D4.1

Report on relevant subsystems to validate CCAM systems

Project short name
SUNRISE

Project full name
Safety assUraNce fRamework for connected, automated mobIlity SystEms

Horizon Research and Innovation Actions | Project No. 101069573
Call HORIZON-CL5-2021-D6-01

Funded by the European Union

ccam-sunrise-project.eu/

Dissemination level	Public (PU) - fully open
Work package	WP4: CCAM V&V framework
Deliverable number	D4.1: Report on relevant subsystems to validate CCAM systems
Status - Version	Final – V1.0
Submission date	30/08/2023
Keywords	simulation framework, safety assurance framework, automated driving systems, scenario-based testing, verification and validation

Authors

Role	Name
Main author:	Anders Thorsén (RISE)
Contributing authors:	Dino Dodig, Philippe Nitsche (AVL), Sara Messara (CAF), Mirko Muro (CRF), Gabriel Villalonga Pineda (CVC), Ilias Panagiotopoulos (ICCS), Georg Stettinger (IFAG), Anastasia Bolovinou (ICCS), Daniel Becker, Philipp Legran (ika), Ashfaq Farooqui, Sten Flystedt, Martin Skoglund, Fredrik Warg (RISE), Hakim Mohellebi (RSA), Tajinder Singh (SISW), Edwin van Hassel (SISW), Mohsen Alirezaei (SISW), Shabbir Ali (VED), Patrick Weißensteiner (ViF), Jason Zhang (WMG)

Quality Control

	Name	Organisation	Date
Peer review 1	Tajinder Singh, Edwin van Hassel, Mohsen Alirezaei, Sander de Vries	SISW	19/07/2023
Peer review 2	Bernhard Hillbrand	ViF	26/07/2023

Version history

Version	Date	Author	Summary of changes
0.1	23/03/2023	Anders Thorsén (RISE)	First draft.
0.2	06/05/2023	Anders Thorsén (RISE), Dino Dodig (AVL), Philipp Legran (ika), Georg Stettinger (IFAG), Tajinder Singh (SISW), Shabbir Ali (VED), Patrick Weißensteiner (ViF)	General updates, Adding of subsystems descriptions

0.3	20/06/2023	Anders Thorsén, Sten Flystedt (RISE), Dino Dodig, Philippe Nitsche, (AVL), Sara Messara (CAF), Mirko Muro (CRF), Gabriel Villalonga Pineda (CVC), Georg Stettinger (IFAG), Philipp Legran (ika), Hakim Mohellebi (RSA), Tajinder Singh (SISW), Shabbir Ali (VED), Patrick Weißensteiner (ViF)	General updates, Adding of subsystems descriptions and appendix.
0.4	28/06/2023	Anders Thorsén (RISE), Philippe Nitsche (AVL)	General editing, restructuring, and finalizing missing sections.
0.5	30/06/2023	Anders Thorsén (RISE), Anastasia Bolovinou (ICCS)	Clean up, minor changes and additions.
0.6	30/06/2023	Anders Thorsén (RISE)	Version for peer review
0.7	23/08/2023	Anders Thorsén, Ashfaq Farooqui (RISE), Ilias Panagiotopoulos (ICCS), Philipp Legran (ika), Georg Stettinger (IFAG), Tajinder Singh (SISW), Shabbir Ali (VED), Patrick Weißensteiner (ViF), Jason Zhang (WMG)	Updates after peer review
0.8	23/08/2023	Anders Thorsén (RISE)	For coordinator check
0.9	29/08/2023	Anders Thorsén (RISE)	Final draft.
1.0	29/08/2023	Anders Thorsén (RISE)	Final version.

Legal disclaimer

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY OF TERMS

Term:	Description	Source
ADS feature	An automated driving system's (ADS's) design-specific functionality at a given level of driving automation within a particular ODD, if applicable.	SAE J3016:2021 [1]
Automated driving system (ADS)	Hardware and software that are collectively capable of performing the entire dynamic driving task (DDT) on a sustained basis, regardless of whether it is limited to a specific operational design domain (ODD).	ISO 34501:2022 [2]
Concrete scenario	Parameterised model of the time sequence of scenes (logical scenario) which begins with an initial scene and defined point in time; the behaviour of the main actor (vehicle under test) is not further specified.	SUNRISE Glossary
Current operational domain (COD)	Real-time real-world conditions that the ADS is experiencing.	ISO 34503:2023 [3]
Dynamic driving task (DDT)	All of the real-time operational and tactical functions required to operate a vehicle in on-road traffic, excluding the strategic functions such as trip scheduling and selection of destinations and waypoints.	ISO 34501:2022 [2]
Logical scenario	Beginning with an initial scene, a model of the time sequence of scenes whose parameters are defined as ranges; at a defined point in time, the behaviour of the main actor (vehicle under test) is not further specified.	SUNRISE Glossary
Operational design domain (ODD)	The 'boundaries of the operating environment within which the ADS can operate, performing the DDT safely.'	SUNRISE Glossary
Operational domain (OD)	Real-world conditions that an ADS may experience.	ISO 34503:2023 [3]
Safety test objective	Safety property of the ADS to be shown via a set of tests.	ISO 34502:2022 [4]
Scenario	Description of a temporal and spatial traffic constellation.	SUNRISE Glossary

Subject vehicle	Vehicle under observation in the process of testing, evaluation, or demonstration.	ISO 34501:2022 [2]
Subsystem	Part of a system, which is itself, a system.	IEC 60050 [5], IEV 192-01-04 (Dependability domain)
System	Set of interrelated items that collectively fulfil a requirement, within a defined real or abstract boundary, whereas external resources (from outside the boundary) may be required to operate.	IEC 60050 [5], IEV 192-01-03 (Dependability domain)
System under test	ADS that is tested with test scenarios.	ISO 34501:2022 [2]
Test case	Set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement.	SUNRISE Glossary
Test scenario	scenario intended for testing and assessing automated driving system(s) (ADS)/subject vehicle(s).	ISO 34501:2022 [2]
Traffic agents	Anyone who uses a road including sidewalk and other adjacent spaces.	ISO 34503:2023 [3]

ABBREVIATIONS AND ACRONYMS

Abbreviation	Meaning
AD	Automated Driving
ADAS	Advanced Driving Assistant System
ADS	Automated Driving System
AV	Automated Vehicle
CCAM	Connected, Cooperative, and Automated Mobility
COD	Current Operational Domain
C-V2X	Cellular-V2X
DDT	Dynamic Driving Task
DSRC	Dedicated Short-Range Communication
FOV	Field Of View
GT	Ground Truth
KPI	Key Performance Index
OBU	On-Board Unit
OD	Operational Domain
ODD	Operational Design Domain
OEM	Original Equipment Manufacturer
OSI	Open Simulation Interface
RSU	Road Side Unit
SAF	Safety Assurance Framework
SOTIF	Safety Of The Intended Function
SUT	System Under Test
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
V&V	Verification and Validation
VRU	Vulnerable Road User
XiL	X in the Loop

EXECUTIVE SUMMARY

Safety assurance of cooperative, connected, and automated mobility (CCAM) systems is crucial for their successful adoption in society, and it is necessary to demonstrate reliability in their complete operational design domains (ODD). For higher level of automation, i.e., when the vehicle takes over the responsibility from the human driver, it is commonly accepted that validation only by means of real test-drives would be infeasible. Instead, a mixture of physical and virtual testing is seen as a promising approach, in which the virtual part accelerates testing procedure and significantly reduces cost. This in turn accelerates the time to market.

The SUNRISE project aims to develop a Safety Assurance Framework (SAF) for scenario-based safety validation of CCAM systems, covering a broad portfolio of use cases and comprehensive test and validation tools. Part of this project focuses onto developing a harmonised verification and validation (V&V) simulation framework for CCAM systems. To overcome the limitations of virtual simulation, the targeted SAF also will include hybrid and real-world testing and validation approaches.

This deliverable presents the findings from the task to identify relevant subsystems of a harmonised V&V simulation framework for virtual validation of CCAM systems applying a scenario-based testing methodology. The involved partners have together identified and agreed on a non-exclusive list of relevant subsystems: (1) test case manager, (2) environment, (3) subject vehicle, (4) traffic agents, (5) connectivity, and (6) simulation model validation. The subject vehicle subsystems include blocks for sensors, AD function, and vehicle dynamics and the AD function block includes subblocks for perception, planning, and control and act.

This deliverable primarily focuses on virtual simulations, but the SAF also covers XiL tests, were some of the listed subsystems can be replaced with the real components. After the subsystems are described, the subsystem requirements are analysed form the perspective of requirements on tools, interfaces, V&V of the simulation framework, and model fidelity. Many of the participants have experience in simulation tools, but the presented work is mainly theoretical, and the actual development of the simulation framework is done in subsequent tasks of WP4. The intention is that the definition of the simulation framework and the listed subsystems shall be versatile and adoptable for future technology development.

1 INTRODUCTION

1.1 Project intro

Safety assurance of cooperative, connected, and automated mobility (CCAM) systems is crucial for their successful adoption in society, yet it remains being a significant challenge.

CCAM systems need to demonstrate reliability in their complete operational design domains (ODD), requiring robust safety argumentation. It is already acknowledged that for higher levels of automation, i.e., when the vehicle takes over the responsibility from the human driver, the validation of these systems by means of real test-drives would be infeasible. In consequence, a carefully designed mixture of physical and virtual testing has emerged as a promising approach, with the virtual part bringing significant weight in this mixture for cost efficiency reasons and accelerates testing procedure. Several worldwide initiatives have started to develop test and assessment methods for automated driving (AD) functions. These initiatives have already moved from conventional validation to a scenario-based approach and combine different test instances (physical and virtual testing) to avoid the million-mile issue.

The initiatives mentioned above provide new approaches to CCAM validation, and many expert groups formed by different stakeholders are already working on CCAM system testing and quality assurance. Nevertheless, the fact that there is a lack of a common European validation framework and homogeneity regarding validation procedures to ensure safety of these complex systems, hampers the deployment of CCAM solutions. In this landscape, the role of standards is paramount in establishing common ground and providing technical guidance. However, standardising the whole pipeline of CCAM validation and assurance is in its infancy, as many of the standards are under development or have been very recently published and still need time to be synchronised and established as common practice.

Scenario databases are another issue tackled by several initiatives and projects, providing silo solutions. A single concrete approach should be used (at least at the European level), dealing with scenario databases with different possible variations, including the creation, editing, parameterisation, storing, exporting, importing, etc. in a universally agreed manner.

Furthermore, validation methods and testing procedures still lack appropriate safety assessment criteria to build a robust safety case and be valid for the whole parameter space of scenarios. Another level of complexity is added, due to regional differences in traffic rules, signs, actors, and situations.

Evolving from the achievements obtained in HEADSTART [6, 7] and taking other initiatives as a baseline, there is a need to move to the next level in the concrete specification. It is necessary to demonstrate a commonly accepted Safety Assurance Framework (SAF) for the safety validation of CCAM systems, including a broad portfolio of use cases [8] and comprehensive test and validation tools. This will be done in SUNRISE, which stand for **S**afety ass**U**ra**N**ce f**R**amework for connected, automated mob**I**lity **S**yst**E**ms.

