space level. Concrete scenarios represent entities and the relations of those entities with the help of concrete values for each parameter in the state space.	(Zhang, Khastgir and Jennings, 2020)  (Menzel, Bagschik and Maurer, 2018)

Source: Author Generated

## 18 | Appendix B – Existing scenario engines using OpenSCENARIO standard

**Table 18-1 Existing scenario engines using OpenSCENARIO standard**

No.	Product/ service name/ toolbox	Company name	Functionalities/ features	Source
1	AAI Scenario Cloning & Extraction	Automotive Artificial Intelligence (AAI) GmbH	<ul style="list-style-type: none"> <li>• Sensor data is processed in order to generate OpenSCENARIO files and maps in OpenDRIVE format.</li> <li>• Criticality evaluation performed automatically in order to discover high-quality scenarios from which to build scenario-based test cases</li> <li>• The categorisation of the scenarios that have been collected into distinguishable categories</li> </ul>	(Automotive Artificial Intelligence (AAI) GmbH, 2021)
2	Astah	Change Vision Inc.	<ul style="list-style-type: none"> <li>• The usage of feature models to generate probable variants from user-defined ontologies (e.g. ISO 34502, JAMA/SAKURA Project Scenario structure) is known as modelling (UML, SysML, GSN, STPA, Ontology). With these capabilities, systematic V&amp;V may be achieved for autonomous driving vehicles by, for example, thoroughly developing test-scenarios (ASAM OpenSCENARIO, OpenDRIVE) for them.</li> </ul>	(Change Vision, 2021)
3	Automotive Simulation Models (ASM)	dSPACE GmbH	<ul style="list-style-type: none"> <li>• The standardised interfaces of the ASMs make it easier to expand models and even create complete virtual vehicles due to their standardised interfaces.</li> </ul>	

			Graphic parameterisation tools make it simple and easy to create road networks and traffic manoeuvres, which can then be shown in a preview before being used in the real world.	(dSPACE GmbH, 2022a)
4	Brix Scenario	iASYS Technology Solutions Pvt. Ltd.	<ul style="list-style-type: none"> <li>• Distributed data processing for large volumes of engineering data using modern technologies, while also creating and deploying scenarios using graph technology at the same time.</li> </ul>	(iASYS Technology Solutions, 2022)
5	Scenario RRR	IPG Automotive GmbH	<ul style="list-style-type: none"> <li>• The toolbox ScenarioRRR is designed to give optimum support for the automated generation of scenarios, which allows the production of scenarios from a variety of different sources of input data, such as raw sensor measurements or object lists generated by perception algorithms. It also allows for the importation of collision database information (in PCM format) into the simulation environment through the use of a special toolbox.</li> </ul>	(IPG Automotive GmbH, 2022)
6	Cognata simulation platform	Cognata Ltd.	<ul style="list-style-type: none"> <li>• Facilitates the creation of parameterisable scenarios, which can range from simple manoeuvres to complex multi-agent urban interaction situations.</li> <li>• Scale scenarios easily and effectively using the Studio UI or robust scripting features, which include support for ASAM OpenSCENARIO®.</li> <li>• The Cognata platform provides a library of customised ready-to-run scenarios. The scenario</li> </ul>	(Cognata, 2021)

			library is easily modifiable and customisable, and it may be supplemented by proprietary created scenarios.	
7	Foretify (tm)	Foretellix Ltd.	<ul style="list-style-type: none"> <li>• To allow a 'measurable safety' ecosystem, use OpenSCENARIO 2.0 to represent both scenarios and coverage objectives at a high level.</li> <li>• Automated generation of scenario variant combinations, as well as monitoring and tracking scenario coverage.</li> <li>• The ability to collect coverage data from execution across all testing platforms (test cars, test tracks, simulation, and X-in-the-Loop)</li> </ul>	(Foretellix, 2022)
8	ModelDesk	dSPACE GmbH	<ul style="list-style-type: none"> <li>• ModelDesk keeps track of parameters such as the route, driving manoeuvres, traffic, and vehicle configuration, as well as the simulation results and, if necessary, measurement data, to ensure readily reproducible simulations.</li> </ul>	(dSPACE GmbH, 2022a)
9	Scenario Generation Service	dSPACE GmbH	<ul style="list-style-type: none"> <li>• Exact and consistent replication of real-world scenarios</li> <li>• For scenario-based testing, generation of logical and replay scenarios</li> </ul>	(dSPACE GmbH, 2022b)
10	Simian	Applied Intuition Inc.	<ul style="list-style-type: none"> <li>• Simple graphical user interface-based approach for scenario creation</li> <li>• Parameter values and scenario modifications generated programmatically</li> </ul>	(Applied Intuition, 2022)

