# D2.2

## Interactive web-based handbook for SUNRISE safety assurance framework

**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** | WP2: CCAM safety assurance framework |
| **Deliverable number** | D2.2: Interactive web-based handbook for SUNRISE safety assurance framework |
| **Deliverable responsible** | John-Fredrik Ehrenhofer Grönvall, Chalmers Industriteknik |
| **Status - Version** | Final – V1.0 |
| **Submission date** | 14/02/2025 |
| **Keywords** | SUNRISE, SAF, Safety Assurance Framework, CCAM, Handbook |

**Authors**



| Role | Name |
|---|---|
| **Main author** | John-Fredrik Ehrenhofer Grönvall (Chalmers Industriteknik) |
| **Contributing authors** | Hafdis Jonsdottir (Chalmers Industriteknik)<br>Jason Zhang (University of Warwick)<br>Stefan de Vries (IDIADA) |



**Quality Control**

| | Name | Organisation | Date |
|---|---|---|---|
| Peer review 1 | José Rodríguez | ERTICO | 03/02/2025 |
| Peer review 2 | Stefan de Vries | IDIADA | 07/02/2025 |

**Version history**

| Version | Date | Author | Summary of changes |
|---|---|---|---|
| 0.1 | 31/01/2025 | Hafdis Jonsdottir,<br>John-Fredrik Ehrenhofer Grönvall,<br>Jason Zhang | First draft of document structure + chapter 1 introduction, 2 approach of 3 thematic areas |
| 0.2 | 04/02/2025 | Hafdis Jonsdottir | Draft version for review. |
| 0.3 | 12/02/2025 | Hafdis Jonsdottir | Update after internal review |

| | | | |
|---|---|---|---|
| 0.4 | 13/02/2025 | Hafdis Jonsdottir | Update after second internal review |
| 1.0 | 14/02/2025 | John-Fredrik Ehrenhofer Grönvall | Final version for publication |

**Legal disclaimer**

# TABLE OF CONTENTS

# LIST OF FIGURES

# ABBREVIATIONS AND ACRONYMS

| Abbreviation | Meaning |
|---|---|
| AD | Automated Driving |
| ADS | Automated Driving System |
| AEB | Autonomous Emergency Braking |
| CCAM | Connected, Cooperative, and Automated Mobility |
| COTSATO | COncretizing Test Scenarios and Associating Test Objectives |
| ISMR | In-Service Monitoring and Reporting |
| KPI | Key Performance Indicator |
| NATM | New Assessment/Test Method for Automated Driving |
| NCAP | New Car Assessment Program |
| ODD | Operational Design Domain |
| OEM | Original Equipment Manufacturer |
| PDF | Probability Density Function |
| SAF | Safety Assurance Framework |
| SCDB | SCenario DataBase |
| SUNRISE | Safety assUraNce fRamework for connected, automated mobIlity SystEms |
| STPA | System-Theoretic Process Analysis |
| SUT | System Under Test |
| UC | Use Case |
| V&V | Verification and Validation |
| XiL | X-in-the-loop |

# EXECUTIVE SUMMARY

Safety assurance of Cooperative, Connected, and Automated Mobility (CCAM) systems is a crucial factor for their successful adoption in society, yet it remains a significant challenge. It is generally acknowledged that for higher levels of automation, the validation of these systems by conventional test methods would be infeasible. Furthermore, certification initiatives worldwide struggle to define a harmonized safety assurance approach enabling massive deployment of CCAM systems.

The **SUNRISE** project develops and demonstrates a **CCAM Safety Assurance Framework** (SAF). The overall objective of the SUNRISE project is to accelerate the large-scale and safe deployment of CCAM systems. In alignment with international twin projects and initiatives, the project aims to achieve this objective by providing a SAF consisting of three main components: a Method, a Toolchain and a Data Framework. The **Method** is established to support the SAF safety argumentation, and includes procedures for scenario selection, sub-space creation, dynamic allocation to test instances and a variety of metrics and rating procedures. The **Toolchain** contains a set of tools for safety assessment of CCAM systems, including approaches for virtual, hybrid and physical testing. The **Data Framework** provides online access, connection and harmonization of external Scenario Databases (SCDBs), allowing its users to perform query-based extraction of safety relevant scenarios, allocation of selected scenarios to a variety of test environments, and reception of the test results.