The SAF is the main element to be developed in the SUNRISE project. This framework takes a central role, fulfilling the needs of different automotive stakeholders that all have their own interests in using it. The overall objective of the SUNRISE project is to accelerate the safe deployment of innovative CCAM technologies and systems for passengers and goods by creating demonstrable and positive impact towards safety. It specifically focuses on the EU's long-term goal of moving close to zero fatalities and serious injuries by 2050 (Vision Zero), and the resilience of (road) transport systems. The project aims to achieve this by creating and sharing a European federated database framework centralising detailed scenarios for testing of CCAM functions and systems. The framework provides a multitude of relevant test cases, based on a virtual harmonised simulation framework combined with hybrid and real-world testing and validation approaches, with standardised, open interfaces and quality-controlled data exchange.

1.2 Purpose of the deliverable

Work package 4 in SUNRISE aims to develop a harmonised verification and validation (V&V) simulation framework for CCAM systems. To overcome the limitations of virtual simulation, the targeted SAF also includes hybrid and real-world testing and validation approaches.

This deliverable presents the work done in the first task of work package 4: Identifying relevant subsystems of a harmonised V&V simulation framework for virtual validation of CCAM systems. The partner contributions to this deliverable are summarized in Table 1.

Table 1. Partner contribution to D4.1.

Partner	Role
RISE	RISE is task leader and main editor for the deliverable.
AVL	AVL is main contributor to the Environmental subsystem description and AD function description.
CAF	CAF has contributed with system-level consistence check of the subsystems.
CRF	CRF has analysed the simulation framework out of V&V requirements, tool requirements, and model fidelities.
CVC	CVC has contributed to the Environmental subsystem description.
ICCS	Participated in meetings and checking the deliverable.
IFAG	IFAG has contributed with by defining the necessary subsystems focusing on sensors (e.g., Radar) and their respective models.
Ika	Ika is the main contributor to the traffic agents' simulator subsystem description, and for the test data analysis and validation description.
RSA	Renault has contributed with the OEM perspective.
SISW	Siemens is the main contributor for the test case manager subsystem and SOTIF evaluation.
WMG	WMG has contributed to the test case manager subsystem.
VED	VEDECOM is the main contributor to the Connectivity subsystem description.
ViF	VIF is the main contributor to the vehicle dynamics subsystem description and with short description of scenario selection and allocation process.

As the task shall identify relevant subsystems of the SUNRISE simulation framework it is important to define what a *subsystem* is. This was discussed in the initial task execution of task 4.1. It was concluded that the definition from the dependability domain found in IEC 60050 IEV 192-01-04 [5] suites the purpose well. This definition states that a subsystem is:

> *part of a system, which is itself, a system*

while system is defined as (IEC 60050 IEV 192-01-03 [5])

> *set of interrelated items that collectively fulfil a requirement*

with the notes that "a system is considered to have a defined real or abstract boundary", "external resources (from outside the system boundary) may be required for the system to operate", and "a system structure may be hierarchical, e.g., system, subsystem, component, etc".

1.3 Intended audience

The intended audience of the deliverable is primarily the partners involved in work package 4. It is also considered relevant for other partners involved in the SUNRISE project as well as for readers outside the SUNRISE consortium.

1.4 Structure of the deliverable and its relation with other work packages/deliverables

This deliverable is structured as follows: Chapter 2 gives an overview of the SUNRISE Harmonized CCAM V&V Simulation Framework, Chapter 3 describes the identified relevant subsystems, Chapter 4 discuss subsystem requirements from a SAF perspective, and finally Chapter 5 summarises the conclusions.

The context of this deliverable is defined by the SAF to be defined in WP2 and the scenario-based methodology to be defined in WP3, which both are work in progress at the time of writing of this deliverable. This deliverable gives input to subsequent tasks in WP4.

2 SUNRISE SIMULATION FRAMEWORK

2.1 The simulation framework in the SUNRISE context

The SUNRISE harmonized V&V simulation framework is a fundamental part of the SUNRISE SAF, with the scenario-based SUNRISE methodology to be defined in WP3. At this stage of the project, both the SUNRISE SAF and the scenario-based SUNRISE methodology are still in the definition phase. Anyhow, for Task 4.1 it is a need to understand the context and a conceptual illustration of the simulation framework's role in a scenario-based approach is shown in Figure 1. The figure does not reflect the actual definition of either SUNRISE SAF or the scenario-based SUNRISE methodology.

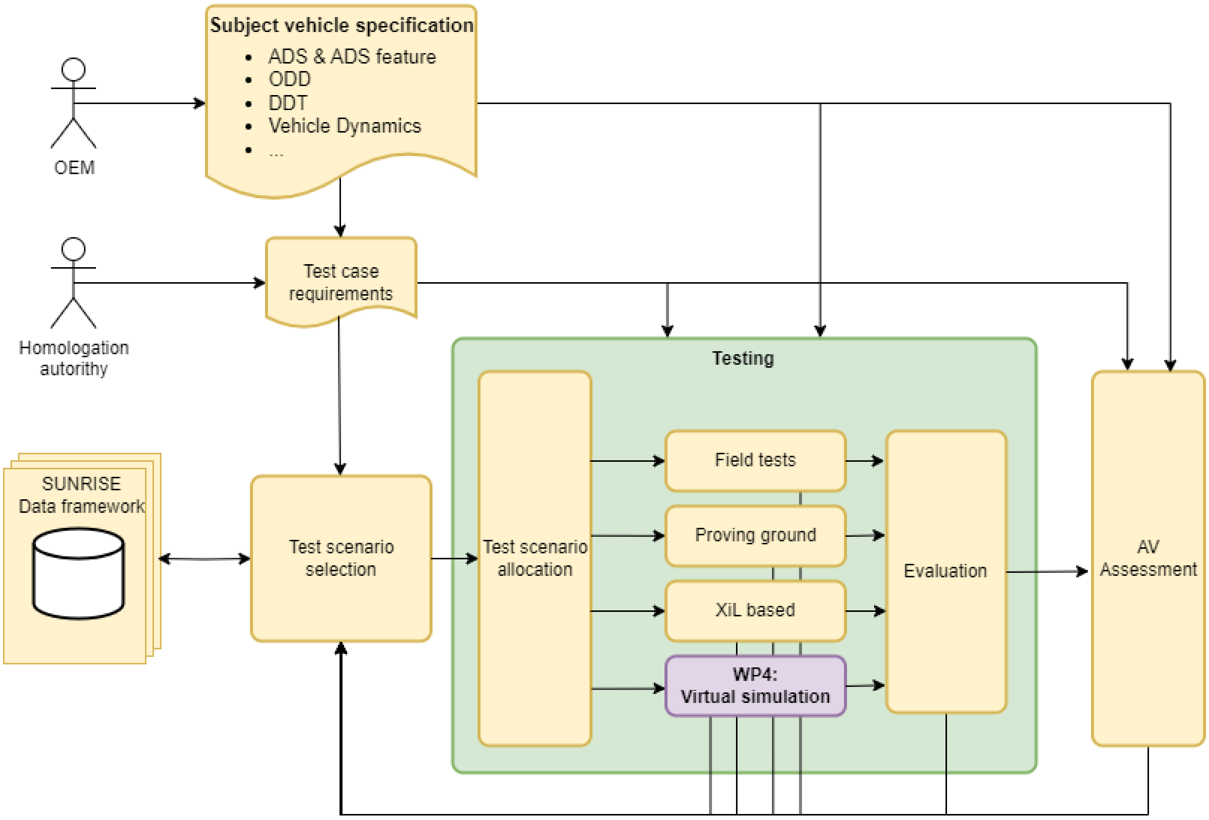

Figure 1. Conceptual illustration of the virtual simulation in the scenario-based methodology.

2.2 High level simulation framework architecture

The fundamental basis of the design of automated driving systems (ADS), lies in a specific set of requirements that outline how the vehicle is expected to behave in certain conditions (i.e., the ODD). It is essential that the ADS enables the vehicle to provide the anticipated AD function in a secure and safe manner within its designated ODD. To ensure the safety of the passengers and the other users on the road, the vehicle must be capable of effectively managing any situations that could occur within the ODD for which the ADS has been designed. In a simulation framework, the tools must allow for the precise definition of the ODD for which automated vehicles (AVs) are designed.

During the execution of task 4.1, the partners have proposed relevant subsystems, compared different inputs, and discussed and reached a common understanding resulting in a non-exclusive list of relevant subsystems. Figure 2 illustrates the SUNRISE Simulation Framework with the identified subsystems. Note that the figure does not reflect the actual definition of either SUNRISE SAF or the scenario-based SUNRISE methodology, that both are under development.

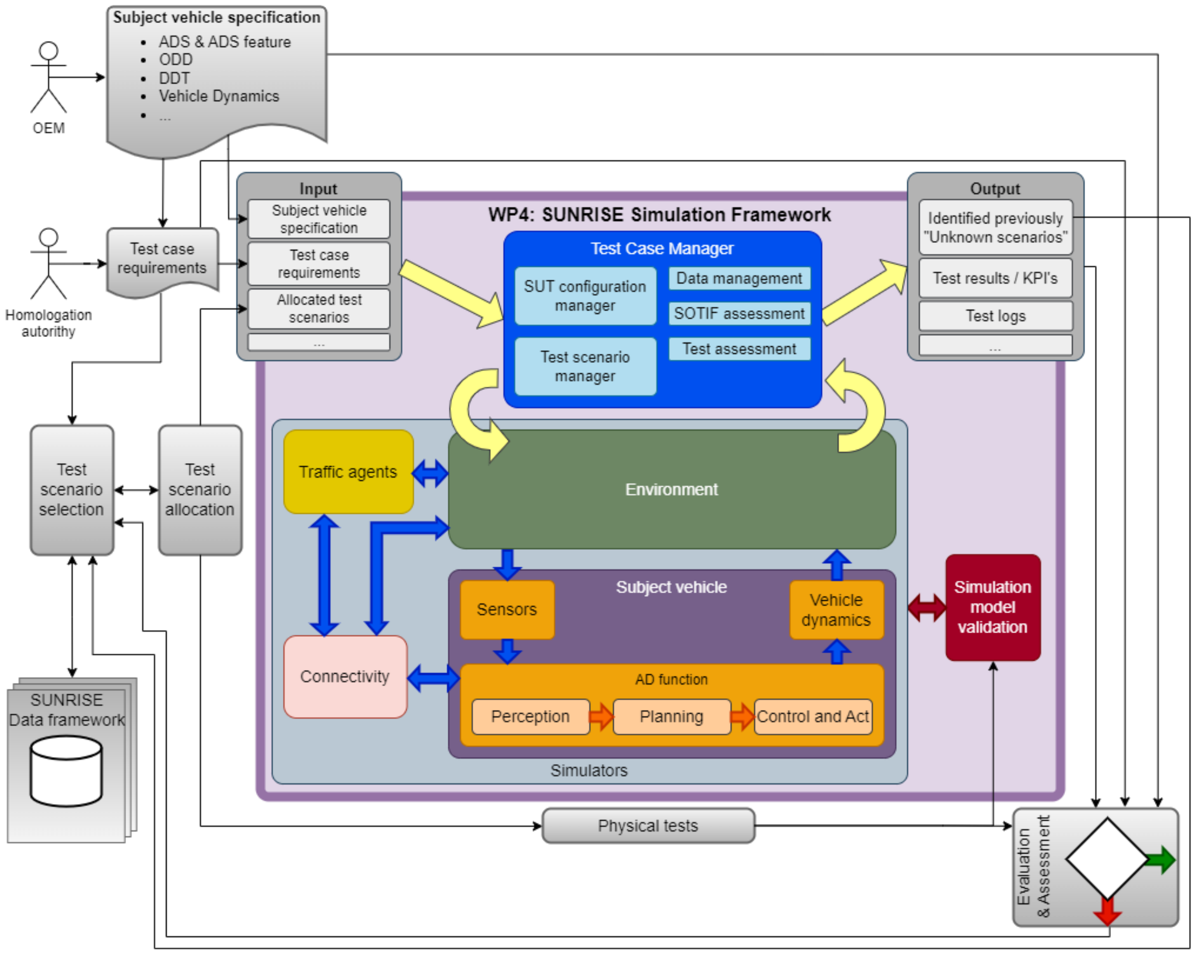

Figure 2. Conceptual view of the SUNRISE Simulation Framework with identified subsystems.