11	The Five Platform	Five	<ul style="list-style-type: none"> <li>• Create or choose scenarios</li> </ul>	(Five AI, 2021)
12	VTD Scale	VIRES Simulationstechnologie GmbH	<ul style="list-style-type: none"> <li>• Parametric control of both static and dynamic content (such as road decoration and placement of pedestrians, obstacles, etc.)</li> </ul>	(VIRES, 2021)
13	ScenarioRunner	CARLA Simulator	<ul style="list-style-type: none"> <li>• A Python interface or the OpenSCENARIO standard can be used to define the scenarios.</li> <li>• Used to prepare AD agents for evaluation by simulating complicated traffic scenarios and routes for the agents to navigate.</li> </ul>	(CARLA Simulator, 2022)
14	esmini player	OpenScenario Environment Simulator Minimalistic (esmini)	<ul style="list-style-type: none"> <li>• Consists of two main libraries; RoadManager – Provides interface to road networks described in the OpenDRIVE format and Scenario Engine – Includes RoadManager library and provides a viewer and API interface to traffic scenarios described in the OpenSCENARIO format.</li> <li>• Supports both OpenSCENARIO v1.1 and v1.0. ASAM transformation scheme (migration0_9_1to1_0.xslt,</li> </ul>	(esmini, 2022)

			part of the OpenSCENARIO release bundle) can be used with the tools for automatic migration of XML files in order to run older versions.		
15	Driving Designer	Scenario	Mathworks	<ul style="list-style-type: none"> <li>• Drag and drop interface to create road and actor models</li> <li>• Load driving scenarios in European New Car Assessment Programme (Euro NCAP) and import roads and lanes data from ASAM OpenDRIVE® (file versions 1.4, 1.5 and 1.6), road data from OpenStreetMap®, HERE HD Live Map [1] , or Zenrin Japan Map API 3.0 into a driving scenario.</li> <li>• Export road networks, actors, and trajectories in a driving scenario to the to the ASAM OpenSCENARIO® 1.0 file format and road networks in a driving scenario to the ASAM OpenDRIVE file format.</li> <li>• MATLAB code generation of the scenario and sensors, and option program modifications and import option for further simulation</li> <li>• Create a Simulink® model using the scenario and sensors, and then test the sensor fusion or vehicle control algorithms using the created models.</li> </ul>	(MathWorks, 2021)

Source: Author Generated

## 19 | References

Aarenstrup, R. (2015) *Managing Model-Based Design, The Mathworks, Inc.* Mathworks. doi:10.1001/archderm.1982.01650190012009.

Applied Intuition, Inc. (2022) *Simian | ADAS and AV simulation for autonomous driving | Applied Intuition*. Available at: <https://www.appliedintuition.com/product/simian> (Accessed: February 28, 2022).

ASAM (2021) *ASAM OpenSCENARIO®*. Available at: <https://www.asam.net/standards/detail/openscenario/> (Accessed: February 28, 2022).

ASAM e.V (2021) *ASAM SIM: Guide Standardization for Highly Automated Driving*. Available at: [www.asam.net](http://www.asam.net).

ASAM e.V (2022a) *ASAM OpenDRIVE®, ASAM*. Available at: <https://www.asam.net/standards/detail/opendrive/> (Accessed: February 25, 2022).

ASAM e.V (2022b) *ASAM XIL, ASAM*. Available at: <https://www.asam.net/standards/detail/xil/> (Accessed: February 25, 2022).

Automotive Artificial Intelligence (AAI) GmbH (2021) *AAI Scenario Cloning & Extraction*. Available at: <https://www.automotive-ai.com/technologies/scenario-cloning> (Accessed: February 28, 2022).