This deliverable D2.2, presents the first version (V1.0) of the Interactive web-based Handbook for SUNRISE Safety Assurance Framework. The first version 1.0 of the Handbook is available at the SUNRISE Web portal. The web-based tool allows engagement with a wide set of stakeholders on the SUNRISE safety assurance framework.

The direct link to the Handbook is: https://ccam-sunrise-project.eu/high-level-overview/
Sunrise Web page-> Tools -> Sunrise Handbook 1.0

The SAF Handbook can be observed as a simplified high-level summary of the SAF developed in task T2.2. Its purpose is to familiarize potential users with the SAF, to guide them through the framework, and to point them towards further details in corresponding project deliverables. To facilitate the understanding of SAF users, the Handbook will later be enriched with Use Case demonstration results obtained in Work Package 7 (WP7). The intended audience of the SAF Handbook includes a wide set of stakeholders involved in CCAM safety assurance. Including vehicle developers, certifiers and regulators, tool developers, researchers and academics and possibly citizens.

It is important to understand that this deliverable aligns with the 1st version (V1.0) of the SAF Handbook. During the remaining course of the SUNRISE project, and aligned with important milestones, several updates of the online SAF Handbook are planned, until end of the project in August 2025.

# 1 INTRODUCTION

## 1.1 Project introduction

Safety assurance of Connected, Cooperative, and Automated Mobility (CCAM) systems is a crucial factor for their successful adoption in society, yet it remains a significant challenge. CCAM systems need to demonstrate reliability in all driving scenarios, requiring robust safety argumentation. It is acknowledged that for higher levels of automation, 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 bearing more significant weight for cost efficiency reasons.

Worldwide, several initiatives have started to develop test and assessment methods for Automated Driving (AD) functions. These initiatives already transitioned 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 systems' testing and quality assurance. Nevertheless, the lack of a common European validation framework and homogeneity regarding validation procedures to ensure safety of these complex systems, hampers the safe and large-scale deployment of CCAM solutions. In this landscape, the role of standards is paramount in establishing common ground and providing technical guidance. However, standardising the entire 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 (SCDBs) are another issue tackled by several initiatives and projects, that generally tends to silo solutions. A clear concrete approach should be used (at least at European level), dealing with scenarios of any 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. These must be set 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 and taking other project initiatives as a baseline, it becomes necessary to move to the next level in the development and demonstration of a commonly accepted **Safety Assurance Framework** (**SAF**) for the safety validation of CCAM systems, including a broad portfolio of Use Cases (UCs) and comprehensive test and validation tools. This will be done in **SUNRISE**, which stands 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 product of the SUNRISE project. As the following figure indicates, it takes a central role, fulfilling the needs of different automotive stakeholders that all have their own interests in using it.