The intention is that the defined SUNIRSE simulation framework, as well as included subsystem, shall be versatile and adoptable for future technology development. If needed, additional subsystems may be added.

The identified subsystems include:

- Test case manager
- Environment
- Subject vehicle
    - Sensors
    - AD function
        - Perception
        - Planning
        - Control and Act
    - Vehicle dynamics
- Traffic agents
- Connectivity
- Simulation model validation

Each subsystem is described in more detail in Chapter 3. It is expected that the subsystems commonly are run in co-simulation, i.e., each of the subsystems is simulated individually, but they are executed simultaneously in a synchronized manner while exchanging relevant data.

Figure 2 illustrates the "Sensors", "AD function" and "Vehicle dynamics" subsystems framed in the Subject vehicle. This is more an illustrative grouping of the subsystems inside the vehicle, than a subsystem itself. It may also be argued that the Connectivity subsystem should be partly included in the Subject vehicle, though it is not illustrated like that in Figure 2.

The grey parts outside and partly outside the box named "WP4: SUNRISE Simulation Framework" are not part of the simulation framework but illustrate relevant items needed for the simulation framework. The figure does not show the actual design of these grey parts.

2.2.1 Simulation framework input and output data

The simulation framework defined in WP4 will need to exchange data with the rest of the SAF.

Input information needed to the simulation framework is foreseen to include (among other) the items listed below. Focus is on information content rather than terminology. This list is not exclusive.

- *Subject vehicle specification*
    - *ADS feature* [1]
      An automated driving system's (ADS's) design-specific functionality at a given level of driving automation within a particular ODD.
    - *Dynamic Driving Task (DDT)* [2]
      All of the real-time operational and tactical functions required to operate a vehicle in on-road traffic, excluding the strategic functions such as trip scheduling and selection of destinations and waypoints.
    - *Operational Design Domain (ODD)* [2]
      The operating conditions under which the ADS or ADS feature is designed to operate and perform the DDT safely and effectively. It includes various factors such as road types, weather conditions, traffic densities, and speed ranges. The ODD defines the boundaries of the AVs capabilities and limitations, and

the system should only be activated within its designated ODD. The ODD may change depending on the level of automation, with higher levels having a broader and more complex ODD.

    - *Vehicle information not covered above*
      Vehicle information needed for the simulations not covered above, e.g., to enable vehicle dynamic modelling or 3D modelling.

- *Test Case Requirements*
    - *Safety Test Objective* [4]
      Safety property of the ADS to be shown via a set of tests. The safety test objectives can be derived from the validation targets or the acceptance criteria of ISO 21448 [1].
- *Selected scenarios allocated to virtual simulations*
  The selected test scenarios allocated to virtual simulations are needed as input. The actual test scenario selection and allocation takes place outside the SUNRISE simulation framework. For details, see SUNRISE deliverable D3.1 [9].
- *Test data from physical tests (arrow to the "Simulation model validation" subsystem)*
  Test data from physical tests used for validation of the simulation models.

Output data from the simulation framework includes, e.g.:

- *Test results / KPIs*
  The test results, e.g., calculated Key Performance Indexes (KPIs).
- *Identified (previously) unknown scenarios.*
  Unknown scenarios [10] identified through an unknown-unsafe identification process as required by the ISO 21448 safety of intended functionality (SOTIF) standard [10]. The search for these unknown unsafe scenarios is based on the ODD and (when available) recorded data in the ODD (see Annex 2 for more details).
- *Test logs.*
  Detailed documentation of the testing process, including the test case execution results, the configurations, and the issues encountered during testing.

3 SUBSYSTEMS DESCRIPTIONS

In the following, the identified relevant subsystems for the SUNRISE harmonized V&V simulation framework, as shown in Figure 2, are described. It should be noted that it is a non-exclusive list and that not all subsystems are needed to be implemented in a certain test.

The system under test (SUT) denotes the ADS that is tested with test scenarios.

3.1 Test case manager

In this section main functionality, composition and interfaces of test case manager are described.

3.1.1 Main function

The test case manager has the following key roles:

1. Interface the simulation framework with the rest of the SAF: Manage inputs to the simulation framework, e.g., ODD and test scenarios, as well as outputs from the simulation framework, e.g., test results. It also connects with the scenario databases for scenario retrieval based on test instruction.
2. Orchestrate execution of test scenarios in the simulation framework: Performs tasks such as configuration of the simulation models and system under test, execution of the simulation models, and retrieval of simulation data. Other tasks may include allocating necessary resources such as computing capabilities.
3. Perform SOTIF assessment: Identify unknown-unsafe scenarios for the ODD and the DDT as required by the ISO 21448 (SOTIF) standard [10]. The search for these unknown unsafe scenarios is based on the ODD and (when available) recorded data in the ODD (see details in Annex 2). The set of unknown-unsafe scenarios are used as part of the test scenarios to be executed in the simulation framework and the new unsafe scenarios (once identified) are used to expand the scenario database.
4. Compute KPIs and metrics for the test scenarios from the simulation data.
5. Perform checks on whether the test scenario simulation has been executed correctly.

The implementation of the test case manager subsystem should consider both usability and standardization of interfaces. Firstly, the usability of the test case manager subsystem may be enhanced with an intuitive user interface, which provides visualization tools for monitoring status and health of test execution and analyzing test results and identified issues in the SUT. Second, standard interfaces and data format ensure that the test case manager subsystem is modular and can be easily integrated into the simulation framework. The use of standard interfaces and data format also support integration of the subsystem in external simulation frameworks for the development and validation process of AVs, to accelerate collaboration with OEMs.

3.1.2 Composition

Following functions are considered relevant to include in the test case manager:

- Test scenario manager:
    - Orchestrate the execution of test cases, including allocated test scenarios, by providing necessary inputs to other subsystems and triggering test execution.
    - For simulations it may in some cases be beneficial to provide the concrete test scenarios as logical scenarios for which concrete parameter values are chosen by the scenario manager out of ranges using predefined steps, distribution functions, or randomly selecting.
- Data management:
    - Store input data to the simulation framework such as the ODD, test scenarios, and description of dynamic driving task.
    - Manage simulation data (states of actors, other metrics related info) retrieved from other simulation subsystems during execution of test cases.
- Test assessment:
    - Perform checks to validate that the test scenario simulation has been executed correctly, e.g., for a cut-in test scenario, check the state of the cut-in vehicle and the subject vehicle to validate that the scenario is correctly simulated.
    - Perform calculations on simulation data to create results for test assessment KPIs, metrics. The test assessment KPIs and metrics (safety test objectives) may vary based on ODD, type of test scenario, and dynamic driving task. The KPIs and metrics are provided as an input to the simulation framework., See *Annex 1 Test data analysis and validation* for details.
- SOTIF assessment:
    - Identify unknown-unsafe scenarios as required by the ISO 21448 “Safety of the intended functionality” (SOTIF) safety standard [10]. The workflow extracts a search space from the ODD (and other relevant information) and discovers unknown-unsafe scenarios within the search space. The set of unknown unsafe scenarios are to then be used as additional test scenarios for the simulation framework. See *Annex 2 SOTIF assessment block in the simulation framework* for details.
- SUT configuration manager:
    - Modify the SUT configuration according to the DDT / use case / or for a specific assessment such as SOTIF assessment. Besides selecting the SUT models corresponding to the DDT and use case, the configuration of the SUT also includes selecting which components of the AV are part of the SUT. It also specifies the level of fidelity of the components which is necessary for the test cases or for specific assessments.

3.1.3 Interfaces

Interfaces foreseen for the test case manager (as shown in Figure 2) are:

- Input:
    - Test scenarios from scenario database
        - Test scenarios from scenario database in a standardized format, e.g., ASAM OpenSCENARIO format.

  - SUT specification and test requirements
    - ODD description in a machine-readable standardized format, e.g., OpenODD format. The ODD description also considers the elements found within the road network in a standardized format, e.g., OpenDRIVE format.
    - Collected COD (Current Operational Domain) data from the scenario execution in a standardised format, which can be checked and monitored against the ODD (if any) provided in a standardized format, e.g., using the *followTrajectory* action in OpenSCENARIO format. The collected data in the ODD increases the understanding of the ODD and can be used, for example, by the SOTIF assessment block, to extract representative actor behaviour in the ODD.
    - Test assessment KPIs, metrics (safety test objective) corresponding to the DDT and the ODD are used to assess the SUT with respect to the test scenarios.
    - Dynamic Driving Task / SUT configuration: description of the AV functionality and the system under test configuration.
  - Simulation data from other subsystems of the simulation framework:
    - Simulation data which is needed for calculation of KPIs and metrics, such as the states of actors during the simulation is received from other subsystems. The necessary simulation data to be received which should be provided to the test case manager depends on what is needed to calculate the test assessment KPIs and metrics.
  - The information may in the future be provided using the in-development standard ASAM OpenTest [11].
- Output:
  - DDT / SUT configuration to subsystems modelling the SUT:
    - DDT / SUT configuration is provided to the subsystems which model the SUT in the simulation framework such as sensors, AD functions, and vehicle dynamics.
  - ODD and test scenario to the Environment subsystem:
    - Scene information, road network, and concrete test scenario such that the environment model and virtual models of actors can be generated.
  - ODD and test scenario to the Traffic agent subsystem:
    - Road network and concrete test scenario such that the behaviour of dynamic actors can be modelled.
  - Test results of allocated test scenarios from scenario database:
    - Test results are provided from the simulation framework to the necessary modules in the SAF, such as the overall AV assessment module, where the results of simulation may be combined with results from other test methods.
  - Identified unknown-unsafe scenarios and their test results:
    - Unknown-unsafe scenarios and their test results are also provided to the SAF. The identified unknown scenarios are also linked back to the scenario database, such that the scenario database can expand with these previously unknown scenarios.