AWS (2022) *What is AWS Site-to-Site VPN? - AWS Site-to-Site VPN, Amazon Web Services*. Available at: [https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC\\_VPN.html](https://docs.aws.amazon.com/vpn/latest/s2svpn/VPC_VPN.html) (Accessed: February 28, 2022).

Balfe, N., Sharples, S. and Wilson, J.R. (2015) "Impact of automation: Measurement of performance, workload and behaviour in a complex control environment," *Applied Ergonomics*, 47, pp. 52–64. doi:10.1016/J.APERGO.2014.08.002.

Banks, J. *et al.* (2010) *Discrete-Event System Simulation*. 5th edn. Prentice Hall.

Barceló, J. (2010) *Fundamentals of Traffic Simulation*. Edited by J. Barceló. Springer. doi:10.1007/978-1-4419-6142-6.

Bertraundt (no date) *Scenario-based test of automated driving functions*. Available at: <https://www.bertraundt.com/en/scenario-based-test> (Accessed: February 28, 2022).

Bhardwaj, R. (no date) *MPLS vs Internet – Difference between MPLS and Internet*. Available at: <https://ipwithease.com/mpls-vs-internet/>.

Boyes, H. *et al.* (2018) "The industrial internet of things (IIoT): An analysis framework," *Computers in Industry*, 101(March), pp. 1–12. doi:10.1016/j.compind.2018.04.015.

BSI (2020a) "BS ISO 20901 : 2020 BSI Standards Publication Intelligent transport systems — Emergency electronic brake light systems ( EEBL ) — Performance requirements and test procedures."

BSI (2020b) *BSI Connected and automated vehicles - Vocabulary BSI Flex 1890 v3.0:2020-10*. Available at: <https://www.bsigroup.com/en-GB/CAV/>.

BSI (2021) *BS ISO 22733-1:2021 Road vehicles - Test method to evaluate the performace of autonomous emergency braking systems, British Standards*. Available at: <https://bsol.bsigroup.com/PdfViewer/Viewer?pid=000000000030364890>.

Business Network Solutions (2021) *VLPS Options*. Available at: <https://business.bt.com/products/networking/> (Accessed: February 28, 2022).

Buttazzo, G.C. (2011) *Hard Real-Time Computing Systems*. Third, Springer. Third. Edited by J.A. Stankovic.

CARLA Simulator (2022) *CARLA ScenarioRunner*. Available at: <https://carla-scenariorunner.readthedocs.io/en/latest/> (Accessed: February 28, 2022).

Chang, K.-H. (2015) "Motion Analysis," *Science Direct*, pp. 391–462. doi:10.1016/B978-0-12-382038-9.00008-9.

Change Vision (2021) *Change Vision, Inc*. Available at: [https://www.change-vision.com/index\\_en.html](https://www.change-vision.com/index_en.html) (Accessed: February 28, 2022).

Clark, G. and Stein, N. (2018) "Industrial Strategy: Automotive Sector Deal," *UK Government, Automotive Council UK*, pp. 1–32. Available at: [https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment\\_data/file/673045/automotive-sector-deal-single-pages.pdf](https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/673045/automotive-sector-deal-single-pages.pdf).

Cognata (2021) *Large-Scale Scenario Generation | AV and ADAS Simulation | Cognata*. Available at: <https://www.cognata.com/large-scale-scenario-generation/> (Accessed: February 28, 2022).

DDS Foundation (2021a) *What is DDS?* Available at: <https://www.dds-foundation.org/what-is-dds-3/> (Accessed: February 28, 2022).

DDS Foundation (2021b) *What is DDS?, Object Management Group*. Available at: <https://www.dds-foundation.org/what-is-dds-3/> (Accessed: February 25, 2022).

Demian, J. (2021) *What is network latency?* Available at: <https://sematext.com/blog/what-is-latency/>.

Department of Defense (2008) "Modeling and Simulation Body of Knowledge (BOK)," p. 182. Available at: <http://www.acqnotes.com/Attachments/DoD M&S Book of Knowledge.pdf>.

Department of Defense (2011) *Department of Defense Modeling and Simulation (M&S) Glossary*. Available at: [www.msco.mil](http://www.msco.mil).