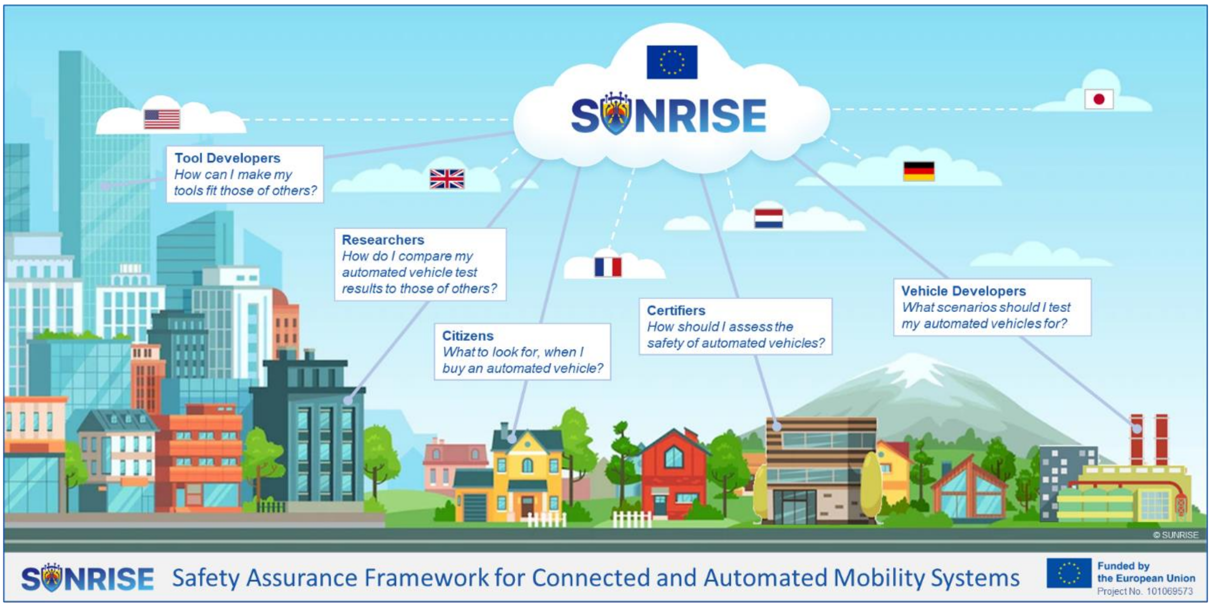


Figure 1: Safety Assurance Framework stakeholders

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, specifically 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 objective by providing a SAF consisting of three main components: a Method, a Toolchain and a Data Framework. The **Method** is established to support the SAF safety argumentation, and includes procedures for scenario selection, sub-space creation, dynamic allocation to test instances and a variety of metrics and rating procedures. The **Toolchain** contains a set of tools for safety assessment of CCAM systems, including approaches for virtual, hybrid and physical testing. The **Data Framework** provides online access, connection and harmonization of external Scenario Databases (SCDBs), allowing its users to perform query-based extraction of safety relevant scenarios, allocation of selected scenarios to a variety of test environments, and generation of the test results. The SAF will be put to the test by a series of **Use Cases demonstrations**, designed to identify and solve possible errors, gaps and improvements to the underlying methods, tools and data.

Following a common approach will be crucial for present and future activities regarding the testing and validation of CCAM systems, allowing to obtain results in a standardised way, to improve analysis and comparability, hence maximising the societal impact of the introduction of CCAM systems.

The following figure shows the general workplan of the SUNRISE project.