3.2 Environment

3.2.1 Main function

The main function of Environment subsystem is to describe the environment in which the AV operates. It serves as a base information in describing and interpreting everything from the outside that encompasses the AV. Depending on the desired vehicle functionalities and target virtual testing complexity, proper fidelity level of environment detail must be defined. For simple automated functions that are based only on detected objects and predicted trajectories, simple environment entities can be defined with only base shape and position. However, for more complex automated functionalities, detailed Environment must cover all necessary details. In other words, it must be sufficiently defined so that it covers the whole OD (Operational Domain) and ODD (Operational Design Domain) for the designed vehicle [3]. ODD represents the environment in which a function is intended to remain safe, while OD represents any condition whatsoever which the vehicle could encounter, thus in this case “OD” and the “Environment” have great similarity.

3.2.2 Composition

Environment subsystem’s composition can be split in two main categories:

- <u>Static environment:</u>
  Static environment encompasses everything that is static during vehicle operation and does not have any predefined dynamic behaviour that is relevant for the AV. In principle it includes everything in the ODD that ISO 34503:2023 [3] classify as scenery and environmental conditions.
    - Topological Road – defines representation of the road network and its connectivity. Topological road modelling emphasizes the relationships between road segments, intersections, and other key features of the road network along with railings, traffic signs and lights, obstacles, construction sites, static objects, etc.
    - Road Surface – defines all relevant road surfaces and shapes that are to be defined for a specific road network.
    - Photorealistic details – defines more advanced physics-based effects that are required to simulate, design and verify advanced perception systems and the effects on sense-plan-act algorithms.
    - Environment Material Properties – defines every relevant material surface of the environment to capture the necessary effects like texture, friction, reflection, elevation, or any irregularities that might impact the vehicle during operation.
    - Weather Conditions – defines surrounding weather conditions and properties to accurately capture their impact on the vehicle.
    - Illumination – defines the lighting conditions in which the vehicle will operate.

- Dynamic elements
  Dynamic elements encompasses everything that is movable elements of the ODD and two main categories are defined as listed below [3]. These elements are included in the environment simulation, but their behaviour is controlled by other subsystems.
    - Subject vehicle
      The behaviour of the subject vehicle is simulated by the subject vehicle simulators described in Sec. 3.3.
    - Traffic agents – all other agents. It may be motor vehicles, non-motor vehicles, vulnerable road users, and animals. They may also be classified as stationary (or parked for vehicle) or moving. The behaviour is simulated in the Traffic agents' subsystem described in Sec. 3.4.

3.2.3 Interfaces

Environment subsystem must include necessary interfaces with the complete subject vehicle model to transfer the conditions and system specific information about the vehicle's surroundings. The AV must be able to extract relevant data from the atmospheric surroundings through various sensors containing all relevant static and dynamic objects (Weather/Lighting conditions, objects positioning and dynamics).

In addition, if needed, the environment subsystem should have interfaces to the traffic agents' subsystem and the Connectivity subsystem to model, e.g., attenuation. The subsystem must also provide relevant information on all static and dynamic objects for all relevant scenario/ODD/"Test case manager" components. This is necessary to effectively pre-process and post-process the information during the V&V phase.

3.3 Subject vehicle

The Subject vehicle includes the "Sensors" subsystem, the "AD function" subsystem, and the "Vehicle Dynamics" subsystem.

3.3.1 Sensors

3.3.1.1 Main function

The sensor subsystem is a key element in enabling the ADS to provide both reliable vehicle localisation and robust environmental perception of the vehicle's surroundings within its ODD [12]. The following section focuses on environmental perception sensors, mainly cameras, radar and LIDAR, and their corresponding sensor models needed for virtual verification and validation tasks within the development process (the principle should be relevant also for other types of sensors). Sensor models will enable the reduction of conventional test drives and physical component testing with simulations in virtual test environments to meet the increasing demands of ADS in terms of development cost, time, and safety. Given the variety and complexity of possible environmental conditions, realistic simulation of perception sensors is a particularly challenging issue.

3.3.1.2 Composition

Sensor modelling is an important and complex task during development. Many different sensor modalities (radar, lidar and camera), as well as different model fidelities are required during the development process, to meet the requirements resulting from the target ODD and its associated behaviour capabilities.

Sensors can be classified using different principles. On possibility is to distinguishes between ideal models, probabilistic models, and physics-based models [13]. Another principle, that is further described in the following, is to distinguish in low-fidelity models, medium-fidelity models, and high-fidelity models, based on its inputs, outputs, and modelling principle [14]. Low-fidelity sensor models are based on geometric aspects such as sensor-specific field of view (FOV) and object occlusion. The input data format is object lists with ground truth information and the output data format is also object lists but with filtered ground truth (GT) information. Medium-fidelity sensor models consider some physical aspects of the real sensor and the material properties of the objects, as well as the sensor's field of view and detection probability. The input for medium fidelity sensor models is object lists corresponding to the ground truth. The output data formats are object lists or raw data processed according to the modelled perceptual effects. High-fidelity sensor models are the most accurate representations of real-world sensors. They incorporate rendering techniques such as rasterization or ray tracing. They combine environmental parameters and material properties with physical effects such as diffraction and interference. High-fidelity sensor models use the entire 3D virtual environment, a mesh describing objects and their surfaces, as input and produce sensor-specific raw data as output. They are computationally intensive and require more computing power, often at the expense of real-time capability. The output of these models can be set up to provide object lists or even raw data, see Figure 3 [15].

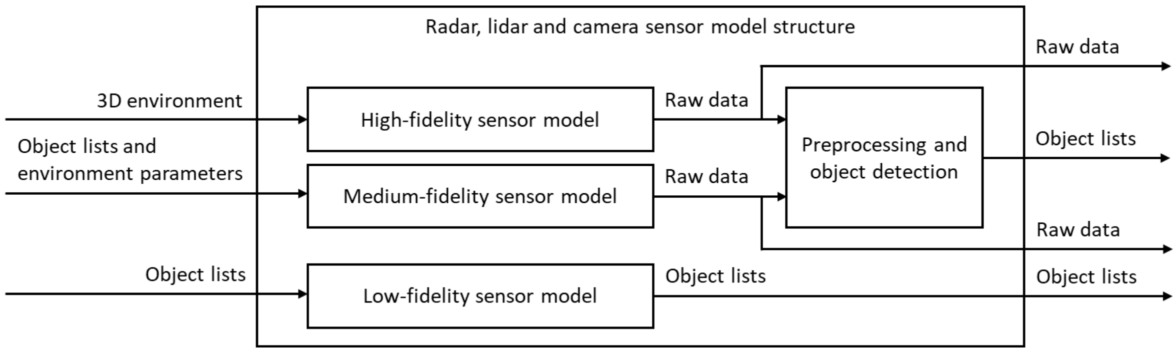

Figure 3. Sensor model classification and corresponding input and output interfaces.

3.3.1.3 Interfaces

Each sensor model fidelity defines input and output data format and the level of detail of the sensor model class. From low-fidelity to high-fidelity sensor models, the level of detail increases, i.e., the sensor model reproduces the behaviour of the real sensor in more detail. In that sense, typically also the required computational power increases. In the sense-plan-act cycle, the sense task is divided into measure and percept. Measure means capturing the environment and generating sensor raw data whereas percept as the second step of the sense task transforms sensor raw data into object lists (object detection). Low-fidelity and some medium-fidelity sensor models include both the task of measure and the task of percept. The output of such sensor models is the object lists. On the contrary, high-fidelity and the rest of the medium-fidelity sensor models include only the measure task. Therefore, its output is sensor raw data, which needs a further object detection algorithm to get the object lists [14].

The most common data formats for input and output are object lists and raw data. Object lists can be considered as a generic data format as all sensor models can consume and generate them with similar parameters. Objects as units of object lists are either static objects (e.g., guardrails, buildings, tunnels, trees) or dynamic objects (e.g., vehicles, trucks, VRUs) [14].

Compared to object lists, sensor raw data depends on the modelled sensor type. Radar data cubes are typically used as radar raw data, point clouds as lidar raw data, and images as camera raw data [14].

ASAM OSI (Open Simulation Interface) allows the connection of sensors, via a standardized interface, to any ADS and to any environment simulation tooling. ASAM OSI contains an object-based environment description using the message format of the protocol buffer library developed and maintained by Google. It defines top-level messages that are used to exchange data between separate models. Top-level messages define the GroundTruth interface, the SensorData interface, the SensorView/Sensor-View configuration interfaces and the FeatureData interface, see Figure 4.

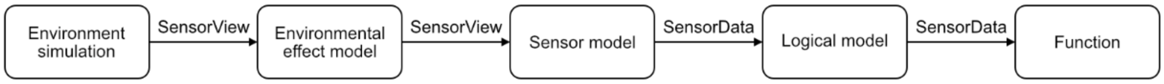

Figure 4. Overview Open Simulation Interface (OSI) [16].

The GroundTruth interface provides an exact view on the simulated objects in a global coordinate system, the ground truth world coordinate system. The FeatureData interface provides a list of simple features in the reference frame of the respective sensor of a vehicle for environmental perception. It is generated from a GroundTruth message and may serve as input for a sensor model that simulates object detection or feature fusion of multiple sensors. OSI also defines interfaces for traffic participant models. ASAM OSI also defines interfaces for traffic participant models. The TrafficCommand interface makes it possible to send commands to traffic participant models. The TrafficUpdate interface makes it possible to receive the updated state from traffic participant models [16].

3.3.2 AD function

This section describes automated driving function subsystem.

3.3.2.1 Main function

AD functions utilize a range of sensors and other input data about vehicle's surroundings to enhance safety and comfort for drivers, passengers, and other road users. By effectively processing the input data, AD functions control the vehicle's response to achieve desired outcomes. Advanced sensor technologies, including cameras, lidar, radar, and ultrasonic sensors, provide crucial information about the vehicle's surroundings. Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication systems further enrich the input data.

Sophisticated algorithms and artificial intelligence techniques analyse the sensor data in real-time, enabling the AD system to accurately perceive the environment, predict the behaviour of other road users, and make informed decisions.

It should be noted that the subsystem does not necessarily simulate the full AD stack. Depending on the requirements and objectives of the test, a suitable abstraction level for the AD function must be chosen.

3.3.2.2 Composition

There are no standardized definitions on what the composition of those functions must look like, but the industry has gradually come to a common split between different components within AD function and those are:

Perception:

The main purpose of this system is to perceive and understand the surrounding environment accurately and comprehensively. It uses various sensors to gather, process and create a detailed representation of the environment and vehicle's surroundings. Depending on the functional requirements, intended behaviour and sensor fidelity, perception system must have sufficient capabilities to process the input data. Perception's objectives can be split in the following categories based on their objectives:

- *Object Detection, Recognition and Classification*

  Perception must detect, recognize, and classify relevant objects from the data that is coming from various sensors and information sources. During the process of detection and recognition, the system must have a sufficient level of system robustness, which is usually achieved by various algorithms and procedures for noise filtering, error/false detection handling. Finally, a detected object must be classified to provide more information on the object itself which plays a significant impact on sensitivity calculations for the planning and control subsystem. Sensitivity in this context means a different classification of an object would impact later decision-making priorities (e.g., object classified as "pedestrian" could be taken with higher priority than an object classified as "animal", or different mitigation actions could be implemented). Additionally, depending on the sensor and perception fidelity, object detection probability varies significantly.