Dey, E. (2021) *What Is Low Latency and Why It Is Important?* Available at: <https://www.arvancloud.com/blog/en/what-is-low-latency-why-it-is-important/#:~:text=Lower latency in the network,it is called low latency.>

dSPACE GmbH (2022a) *ASM ModelDesk - dSPACE*. Available at: [https://www.dspace.com/de/gmb/home/products/sw/automotive\\_simulation\\_models/produkte\\_asm/modeldesk.cfm#175\\_26324](https://www.dspace.com/de/gmb/home/products/sw/automotive_simulation_models/produkte_asm/modeldesk.cfm#175_26324) (Accessed: February 28, 2022).

dSPACE GmbH (2022b) *Scenario Generation - dSPACE*. Available at: [https://www.dspace.com/en/inc/home/applicationfields/our\\_solutions\\_for/driver\\_assistance\\_systems/data\\_driven\\_development/scenario\\_generation.cfm](https://www.dspace.com/en/inc/home/applicationfields/our_solutions_for/driver_assistance_systems/data_driven_development/scenario_generation.cfm) (Accessed: February 28, 2022).

Dupuis, M., Hekele, E. and Biehn, A. (2019) "OpenDRIVE® Format Specification, Rev. 1.5," *OpenDRIVE, (M)*, pp. 1-133.

ENISA (2020) *CYBERSECURITY STOCKTAKING IN THE CAM Stakeholder mapping and stocktaking of connected and automated mobility (CAM) cybersecurity*. Available at: <https://www.enisa.europa.eu/publications/cybersecurity-stocktaking-in-the-cam>.

esmini (2022) *Environment Simulator Minimalistic - a basic OpenSCENARIO player esmini* · *GitHub*. Available at: <https://github.com/esmini> (Accessed: February 28, 2022).

ETSI (2014) "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service," *Etsi*, 1, pp. 1-73. Available at: [http://portal.etsi.org/chaircor/ETSI\\_support.asp](http://portal.etsi.org/chaircor/ETSI_support.asp).

Euro NCAP (2017a) "Euro NCAP 2025 Roadmap: In pursuit of Vision Zero," *Euro NCAP Technical Paper*, pp. 1-19.

Euro NCAP (2017b) *EUROPEAN NEW CAR ASSESSMENT PROGRAMME (Euro NCAP) TEST PROTOCOL-SPEED ASSIST SYSTEMS*. Available at: <https://cdn.euroncap.com/media/32290/euro-ncap-sas-test-protocol-v20.pdf> (Accessed: March 17, 2022).

Euro NCAP (2020) *Euro NCAP | AEB Pedestrian, EURO NCAP*. Available at: <https://www.euroncap.com/en/vehicle-safety/the-ratings-explained/vulnerable-road-user-vru-protection/aeb-pedestrian/> (Accessed: February 25, 2022).

European Commission (2022) *Connected and automated mobility | Shaping Europe's digital future*. Available at: <https://digital-strategy.ec.europa.eu/en/policies/connected-and-automated-mobility#> (Accessed: February 28, 2022).

Exponential-e Ltd (2022) *MPLS vs VPLS Explained*. Available at: <https://www.exponential-e.com/mpls-vs-vpls#:~:text=MPLS offers increased flexibility and,in a way that optimises.>

Fagnant, D.J. and Kockelman, K. (2015) "Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations," *Transportation Research Part A: Policy and Practice*, 77(July), pp. 167–181. doi:10.1016/j.tra.2015.04.003.

Fagnant, D.J. and Kockelman, K.M. (2014) "The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios," *Transportation Research Part C: Emerging Technologies*, 40, pp. 1–13. doi:10.1016/J.TRC.2013.12.001.

Five AI (2021) *Autonomous vehicle development and safety assurance platform*. Available at: <https://www.five.ai/> (Accessed: February 28, 2022).

Foretellix (2022) *Technology - Foretellix*. Available at: <https://www.foretellix.com/technology/> (Accessed: February 28, 2022).

Fujimoto, R.M. (2000) *Parallel and Distributed Simulation Systems, Proceedings - Winter Simulation Conference*. JOHN WILEY & SONS, INC. doi:10.1109/WSC.2015.7408152.