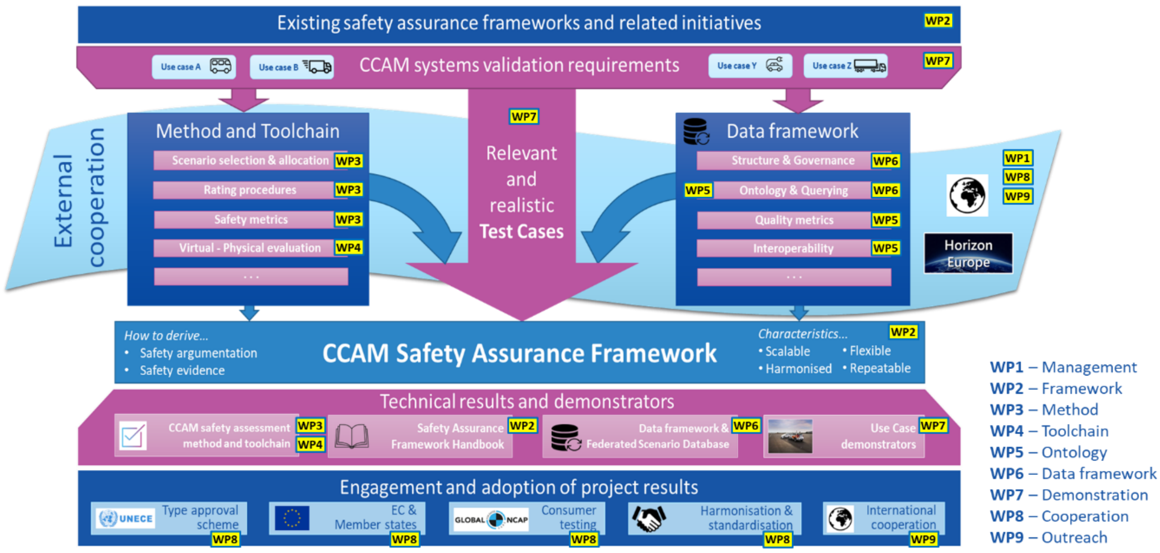


Figure 2: Workplan of the SUNRISE Project

## 1.2 Purpose of deliverable

This deliverable provides an introduction to 1st version of the Handbook as an interactive tool to familiarize with the Safety Assurance Framework (SAF). This tool allows for engagement with a wide set of stakeholders on the SUNRISE SAF.

The SAF Handbook can be observed as a simplified high-level summary of the SAF developed in task T2.2. Its purpose is to familiarize potential users with the SAF, to guide them through the framework, and to point them towards further details in corresponding project deliverables. To facilitate the understanding of SAF users, the handbook will later be enriched with Use Case demonstration results obtained in Work Package 7 (WP7).

It is important to understand that this deliverable aligns with the 1st version (V1.0) of the SAF Handbook. During the remaining course of the SUNRISE project and aligned with important milestones, several updates of the online SAF Handbook are planned. Handbook (V2.0) is expected to be launched on the 10th of March 2025. Handbook (v3.0) is expected to be launched on the 1st of June 2023. The final update of the Handbook (V4.0) will be published on the 30th of August.

## 1.3 Intended audience

The intended audience of the SAF Handbook includes a wide set of stakeholders involved in CCAM safety assurance. The intended audience is categorized into primary and secondary groups, based on their level of involvement in CCAM safety assurance.

**Primary audience** are those who directly engage with or depend on safety assurance processes in their professional activities and include vehicle developers, certifiers and regulators, and tool developers. The primary audience will be able to use the Handbook as a guidance for the organisations internal process for CCAM safety assurance.

**Secondary audience** are individuals or groups who indirectly benefit from or are influenced by the outcomes of safety assurance practices. These include researchers and academics and possibly citizens. The secondary audience will benefit from the Handbook by gaining a clearer understanding of the Safety Assurance process and how it applies to their individual usage and activities in the field of CCAM safety assurance.

Besides the SAF Handbook audiences mentioned above, specific audiences of this deliverable include both internal *and* external stakeholders of the SUNRISE project. The *internal* stakeholders (SUNRISE consortium members) are expected to provide feedback on the SAF Handbook explained in this deliverable. For that internal feedback channels have been established, including a dedicated spreadsheet and several project meetings. The *external* stakeholders (potential SAF users) are encouraged to provide feedback on the SAF Handbook through a dedicated form integrated into the online SAF Handbook.

## 1.4 Deliverable structure and relation to other parts of project

The contents of this deliverable are divided in the following chapters:

**Chapter 2:** High-Level Overview. Provides a broad introduction to the structure of the Handbook.

**Chapter 3:** Inputs to the SAF. Describes the input into Safety Assurance Framework (SAF)

**Chapter 4:** Data Framework. Introduces the SUNRISE Data Framework.

**Chapter 5:** Performance Assurance. Explains the Safety Performance Assurance block.

**Chapter 6:** Audit. Provides a stepwise approach for auditing the application of the SUNRISE SAF.

**Chapter 7:** Case Studies. Describes the four use case categories.

**Chapter 8:** Submit Feedback. Explains the process for users to provide feedback on the Handbook v1.0

**Chapter 9:** Conclusions. Summary of the main output of the deliverable.

This deliverable is based on the online SAF Handbook, published on the SUNRISE website. The contents of the SAF Handbook are based on the present draft SAF developed until 1 February 2025 in Task 2.2. The SAF will be updated during the remaining period of the project. In parallel to that, the handbook will be updated in new versions, until end of the project, when the final version of the Handbook will be published.