  Low-fidelity sensor models already provide ideal object detection and classification, while high-fidelity physical-based models provide raw data that needs to be fully processed to get the object lists. This processing for advanced perception systems usually involves advanced AI/ML algorithms to effectively process the raw data.
- *Environmental Mapping*

  A detailed map of the environment by integrating data from various sensors is created here. This map includes road structure, traffic signs, traffic lights and many other features of a specific environment. Developed algorithms can also base their mapping on already available, pre-developed high-fidelity map data to map the actual road more accurately in real-time.

- *Localisation and Positioning*
  The Localisation and Positioning function uses the precise location data from sensors to determine the exact vehicle position and orientation within the environment map.
- *Sensor Data Fusion*
  Sensor fusion takes the data of multiple sensors and creates a unified and accurate representation of the environment including static and dynamic objects. It uses various algorithms and filters to fuse the data.
- *Motion Tracking and Trajectory Estimation*
  This function uses various algorithms to analyse motion of all relevant detected objects and estimates their intended trajectories and dynamic properties.

Planning:

Processed data coming out of perception system is fed to the planning system. The main purpose of this block is to utilize the perceived data to make specific decisions that will ultimately be used to control the vehicle’s behaviour. Additionally, planning subsystem must be able to meet the target decisions within the complete ODD environment and intended scenario use case, considering vehicle dynamics properties. The output from this block will be the planned trajectory of the subject vehicle as an input to the control and act block.

Control and Act:

Decisions derived in planning are then fed to Control and Act block with the main purpose to control and execute those decisions through various actuation systems within a vehicle. Decisions that are transformed to actions in actual vehicle systems are continuously controlled within this component and additionally throughout constant influx of new perceived data that may result in newly updated decisions that have to be applied. The control system generates specific commands for steering, throttle, braking and other actuation systems to achieve the desired trajectory and behaviour. These commands are translated into physical actions by the car's drive-by-wire system or other control mechanisms. The actuators respond to the commands, adjusting the vehicle's motion accordingly.

3.3.2.3 Interfaces

Interfaces are defined between each block, based on AD function and its target capabilities, sensor fidelity and various vehicle actuation systems that is simulated by the vehicle dynamics block. Advanced AD functions require more comprehensive data on the input side from “sensing” part which usually includes high-fidelity sensor models, additionally, it will still require certain actuation systems to be able to control the vehicle as illustrated in Figure 5.

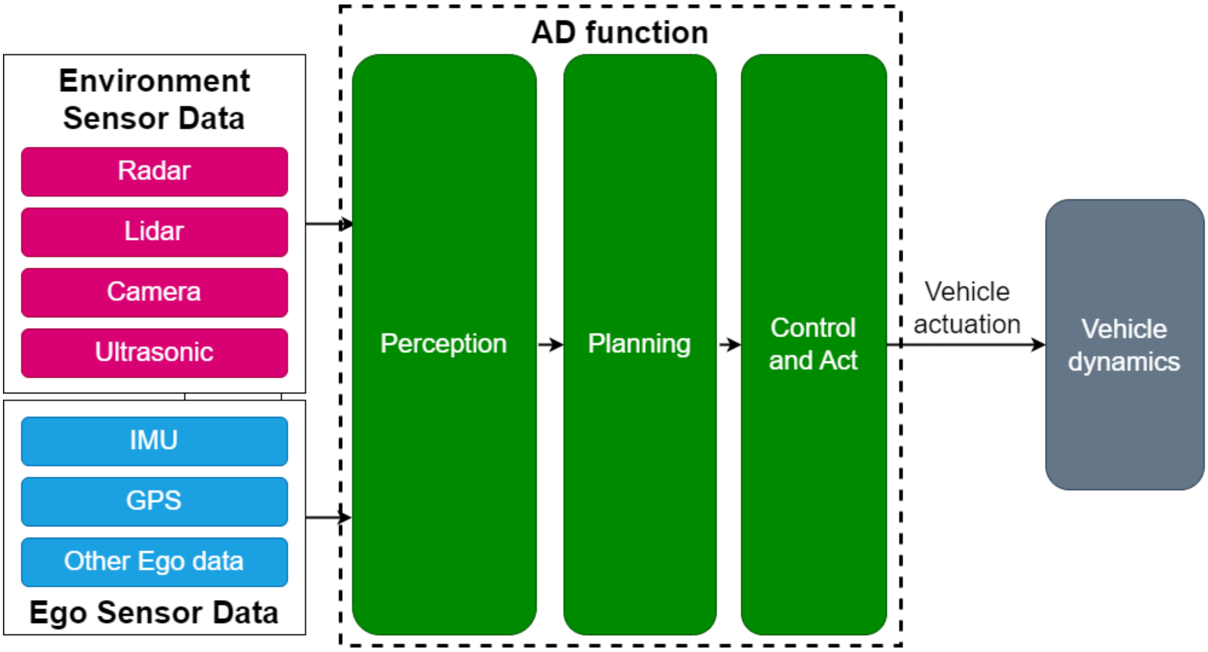

Figure 5. Sense-Plan-Act data flow of AD function block

3.3.3 Vehicle Dynamics

This section describes vehicle dynamics subsystem.

3.3.3.1 Main function

Vehicle dynamics describe the motion of a vehicle based on specific inputs (e.g., external and internal forces). The simulation of vehicle dynamics has a wide array of applications, ranging from the development of vehicle technologies (e.g., active suspensions, driver assistance systems) towards the usage for the validation of ADS. The respective simulation environments and mathematical foundations are considered to be well-established [17].

3.3.3.2 Composition

In general, vehicle dynamics includes certain aspects like tire and brake dynamics, engine and powertrain management, aerodynamics, suspension and steering, and vehicle modelling. To what extent these components are modelled differs greatly based on the respective application. Different vehicle dynamics models exist, which can be categorized by their respective model fidelity (low, medium, and high). Such fidelity categories are also part of recent regulations regarding such model classifications [18]. These different model fidelities represent the usual trade-off in modelling between realism (including as many of the mentioned components with dedicated models) and simplicity.

Low-fidelity:

The models in this category are point-mass and kinematic models. They degenerate the vehicle dynamics down to one point of mass (representing the vehicle's mass) or only consider particular kinematics of the vehicle motion (e.g., in the longitudinal and/or lateral domain). These models are applied if detailed modelling is not required, and fast calculation and easy parametrization are at the forefront.

Medium-fidelity:

The models in this category are primarily single-track and double-track models. Such models can be used to calculate, assuming a rigid two-axle vehicle body, the vehicle's longitudinal, lateral, and yaw motion. These models account for the body mass and the aerodynamic drag between the axles induced by acceleration and steering. In the case of the single-track model, the front and rear axle's tires are respectively reduced to one, located at the lateral centre of the vehicle. Therefore, no lateral load transfer can be considered, and the tire forces act along the vehicle's centre line. For the double-track model, no such simplification is performed. Therefore, the forces act at the vehicle's corners (assumed tire/road contact point). Both models are suitable if the vehicle's pitch, roll, and vertical motion are insignificant. The respective tire forces can be considered via an external tire model or simplified (either linear or non-linear, depending on the exact application) [19].

Tire modelling: To accurately capture the vehicle stability characteristics, it is necessary to model the vehicle's tires separately. Considering the tire contact point, these models should incorporate the interaction between the tire and the road surface. They should also consider the saturation characteristics of slip and camber angles while respecting load variation. However, simplified tire models overlook essential aspects such as roll and deflection steer.

Among the commonly utilized models for tire dynamics is an empirical model known as the Magic Formula. This model utilizes curve fitting techniques based on measurement data to represent tire behaviour accurately. It is widely used due to its effectiveness [20]. However, one of the most challenging aspects of tire modelling is capturing the correct non-linearities of the tire forces. These forces can vary significantly between different types of tires and substantially impact the overall dynamics of the vehicle.

High-fidelity:

This category contains models with the highest fidelity where the vehicle is modelled as a multi-body system characterized by different individual bodies. The overall motion of the multi-body system is described using differential and algebraic equations [21].

Generally, the vehicle is separated into different bodies:

- The vehicle body (chassis), where the mass or other loads are applied. Furthermore, the aerodynamic forces/torques are considered to be applied to the vehicle body.
- The four-wheel carriers (one at each wheel location): There, the gravity of the wheel carrier is applied, in addition to the tire forces/torques, the brake moment and the gyroscopic moment. The position of these wheel carriers is also used to model the exact tire contact point.
- Furthermore, the steering rack (and its position) influence the elasto-kinematics and the kinematic constraints.

Based on the steering and gas/brake pedal input, the model calculates the steering rack position and updates the vehicle kinematics (using the respective equations). Based on that, the respective forces and torques are applied (e.g., gravity, wind, aerodynamic drag) and calculated (e.g., tire forces – see detailed description in the medium fidelity section) at the individual bodies of the multi-body system.

Such models have the highest modelling detail and the highest number of parameters.

The model fidelity to choose for the vehicle dynamics in ADS validation depends on the ADS-equipped vehicle's respective ODD and behaviour competencies. The expected motion of the subject vehicle strongly influences the model choice and is an immediate requirement (for the model).

3.3.3.3 Interfaces

Regarding the subsystem vehicle dynamics interface, it needs to be distinguished between online (during scenario execution) and offline (before and after scenario execution).

Online:

In general, during the execution of such models, the actuation is the input to the model. The exact inputs differ based on the used model. The input for high-fidelity multi-body system models is mostly steering and gas/brake pedal commands. However, for simple models, the requested change in acceleration and steering could be used as input. The output is the updated vehicle's pose after each calculation step.

Offline:

Next to the interfaces for the model execution (input and output), other interfaces need to be considered. Based on the used models, additional parameters like the road network and road surface (e.g., in the ASAM standards OpenDRIVE® and OpenCRG® format) and all the parameters which are required for the respective model should be considered.

3.4 Traffic agents

The traffic agents' subsystem simulates the behaviour of various types of traffic agents, i.e., all dynamic elements except the subject vehicle. Traffic agents include all living beings, transport systems for living beings and goods, and moving objects on roads. [16].

3.4.1 Main function

Modern microscopic simulation environments such as VTD, SUMO or CARLA feature simulation of individual traffic agents. Commonly, the simulated traffic behaviour in such simulation environments can be set up on a macroscopic level before the simulation starts. Simulations differ in functionality to control the behaviour of multiple or individual road users at runtime, as they are tailored to different use cases.

External manipulation of traffic behaviour is typically facilitated by proprietary command protocols or APIs that are designed for the specific simulation environment and lack compatibility with other traffic simulations. Recent developments towards a modular simulation environment have resulted in the open standard ASAM OSI [16], which provides an interface for including traffic agents (also denoted traffic participants in ASAM OSI) models in microscopic simulations. This protocol not only promotes interchangeability, but also allows the development and use of traffic participant models independent of the chosen traffic simulation.