Gangopadhyay, B. *et al.* (2019) "Identification of Test Cases for Automated Driving Systems Using Bayesian Optimization," *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019*, pp. 1961–1967. doi:10.1109/ITSC.2019.8917103.

Google Developers (no date) *Protocol Buffers*. Available at: <https://developers.google.com/protocol-buffers> (Accessed: March 17, 2022).

gRPC Authors (2022) *Introduction to gRPC*. Available at: <https://grpc.io/docs/what-is-grpc/introduction/> (Accessed: March 16, 2022).

Horváth, M.T. *et al.* (2019) "Vehicle-In-The-Loop (VIL) and Scenario-In-The-Loop (SCIL) Automotive Simulation Concepts from the Perspectives of Traffic Simulation and Traffic Control," *Transport and Telecommunication*, 20(2), pp. 153–161. doi:10.2478/ttj-2019-0014.

iASYS Technology Solutions (2022) *Scenario based testing for ADAS / Autonomous driving - iASYS Technology Solutions*. Available at: <https://www.iasys.co.in/scenario-based-testing/> (Accessed: February 28, 2022).

IBM (2021) *What is Middleware? | IBM, IBM Cloud Education*. Available at: <https://www.ibm.com/cloud/learn/middleware> (Accessed: February 25, 2022).

IEEE (1989) "IEEE Standard Glossary of Modeling and Simulation Terminology," *IEEE Std 610.3-1989*, p. 19. doi:10.1109/IEEESTD.1989.94599.

IEEE Standard (2010) "IEEE Std 1516™-2010, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Framework and Rules," *IEEE Standards* [Preprint].

IEEE Standard (2011) "INTERNATIONAL STANDARD ISO / IEC / IEEE 42010 Systems and software engineering - Architecture description," *IEEE Standards*, 2011.

IEEE Standard (2012) *IEEE Standard for Distributed Interactive Simulation--Application Protocols, IEEE Std 1278.1-2012 (Revision of IEEE Std 1278.1-1995)*. doi:10.1109/IEEESTD.2012.6387564.

International Council on Systems Engineering (INCOSE) (2015) *Systems Engineering Handbook*. Fourth. Wiley.

IPG Automotive GmbH (2022) *ScenarioRRR | IPG Automotive*. Available at: <https://ipg-automotive.com/en/products-solutions/engineering-services/scenariorrr/> (Accessed: February 28, 2022).

ISO (2013) "ISO 22839:2013 - Intelligent transport systems — Forward vehicle collision mitigation systems — Operation, performance, and verification requirements," *ISO* [Preprint]. Available at: <https://www.iso.org/standard/45339.html> (Accessed: February 28, 2022).

ISO (2017) *ISO 15623:2013 - transport systems — Forward vehicle collision warning systems — Performance requirements and test procedures, ISO*. Available at: <https://www.iso.org/standard/56655.html> (Accessed: February 28, 2022).

ISO/IEC (2011) "BS ISO/IEC 25010:2011 Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE)-System and software quality models," *British Standards* [Preprint].

Janet Network (2022) *Janet Network | Jisc*. Available at: <https://www.jisc.ac.uk/janet> (Accessed: February 28, 2022).

Jisc (2021) *Tech 2 Tech - low latency networking on Janet | Jisc*. Available at: <https://www.jisc.ac.uk/events/tech-2-tech-low-latency-networking-on-janet-27-oct-2021> (Accessed: February 28, 2022).

Junietz, P. *et al.* (2018) "Criticality Metric for the Safety Validation of Automated Driving using Model Predictive Trajectory Optimization," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, 2018-November*, pp. 60–65. doi:10.1109/ITSC.2018.8569326.

Juniper Networks Inc (2017) "Network Configuration Example." Available at: [https://www.juniper.net/documentation/en\\_US/release-independent/nce/topics/concept/vpls-understanding.html](https://www.juniper.net/documentation/en_US/release-independent/nce/topics/concept/vpls-understanding.html).

Khastgir, S. (2020) "PAS 1883:2020 Operational Design Domain (ODD) taxonomy for an automated driving system (ADS) – Specification," *BSI* [Preprint].

Lahiri, S.K. (2017) *Multivariable Predictive Control, Wiley*. Wiley. doi:10.1002/9781119243434.