# 2 HIGH-LEVEL OVERVIEW

This chapter describes the landing page of the Handbook, see Figure 3. The direct link to the Interactive web-based Handbook is: https://ccam-sunrise-project.eu/high-level-overview/

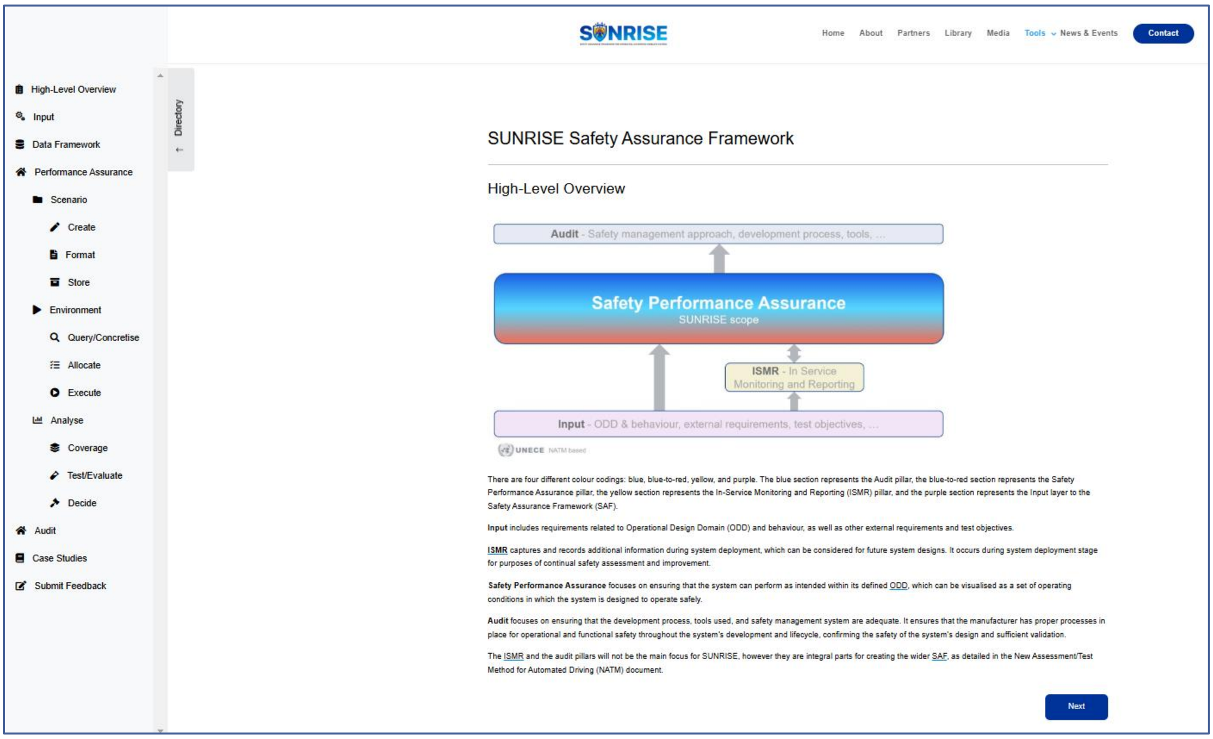


*Figure 3: Screenshot of the landing page in the Handbook v1.0*

Figure 4 is an enlarged picture showing the pillars of the UNECE NATM document.