3.4.2 Composition

The behavior of traffic participants can be encapsulated in behavioral models. Following the ASAM OSI structure as shown in Figure 6 [16], an exemplary composition of a traffic agents simulation is presented.

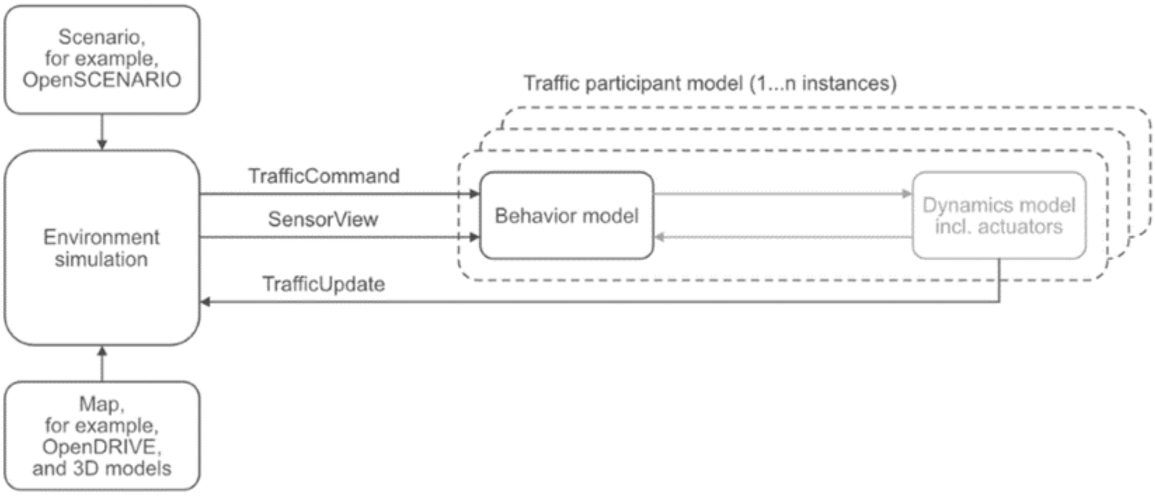

Figure 6. Possible setup of a traffic agent in a simulation architecture [16].

All traffic participant models together can be considered as a traffic system. Alternatively, it is reasonable to extend the presented system by a component that acts as an interface between the traffic agents and the rest of the subsystems. Such an interface, together with the traffic participant models, could be referred to as the traffic simulation.

Traffic commands are defined to direct traffic participant models within the traffic simulation. Traffic participant models can interpret and execute traffic participant commands with respect to the ground truth. The simulation-relevant state of the traffic participant, such as its position and rotation, is communicated to the simulation environment.

The behaviour of traffic participants can be specified using traffic commands of different levels of detail:

- High Level: The behaviour of traffic participants can be formulated in a statistical way as traffic densities. These types of definitions are open for interpretation and may yield different behaviour depending on the execution by the traffic simulation.
- Medium Level: The behaviour of road users can be controlled by traffic commands to request manoeuvres. These commands may request actions, that must be interpreted by the behaviour model. Example actions are acquiring position and change lane.
- Low Level: In a scenario-based workflow, the behaviour of individual road users can be scripted precisely on a low level. For example, events and conditions can be defined to trigger prescribed movement along a trajectory defined within the scenario.

Depending on the level of the traffic description (see 3.4.1), behaviour models of varying degree of freedom of their actions are required. The less explicit the traffic commands for the traffic participant are, the higher are the requirements for the traffic participant model to display realistic behaviour within the given amount of freedom.

To implement high- or medium-level traffic commands, advanced models are required to describe the behaviour of a human driver, an AV, or VRUs. The models can be parameterized according to the requirements of the particular test, for example to represent aggressive driving behaviour or to characterize driving errors.

Low-level commands, such as driving along a trajectory, can be interpreted by simple models. Movement along the trajectory can be replayed by interpolating the trajectory, or the trajectory can be interpreted by controller-driven tracking models.

If high demands are placed on the quality of the traffic participant model, it is possible to integrate models developed for subject vehicle, such as the perception and vehicle model (see Section 3.3), into the traffic participant models.

3.4.3 Interfaces

Input**:** The traffic agent subsystem receives the scenario to be played from the test manager. Depending on the scenario, behaviour and motion models are selected and initialized with the model parameters required for the scenario. The scenario also defines the overall test procedure so that traffic agents can be commanded accordingly.

The traffic simulation requires cyclic updates about the simulation ground truth, consisting of the drivable surface (road network), reference lines, walkable surface, dynamic traffic rules and all dynamic states of objects not computed within the traffic simulation. Since the behaviour of the subject vehicle is computed outside the traffic participant subsystem, the state of the subject vehicle is also forwarded to the traffic participant submodule via the environment subsystem.

V2X communication messages that are sent out by the subject vehicle to the V2X subsystem, may be communicated to the traffic simulation through specified communication protocols such as ETSI C-V2X.

Output**:** A traffic simulation provides information about all traffic participants according to the required fidelity. The traffic participant simulation runs in a discretized time domain and provides updates to the environment subsystem on a cyclic basis. The information can be condensed to object lists, that encompass the traffic participants position, orientation, and bounding box. To provide more accurate information about the behaviour of traffic participants, the data could be enriched by details such as individual body part positions for VRUs or blinker state for vehicles.

According to the simulation requirements of V2X communication the traffic simulation can communicate through specified communication protocols such as ETSI C-V2X.

3.5 Connectivity

In the simulation framework, connectivity subsystem implements V2X (Vehicle-to-Everything) communication that enables vehicles to establish communication links with other vehicles, pedestrians, cyclists, infrastructure elements, and network services in their surroundings [22–25]. This technology is essential for CCAM and brings various advantages, such as improved safety, efficiency, and mobility on the roads [26]. V2X communication encompasses V2V (Vehicle-to-Vehicle), V2I (Vehicle-to-Infrastructure), V2P (Vehicle-to-Pedestrian), and V2N (Vehicle-to-Network) communication types [3].

V2X communication employs technologies like DSRC and C-V2X [27, 28]. DSRC enables direct, low-latency vehicle-to-infrastructure communication, while C-V2X utilizes cellular networks (4G, 5G) and satellite connectivity for flexible communication between vehicles, pedestrians, cyclists, and network services when direct V2V communication is not possible.

3.5.1 Main function

The following outlines key services or functionalities that as of today is foreseen to be implemented in the connectivity subsystem [22–24]. Other services should be possible to add in the future as they become available.

Cooperative Awareness: Through standardized cooperative awareness messages, the connectivity subsystem enhances simulation dynamics. By enabling vehicles to effectively share vital data – speed, acceleration, position, heading, dimensions – this service bolsters the execution of Advanced Driving (AD) functions.

Cooperative Perception: Extending vehicle perception beyond individual sensors, the connectivity subsystem broadens cognitive horizons through exchange of perception data among the vehicles and the infrastructure in the simulation. Perception information exchange encompasses intricate road details, traffic dynamics, and potential hazards, enriching the sensory input available to AD function's perception module.

Cooperative Planning and Control: Facilitating harmony in AD function planning and execution, the connectivity subsystem al-lows to simulate collaborative optimization of trajectories, lane changes, and merging manoeuvres. Intentions exchange through dedicated V2X connectivity messages.

Intersection Management: The connectivity subsystem enables simulations of impactful vehicle-infrastructure interaction within intersections. This interaction encompasses vital elements such as traffic lights and road signs. For intersection-focused simulations, the subsystem equips vehicles to ex-change nuanced data with these infrastructure components, enabling simulation of responsive AD functions.

Vulnerable Road User Protection: Prioritizing safety, the connectivity subsystem enable simulates scenarios safeguarding pedestrians and cyclists. Functioning as a conduit, it facilitates direct interaction between vehicles and these individuals through smartphones or designated devices. Real-time alerts about vulnerable road users enable the simulation of safety-centric AD functions.

3.5.2 Composition

Within the simulation framework, the connectivity subsystem comprises of the following important components [24]:

1. *Scenario Module:* This module creates simulation scenarios based on inputs from the Environment Subsystem. The Scenario Module generates scenarios for connectivity simulation, which may involve utilizing open street maps to construct SUMO scenarios.

2. *On-Board Units (OBUs):* OBUs components within the V2X connectivity simulation framework serve as communication gateways. OBUs assume the crucial role of transmitting and receiving V2X messages. Each vehicle, represented as a Traffic Agent in the simulation, is equipped with an OBU, thereby enabling connectivity.

3. *Roadside Units (RSUs):* These are infrastructure elements strategically positioned throughout the simulation scenario to facilitate effective communication between vehicles and the surrounding infrastructure. Environment elements, such as traffic lights and road signs, are equipped with RSUs, reinforcing their role in the interconnected communication network.

4. *Communication Network:* This module simulates the intricate web of connections between OBUs and RSUs attributed to Traffic Agents. It operates by harnessing technologies such as Dedicated Short-Range Communication (DSRC) and Cellular-Vehicle-to-Everything (C-V2X), which enable direct Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communication.

3.5.3 Interfaces

In the simulation framework, the connectivity subsystem interfaces with three distinct and essential domains: the Traffic Agents Subsystem, the Environment Subsystem, and the System Under Test Subsystem.

1. *Traffic Agents Subsystem Interface:* The connectivity subsystem interfaces directly with the Traffic Agents Subsystem. Traffic agents equipped with On-Board Units (OBUs), interact with the connectivity subsystem to transmit and receive V2X messages. These messages flow through the connectivity subsystem, fostering cooperative awareness, perception, and coordinated planning.

2. *Environment Subsystem Interface:* Interfacing with the Environment Subsystem, the connectivity subsystem engages with critical scenario information. The Environment Subsystem holds essential details about the simulated scenario, including road layouts, building configurations, traffic lights, and traffic signs. These elements of the environment subsystem can be equipped with Roadside Units (RSUs) enabling connectivity.

3. *System Under Test (SUT) Subsystem Interface:* The connectivity subsystem's interface with the System Under Test (SUT) Subsystem facilitating Advanced Driving (AD) functions pertaining to the specific use case. The SUT Subsystem simulates these AD functions, and it relies on the connectivity subsystem for timely and relevant information. The

exchange of data between these subsystems enables the simulation of AD functions that react and adapt based on V2X communication.

3.6 Simulation model validation

3.6.1 Main function

As depicted in Figure 2, the subsystems further include the “Simulation model validation”. For verification and validation, it is necessary to not only use trustworthy simulation models but also to approve their quality and correlation to reality (see, e.g., the UNECE/VMAD and the discussions on credible simulation framework [29]). This is especially important for the certification of ADAS/AD functions in multiple vehicle variants, supplemented by simulation. For example, validated suspension models will affect the virtual sensor output, such as radar, lidar or camera, in a realistic manner, including e.g., pitching, and rolling motion of the chassis. Accurate tire models will result in realistic tire-surface interaction, especially on rough, non-even surfaces. Usually, the simulation quality and correlation with real-world physical tests are assessed on three levels: 1) vehicle dynamics behavior, 2) sensor and perception behavior, and 3) ADAS/AD system behavior, i.e., the controller output to the actuators.