Lee, E.A. and Seshia, S.A. (2015) *Introduction to Embedded Systems -- A Cyber-Physical Systems Approach*. Second. MIT Press. Available at: <http://leeseshia.org/>.

Levermore, T. and Peters, A. (2020) *Introducing VeriCAV: Addressing the challenges of virtual testing of Automated Vehicles*. Available at: <https://vericav-project.co.uk/> (Accessed: March 16, 2022).

MathWorks (2021) *Design driving scenarios, configure sensors, and generate synthetic data - MATLAB*. Available at: <https://www.mathworks.com/help/driving/ref/drivingsceniodesigner-app.html> (Accessed: February 28, 2022).

Menzel, T. et al. (2019) "From Functional to Logical Scenarios: Detailing a Keyword-Based Scenario Description for Execution in a Simulation Environment," *IEEE Intelligent Vehicles Symposium, Proceedings, 2019-June*, pp. 2383–2390. doi:10.48550/arxiv.1905.03989.

Menzel, T., Bagschik, G. and Maurer, M. (2018) "Scenarios for development, test and validation of automated vehicles," *arXiv [Preprint]*.

Minerva, R., Biru, A. and Rotondi, D. (2015) "Towards a definition of the {Internet of Things (IoT)," *IEEE Internet Initiative*, pp. 1–86.

Mooney, J. (1997) "Bringing portability to the software process," *Dept. of Statistics and Comp. Sci., West Virginia Univ., Morgantown WV*, pp. 559–570. Available at: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5961733>.

MQTT (2022) *MQTT - The Standard for IoT Messaging, MQTT Org*. Available at: <https://mqtt.org/> (Accessed: February 25, 2022).

NCAP (2020) *EUROPEAN NEW CAR ASSESSMENT PROGRAMME (Euro NCAP) TEST PROTOCOL-AEB VRU systems*. Available at: <https://cdn.euroncap.com/media/58226/euro-ncap-aeb-vru-test-protocol-v303.pdf> (Accessed: March 17, 2022).

Nickovic, D. and Herzner, W. (2019) *ENABLE S3 D47(D3.11)- D3.2.2 v3 V&V Methodology*.

Open Science Grid (2022) *MaDDash - Monitoring and Debugging Dashboard*. Available at: <https://psmad.opensciencegrid.org/maddash-webui/index.cgi?dashboard=UK%Mesh%Config> (Accessed: February 28, 2022).

Palo Alto Networks (2022) *MPLS | What Is Multiprotocol Label Switching*. Available at: <https://www.paloaltonetworks.com/cyberpedia/mpls-what-is-multiprotocol-label-switching#:~:text=MPLS Meaning,over private wide area networks.>

Paloalto Networks (2021) *Quality of Service*. Available at: <https://docs.paloaltonetworks.com/pan-os/8-0/pan-os-admin/quality-of-service> (Accessed: February 25, 2022).

Petty, M.D. (1995) "Computer-generated forces in distributed interactive simulation," *Distributed Interactive Simulation Systems for Simulation and Training in the Aerospace Environment: A Critical Review*, 10280, p. 102800I. doi:10.1117/12.204224.

Puthuff, N. (2021) *The Indy Autonomous Challenge: Achieving Extreme Performance with ROS 2, RTI*. Available at: <https://www.rti.com/blog/the-indy-autonomous-challenge-achieving-extreme-performance-with-ros-2> (Accessed: March 1, 2022).

Raab, F. (2019) "System Under Test," *Encyclopedia of Big Data Technologies*, pp. 1663–1665. doi:10.1007/978-3-319-77525-8\_124.

Raczkowski, B.C. et al. (2020) "A MATLAB Simulink Based Co-Simulation Approach for a Vehicle Systems Model Integration Architecture," *SAE Technical Papers*, 2020-March(March). doi:10.4271/2020-01-0005.

SAE (2017) "J3087: Automatic Emergency Braking (AEB) System Performance Testing - SAE International," *SAE* [Preprint]. Available at: [https://www.sae.org/standards/content/j3087\\_201710/](https://www.sae.org/standards/content/j3087_201710/) (Accessed: February 28, 2022).

SAE (2021) "SAE J3016 - Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles," *SAE Technical Papers* [Preprint].