*Figure 4: Pillars of the UNECE NATM document.*

As shown in Figure 4 above, there are four different colour codings: blue, blue-to-red, yellow, and purple. The blue section represents the Audit pillar, the blue-to-red section represents the Safety Performance Assurance pillar, the yellow section represents the In-Service Monitoring and Reporting (ISMR) pillar, and the purple section represents the Input layer to the Safety Assurance Framework (SAF).

**Input** includes requirements related to Operational Design Domain (ODD) and behavior, as well as other external requirements and test objectives.

**ISMR** captures and records additional information during system deployment, which can be considered for future system designs. It occurs during system deployment stage for purposes of continual safety assessment and improvement.

**Safety Performance Assurance** focuses on ensuring that the system can perform as intended within its defined ODD, which can be visualised as a set of operating conditions in which the system is designed to operate safely.

**Audit** focuses on ensuring that the development process, tools used, and safety management system are adequate. It ensures that the manufacturer has proper processes in place for operational and functional safety throughout the system's development and lifecycle, confirming the safety of the system's design and sufficient validation.

The ISMR and the audit pillars will not be the main focus for SUNRISE, however they are integral parts for creating the wider SAF, as detailed in the New Assessment/Test Method for Automated Driving (NATM) document.

# 3 INPUTS TO THE SAF

This chapter describes the Inputs to the SAF. The direct link to the Inputs page of the Interactive web-based Handbook is: [https://ccam-sunrise-project.eu/input/](https://ccam-sunrise-project.eu/input/)

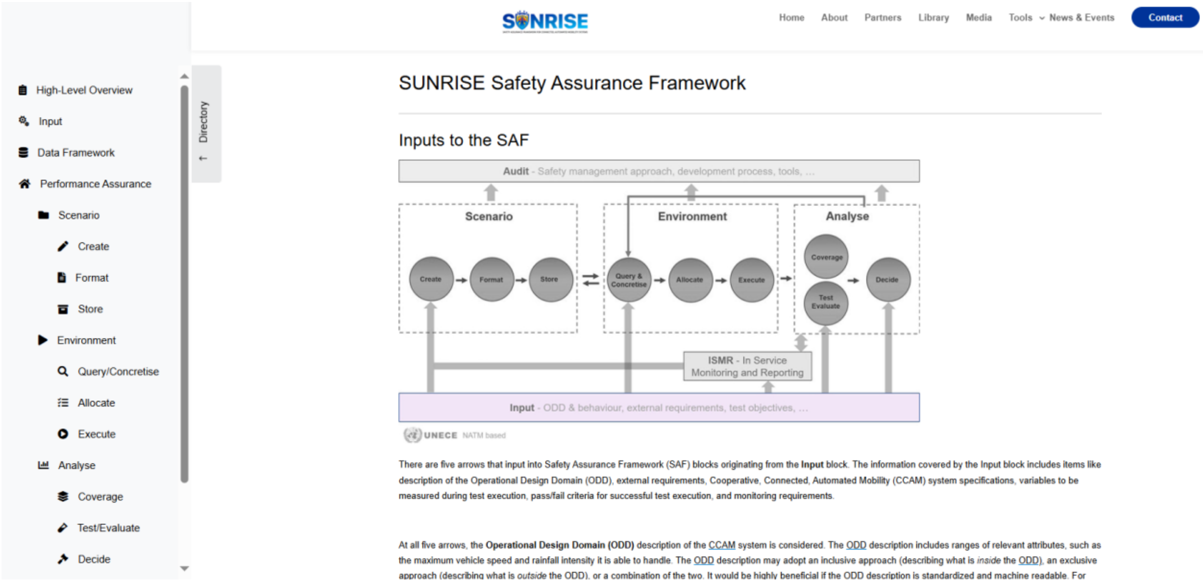


*Figure 5: Screenshot of the Input to the SAF page in the Handbook v1.0*

Figure 6 is an enlarged picture showing the Input to the Safety Assurance Framework (SAF) blocks.