3.6.2 Composition

The proposed subsystem includes a decision-making function that feeds back the results of the correlation analysis and decides whether and which additional physical tests or simulations are needed to improve the quality. As soon as every KPI meets a certain quality threshold, e.g., 95% accuracy, the simulation can be upscaled to complete the remaining tests and vehicle variants, without the need for any more physical tests. For the example of an AEB function verification, the simulation variants can include different weight distributions, types of tires (e.g., summer, winter), surface friction etc. The resulting test report, a so-called model quality matrix, can be used to provide proof of simulation accuracy.

The simulation model quality will be an essential part of SUNRISE Task 4.5, where the overall simulation toolchain will be validated.

3.6.2.1 Interfaces

Input: Test cases, test results from simulations, test results from physical tests.

Output: Simulation model validation test report.

4 SUBSYSTEM REQUIREMENTS

An essential aspect for virtual validation is the validity of the simulations in relation to the real world. The previous two chapters have described the SUNRISE simulation framework and the included subsystems. This chapter will discuss the subsystem requirements from a SAF perspective by analysing tools requirements, interface requirements, and model fidelity requirements.

4.1 Tools requirements

The V&V simulation framework requires simulation tools for all the subsystems. The previous chapter described the individual subsystems. From a simulation SAF perspective, the following more general requirements should be considered to support the testing and analysis processes:

- **Test Automation Tools** are used to assist in automating the execution of simulation tests, including test case generation, test execution, and result analysis. Test automation has the purpose of helping the verification and validation process by reducing manual effort and improving test coverage required to guarantee the system quality. In the simulation framework, the test case manager subsystem is main responsible for that.
- **Modelling and Simulation Tools** must satisfy the requirement related to the capability of creating realistic virtual models of automotive systems, including vehicle dynamics, powertrain, control systems, and environmental conditions. To guarantee the real-time requirement, specific real-time simulation platforms must execute and validate automotive simulation models in real-time. These platforms often include hardware and software components to ensure accurate and when relevant deterministic simulation results. Specifically, Software-in-the-Loop (SIL) and Hardware-in-the-Loop (HIL) tools should be used to enable the integration of simulation models with real or virtual hardware components, such as electronic control units (ECUs) and sensors, to perform system-level testing and validation. Lastly, Data Logging and Analysis Tools should be utilized to capture and record simulation data during test execution, enabling post-processing and visualization of the logged data to identify issues, validate system behaviour, and assess performance. As these are divided into several separate subsystems, co-simulation capabilities will be needed to connect the simulations between different subsystems.

Besides the tools for modelling and simulation of the subsystems in the simulation framework, tools are also needed to support compliance with standards to qualify the simulation framework for safety assurance [10, 30]:

- **Requirements Management Tools:** They must facilitate the capture, tracking, and traceability of ADS' requirements throughout the development lifecycle. These tools must ensure that simulation models versions align with the specified requirements and related tests for each test case.
- **Data Management and Analysis:** Virtual V&V generate a significant amount of data that needs to be efficiently managed and analysed. The simulation tools should provide capabilities for data storage, retrieval, and post-processing. Advanced data visualization,

statistical analysis, and reporting features are necessary to extract meaningful insights from simulation results and support decision-making.

- **Version Control, Configuration Management and Reporting Tools:** Version control and configuration management tools should be utilized to help, manage, and track changes to simulation models, test cases, and related artifacts, ensuring the requirements related to proper versioning, collaboration, and traceability. In addition, Reporting Tools may assist in generating technical reports, test reports, and documentation for satisfying requirements related to compliance and regulatory purposes.

4.2 Interface requirements

The interfaces between the SAF and the simulation framework as well as the interfaces between the subsystems are illustrated in Figure 2. Input and output data flow to and from the simulation framework are described in Sec. 2.2.1, and the input and output interfaces to respectively subsystem are included in the subsystems descriptions in Sec. 3.

To have a versatile definition of subsystems and the simulation framework, the task members chose to specify the type of data and signals to be exchanged rather than requiring a specific standard. Use of open standards for the interfaces are strongly encouraged. In the cases a standard is mentioned it shall be seen as an example and not a requirement. The actual choices of interfaces are left to subsequent tasks designing the simulation framework. It is also expected that new suitable standards will be developed in the future.

4.3 Model fidelity requirements

Model fidelity refers to the level of accuracy and representation to which the virtual models used in simulations and analyses reproduces the state and behaviour of a real-world object, feature, or condition. It is a critical aspect as the fidelity of the models directly affects the reliability and confidence in the V&V results.

The virtual models should be at the appropriate fidelity levels based on testing objectives. The fidelity of the models should be validated and calibrated against experimental data or known benchmarks to ensure accuracy and reliability. Model fidelity could include realistic environment characterization to simulate real-world environmental conditions that ADSs encounter. Achieving high model fidelity often comes with increased computational costs and time requirements. It's important to strike a balance between model fidelity and practical considerations such as computational resources, time constraints, and available validation data. Decision-making regarding fidelity levels should consider the specific objectives of the validation and verification activities.

Aspects of fidelity that must be considered, include:

- **Geometric fidelity** in virtual models refers to the accuracy and realism with which physical objects and their spatial relationships are represented in a virtual environment. When it comes to virtual models, geometric fidelity encompasses several aspects. Firstly, it involves the precise representation of the shape, dimensions, and proportions of objects. Virtual models should faithfully reproduce the geometry of physical entities, capturing their intricate details and surface characteristics. This level of accuracy is crucial for tasks such as visualization and virtual prototyping. It involves accurately positioning objects relative to one another and replicating their real-world alignments and orientations. This aspect is essential for simulating and analysing complex systems such as vehicles on a road, where the positioning and interaction of objects play a significant role. The virtual models should accurately represent the geometric features and dimensions of the physical components or systems being simulated. High geometric fidelity ensures that the simulations closely resemble the real-world counterparts and enables accurate analysis of spatial relationships, clearances, and interaction between components.

- **Material properties** in virtual models should reflect the actual physical properties of the materials used in automotive systems. Accurate material properties, such as stiffness, density, thermal conductivity, and damping, are crucial for predicting the behaviour and performance of vehicles operating in different conditions. For example, the reflectivity of the vehicle body shape from the radar sensor's point of view is crucial to simulate the real-world radio frequency signal propagation.
- **Sensor and actuator models** are crucial for capturing the interactions between the control system and the physical environment. The fidelity of these models affects the accuracy of system-level simulations, especially in scenarios involving ADAS, AD, or vehicle dynamics control. Realistic sensor and actuator models enable thorough testing and assessment of control algorithms and system responses. If the perspective is changed from sensors and actuators to vehicle systems or subsystems, an important aspect for virtual models should be the representation of the accurate underlying physics and dynamics of the automotive systems. This includes capturing the mechanical, thermal, electrical, and fluid dynamics behaviours relevant to the specific system being analysed. High-fidelity physics models enable accurate predictions of system responses, interactions, and performance. It is important to consider that each virtual model is part of the vehicle systems that are highly interconnected and often involve complex interactions between different subsystems. Virtual models should capture these interactions faithfully, accounting for coupling effects, feedback loops, and control strategies. Accurate representation of system interactions ensures comprehensive analysis and validation of the integrated system's behaviour.
- **Functional fidelity** refers to the degree to which the behaviour and response of the simulated subject vehicle, with its ADS feature, matches the behaviour and response of the real subject vehicle. Test scenarios, inputs, and boundary conditions should be used to evaluate that the functional performance of the system is properly simulated. Moreover, it should be ensured that the virtual model properly interacts with real-time inputs and outputs to allow for seamless integration with physical components and control systems.

Overall, ensuring high model fidelity in automotive virtual validation and verification is essential for generating reliable and actionable results, reducing the need for physical testing, and accelerating the development and optimization of automotive systems.

5 CONCLUSIONS

The presented list of relevant **subsystems** is a result of proposals from the task partners that have been merged and condensed into the presented non-exclusive list.

- *Test case manager*
  The test case manager subsystem (Sec. 3.1) interfaces the simulation framework with the rest of the SAF, orchestrates execution of test scenarios, performs SOTIF assessment, and computes KPIs and metrics from the executed test scenarios.
- *Environment*
  The environment subsystem (sec. 3.2) simulates the subject vehicle's ODD.
- *Subject vehicle*
  The subject vehicle subsystem (Sec. 3.3) simulates the behaviour of the subject vehicle. Included are three blocks:
  - *Sensors*
    The sensors block (Sec. 3.3.1) models the sensors required for the ADS to operate.
  - *AD function*
    The AD function block (Sec. 3.3.2) models the actual AD driving function and includes three subblocks:
    - Perception – out of sensor data the surrounding environment is accurately and comprehensively perceived and understood.
    - Planning – based on information received from the perception subblock tactical decisions and the trajectories for vehicle are calculated.
    - Control and Act – based on the output from the planning subblock, the decisions are transformed to actions controlling the vehicle.
  - *Vehicle dynamics*
    The vehicle dynamics block (Sec. 3.3.3) describes the motion based on the inputs.
- *Traffic agents*
  The traffic agents' subsystem (Sec. 3.4) simulates the behaviour of all dynamic elements except the subject vehicle. Traffic agents include all living beings, transport systems for living beings and goods, and moving objects.
- *Connectivity*
  The connectivity subsystem (Sec. 3.5) simulates wireless communication with elements surrounding the subject vehicle.
- *Simulation model validation*
  The simulation model validation subsystem (Sec. 3.6) is responsible for the validation of the simulation quality and correlation with real-world measurements.

The focus of this deliverable has been on pure virtual simulation, but the SAF will also cover XiL tests, where some of the listed subsystems can be replaced with the real components. It should be noted that, although many of the participants have experience in simulation tools and frameworks, the presented work is mainly theoretical. The actual development of the simulation framework is done in subsequent tasks of WP4.

Finally, an analysis is presented of the subsystem’s tools **requirements**, interface requirements, and model fidelity requirements from a SAF perspective.

- Tool requirements for the simulation framework (Sec. 4.1).
    - Test automation tools are needed. In SUNRISE simulation framework the test case manager is mainly responsibility for that.
    - Modelling and Simulation tools are needed to satisfy the requirement related to the capability of creating realistic virtual models of automotive systems. That is relevant for all subsystems with simulators. To guarantee the real-time requirement, simulation platforms must execute and validate models in real-time. In addition, as the simulation framework is divided into several separate subsystems, co-simulation capabilities will be needed to connect the simulations between different subsystems.
- Tools are also required to support compliance with standards to qualify the simulation framework for safety assurance (Sec. 4.1).
    - Requirements management tools
    - Data management and analysis tools
    - Version control, configuration management and reporting tools
- Interface requirements (Sec. 4.2).
    - For versatile definition of the subsystems and the simulation framework, the type of data and signals to be exchanged is defined, rather than imposing a specific standard interface. Use of open standards is encouraged.
- Fidelity requirements (Sec. 4.3).
    - To trust the simulation framework it is important to, based on test objective, show an appropriate degree of fidelity on both subsystem and simulation framework level. Aspects that must be considered include geometric fidelity, material properties fidelity, sensor and actuator models fidelity, and functional fidelity.