Safety Pool (2022) *Safety Pool - Powered by Deepen AI and WMG University of Warwick*. Available at: <https://www.safetypool.ai/> (Accessed: March 17, 2022).

Schramm, D., Bardini, R. and Hiller, M. (2017) *Vehicle dynamics Modeling and Simulation*. Second, Springer. Second. Berlin, Germany: Springer. doi:10.1007/978-3-662-54483-9.

Sivakumar, P. et al. (2015) "Model based design approach in automotive software and systems," *International Journal of Applied Engineering Research*, 10(11), pp. 29857–29865. doi:10.37622/ijaer/10.11.2015.29857-29865.

Sjöberg, J. et al. (2010) *Driver models to increase the potential of automotive active safety functions*, *IEEE European Signal Processing Conference*. Available at: <https://ieeexplore.ieee.org/document/7096411> (Accessed: March 16, 2022).

Soltani, A. and Reed, N. (2021) "Interoperable simulation across CAM Testbed UK," p. 17.

Thule, C. et al. (2018) "Co-Simulation : A Survey," *ACM Computing Surveys*, 51(3).

Tolk, A. (2012) "Distributed Simulation," in *Engineering Principles of Combat Modeling and Distributed Simulation*. First. JOHN WILEY & SONS, INC. doi:10.1002/9781118616727.ch7.

Tredinnick, R. et al. (2017) "Uni-CAVE: A Unity3D plugin for non-head mounted VR display systems," *Proceedings - IEEE Virtual Reality*, pp. 393–394. doi:10.1109/VR.2017.7892342.

Tutorialspoint (2021) *Computer Network Topologies*. Available at: [https://www.tutorialspoint.com/data\\_communication\\_computer\\_network/computer\\_network\\_topologies.htm#:~:text=A Network Topology is the,different in a same network.](https://www.tutorialspoint.com/data_communication_computer_network/computer_network_topologies.htm#:~:text=A Network Topology is the,different in a same network.)

Unity Technologies (2022) *Unity - Scripting API: Time.fixedDeltaTime*. Available at: <https://docs.unity3d.com/ScriptReference/Time-fixedDeltaTime.html> (Accessed: February 28, 2022).

Venu, M. and Joe, I. (2014) "Improving performance of optimistic simulation for distributed simulation system using speculative computation," *International Conference on ICT Convergence*, pp. 428–432. doi:10.1109/ICTC.2014.6983173.

Vicon (2019) "Vicon Tracker 3."

le Vine, S. *et al.* (2016) "Automated cars: Queue discharge at signalized intersections with 'Assured-Clear-Distance-Ahead' driving strategies," *Transportation Research Part C: Emerging Technologies*, 62, pp. 35–54. doi:10.1016/J.TRC.2015.11.005.

VIRES (2021) *Features - VTD*. Available at: <https://vires.mscsoftware.com/features/> (Accessed: February 28, 2022).

Vogel, S. *et al.* (2019) "Improvements to the Co-simulation Interface for Geographically Distributed Real-time Simulation," *IECON Proceedings (Industrial Electronics Conference)*, 2019-October, pp. 6655–6662. doi:10.1109/IECON.2019.8926918.

Wang, Wenguang, Tolk, A. and Wang, Weiping (2009) "The levels of conceptual interoperability model: Applying systems engineering principles to M&S," *Spring Simulation Multiconference 2009 - Co-located with the 2009 SISO Spring Simulation Interoperability Workshop* [Preprint].

X-Engineer (no date) *What is the V model for software development - x-engineer.org, X-Engineer.Org*. Available at: <https://x-engineer.org/v-model-software-development/> (Accessed: February 28, 2022).

Zhang, X. *et al.* (2021) "Test Framework for Automatic Test Case Generation and Execution Aimed at Developing Trustworthy AVs from Both Verifiability and Certifiability Aspects," *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, 2021-September*, pp. 312–319. doi:10.1109/ITSC48978.2021.9564542.

Zhang, X., Khastgir, S. and Jennings, P. (2020) "Scenario Description Language for Automated Driving Systems: A Two Level Abstraction Approach," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2020-October, pp. 973–980. doi:10.1109/SMC42975.2020.9283417.

To find out more, please contact [info@zenzic.io](mailto:info@zenzic.io)