The analysis provides insight on what to include in the simulation framework and gives input to subsequent WP4 tasks of the SURISE project.

6 REFERENCES

1. SAE (2021) SAE J3016 Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. SAE International

2. ISO (2022) ISO 34501:2022 - Road vehicles -- Test scenarios for automated driving systems -- Vocabulary

3. ISO (2023) ISO 34503:2023 “Road Vehicles – Test scenarios for automated driving systems – Specification for operational design domain”

4. ISO (2022) ISO 34502:2022 - Road vehicles -- Test scenarios for automated driving systems -- Scenario based safety evaluation framework

5. IEC 60050 - International Electrotechnical Vocabulary - Welcome. https://www.electropedia.org/iev/iev.nsf/6d6bdd8667c378f7c12581fa003d80e7?OpenForm. Accessed 24 May 2023

6. HEADSTART project is on web. In: Headstart Project. https://www.headstart-project.eu/. Accessed 30 Jun 2023

7. HARMONISED EUROPEAN SOLUTIONS FOR TESTING AUTOMATED ROAD TRANSPORT | HEADSTART Project | Fact Sheet | H2020. In: CORDIS | European Commission. https://cordis.europa.eu/project/id/824309. Accessed 19 Jun 2023

8. Ilias Panagiotopoulos, Bernhard Hillbrand, Patrick Weissensteiner, et al (2023) SUNRISE D7.1 CCAM use cases validation requirements

9. Anders Thorsén, Daniel Becker, Ghada ben Nejma, et al (2023) SUNRISE D3.1 Report on baseline analysis of existing Methodology

10. ISO (2022) ISO 21448:2022 - Road vehicles -- Safety of the intended functionality

11. ASAM e.V. C_2023_01 ASAM OpenTestSpecification. https://www.asam.net/project-detail/c-2023-01-asam-opentest/#backToFilters. Accessed 22 Jun 2023

12. Muckenhuber S, Softic K, Fuchs A, et al (2021) Sensors for Automated Driving. In: Van Uytsel S, Vasconcellos Vargas D (eds) Autonomous Vehicles. Springer Nature Singapore, Singapore, pp 115–146

13. Muckenhuber S, Holzer H, Rubsam J, Stettinger G (2019) Object-based sensor model for virtual testing of ADAS/AD functions. In: 2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE). IEEE, Graz, Austria, pp 1–6

14. Schlager B, Muckenhuber S, Schmidt S, et al (2020) State-of-the-Art Sensor Models for Virtual Testing of Advanced Driver Assistance Systems/Autonomous Driving Functions. SAE Intl J CAV 3:233–261. https://doi.org/10.4271/12-03-03-0018

15. Magosi ZF, Li H, Rosenberger P, et al (2022) A Survey on Modelling of Automotive Radar Sensors for Virtual Test and Validation of Automated Driving. Sensors 22:5693. https://doi.org/10.3390/s22155693

16. ASAM e.V. (2022) ASAM OSI - Open Simulation Interface v3.5.0

17. Kutluay E, Winner H (2014) Validation of vehicle dynamics simulation models – a review. Vehicle System Dynamics 52:186–200. https://doi.org/10.1080/00423114.2013.868500

18. ISO (2022) ISO/DIS11010-1 “Passenger Cars—Simulation Model Classification—Part 1: Vehicle Dynamics”

19. Schramm D, Hiller M, Bardini R (2014) Single Track Models. In: Vehicle Dynamics. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 223–253

20. Hans B. Pacejka (2012) Tire and Vehicle Dynamics. Elsevier

21. Dempsey M, Fish G, Delgado Beltran JG (2015) High Fidelity Multibody Vehicle Dynamics Models for Driver-in-the-Loop Simulators. pp 273–280

22. Alalewi A, Dayoub I, Cherkaoui S (2021) On 5G-V2X Use Cases and Enabling Technologies: A Comprehensive Survey. IEEE Access 9:107710–107737. https://doi.org/10.1109/ACCESS.2021.3100472

23. Garcia MHC, Molina-Galan A, Boban M, et al (2021) A Tutorial on 5G NR V2X Communications. IEEE Communications Surveys & Tutorials 23:1972–2026. https://doi.org/10.1109/COMST.2021.3057017

24. Singh PK, Nandi SK, Nandi S (2019) A tutorial survey on vehicular communication state of the art, and future research directions. Vehicular Communications 18:100164. https://doi.org/10.1016/j.vehcom.2019.100164

25. TransAID Project Transition Areas for Infrastructure-Assisted Driving | TransAID Project | Fact Sheet | H2020. In: CORDIS | European Commission. https://cordis.europa.eu/project/id/723390. Accessed 12 Jun 2023

26. Weber R, Misener J, Park V (2019) C-V2X - A Communication Technology for Cooperative, Connected and Automated Mobility. In: Mobile Communication - Technologies and Applications; 24. ITG-Symposium. pp 1–6

27. Kenney JB (2011) Dedicated Short-Range Communications (DSRC) Standards in the United States. Proceedings of the IEEE 99:1162–1182. https://doi.org/10.1109/JPROC.2011.2132790

28. Gyawali S, Xu S, Qian Y, Hu RQ (2021) Challenges and Solutions for Cellular Based V2X Communications. IEEE Communications Surveys & Tutorials 23:222–255. https://doi.org/10.1109/COMST.2020.3029723

29. ECE/TRANS/WP.29/2022/58 Secretariat (GRVA) New Assessment/Test Method for Automated Driving (NATM) Guidelines for Validating Automated Driving System (ADS) – amendments to ECE/TRANS/WP.29/2022/58 | UNECE. https://unece.org/transport/documents/2022/05/informal-documents/grva-new-assessmenttest-method-automated-driving. Accessed 15 Aug 2023

30. ISO (2018) ISO 26262:2018 Road Vehicles : Functional Safety. ISO

31. ASAM e.V (2022) ASAM Test Specification Study Group Report 2022 V1.0.0

ANNEX 1. TEST DATA ANALYSIS AND VALIDATION

This annex describes the test data analysis and validation part handled in the test assessment block that is part of the test case manager subsystem of the SUNRISE V&V simulation framework.

Main function

Test data plays a key role in the design, development and associated verification and validation activities of ADS. It needs to be planned and designed, modelled and stored before it can be properly used in the development process [31]. The more advanced the test data management strategy, the more efficient the testing of ADS will be. If ADS failures can be detected and identified early in the development process, they can be fixed more efficiently and effectively. In the context of ADS, high quality test data management, and hence data exchange and fusion, is key to facilitating interoperability between systems, manufacturers, engineering services and other stakeholders. In the following section, key issues in this area related to test data management, test evaluation and KPI assessment and reporting are described in more detail.

Composition

Three key elements within Test data analysis and validation namely test data management, test evaluation and KPI assessment and reporting are outlined in detail in the following paragraphs.

Test Data Management

The development of the ADS is highly data driven. At all stages of validation, huge amounts of pre-recorded test data, scenarios and test cases are required to investigate the behaviour of the ADS in different simulations and real-world tests related to the intended ODD where the ADS will be deployed. Therefore, vehicles and test setups need to collect huge amounts of test data while driving on real and virtual roads. Considering the importance of having test data and test descriptions available throughout the entire development, simulation and real-world testing process, it is clear that efficient scenario, test case and test data management is key to an efficient development and validation process [31].

[31] developed a high-level ADS-specific domain model as a basis for proper test data management, including important artefacts such as test inputs, test outputs, or information needed to support test preparation. These artefacts were grouped according to their relationship to each other. The approach distinguishes between information needed to (re-) perform a test run and information generated during or after a test run.

Test evaluation

On the one hand, test strategies are becoming more complex and holistic, and on the other hand, the data required to correctly evaluate the results of the test strategy are becoming more diverse [31]. As a result, type approval and homologation are also more complex and must be based on test or analysis results obtained from test data management and its proper implementation. The homologation of an ADS must demonstrate the safe operation of the targeted behavioural competencies within the specified target ODD. These criteria include a robust demonstration that the ADS can autonomously perform the driving task within the intended ODD, comply with traffic rules and regulations, and autonomously switch to minimum risk manoeuvres in the event of leaving the approved ODD. Test strategies are an important pillar needed to approve all this. At the same time, it is necessary to make them evaluable and assessable so that the technical services can check them. In this sense, consistent data management is mandatory.

KPI assessment and reporting

In order to evaluate the individual scenarios associated with specific test cases, several KPIs are evaluated to decide whether the scenarios have been successfully passed or the ADS under test has failed. In this aspect, different categories of KPIs are used to evaluate the scenarios associated with specific test cases. Typically, KPI categories include operational aspects such as compliance with traffic rules and general safety in terms of accident-free operation within the target ODD, regulatory aspects such as defined KPIs within regulations e.g., UN-ECE R 131, R 157 and finally development aspects related to the sense-plan-act principle of the ADS under test. It is expected to report KPI results related to deployment and regulatory categories for type approval, while development KPIs remain confidential within the OEM to protect its IP. The successfully passed scenarios and associated test cases form the basis for evaluating the test case coverage related aspects associated with the homologation of the ADS.

ANNEX 2. SOTIF ASSESSMENT BLOCK IN THE SIMULATION FRAMEWORK

Main function

This block (or sub-subsystem) is responsible for the identification of unknown-unsafe scenarios. Identification of unknown-unsafe scenarios is required as part of the SOTIF safety standard. The output unknown-unsafe scenarios (or a representative set) shall be used to assess the AV as part of the overall safety assessment process. Thus, the reported test results shall also include the test results for the set of unknown-unsafe scenarios. The identified unknown scenarios shall also be provided to the scenario database, thus expanding the scenario database and the number of known scenarios through simulation.

Composition

- Extracting search space from the ODD description and collected data in the ODD *(if collected data is available)*: extract possible actor behaviours, static environment, and dynamic conditions e.g., weather and illumination conditions.
- Optimization study to identify unknown-unsafe scenarios within the search space.
- Selection of a representative set of unknown-unsafe scenarios.

Interfaces

1. Input:
    a. *(initial)* ODD description, which includes a road network e.g., in an OpenDRIVE file.
    b. *(initial)* collected data in the ODD (if available)
    c. *(during optimization)* simulation data (outputs) for concrete test cases
2. Output:
    a. (during optimization) concrete test cases
    b. *(final)* a (representative set of) unknown-unsafe scenarios for AV assessment
    c. *(final)* identified unknown scenarios provided to the scenario database.

Important notes:

- The test cases which are generated and executed **during optimization process** and their corresponding test results are only temporary; these should be discarded once SOTIF assessment is completed and are not part of the output of the simulation framework.
- Calculation of metrics for objective function in the optimization process (from simulation data) is done within the SOTIF assessment block. These metrics may be specific to the SOTIF assessment.