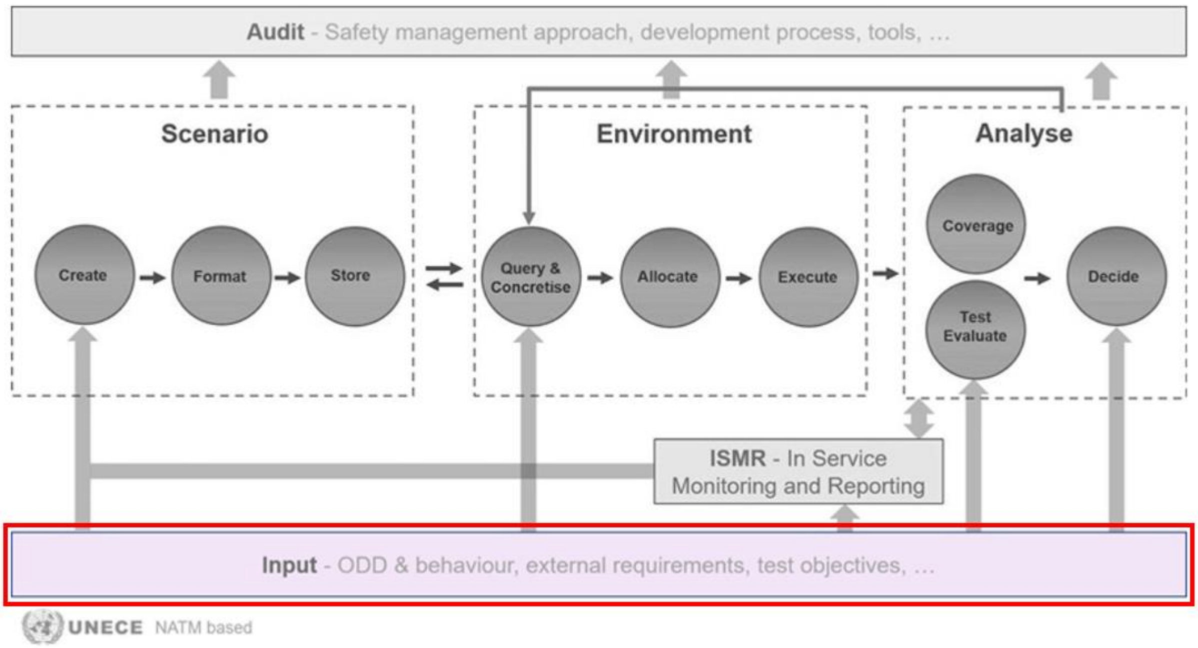


*Figure 6 : Inputs to the SAF*

There are five arrows that input into Safety Assurance Framework (SAF) blocks originating from the **Input** block. The information covered by the Input block includes items like description of the Operational Design Domain (ODD), external requirements, Cooperative, Connected, Automated Mobility (CCAM) system specifications, variables to be measured during test execution, pass/fail criteria for successful test execution, and monitoring requirements.

At all five arrows, the **Operational Design Domain (ODD)** description of the CCAM system is considered. The ODD description includes ranges of relevant attributes, such as the maximum vehicle speed and rainfall intensity it is able to handle. The ODD description may adopt an inclusive approach (describing what is inside the ODD), an exclusive approach (describing what is outside the ODD), or a combination of the two. It would be highly beneficial if the ODD description is standardized and machine readable. For formatting the ODD, it is suggested to follow the norms related to the ODD definition format listed in ISO 34503 (2023).

The Create block can utilize the ODD description to create scenarios that are part of the described ODD.

The Query & Concretize block uses the ODD description to generate the test cases that are needed for the safety assurance of the CCAM system, including the description of the needed output to analyse the results.

The Test Evaluate block uses the ODD description to determine whether the CCAM system operated safely within its ODD in a specific test.

The Coverage block uses the ODD description to check whether the ODD space is sufficiently covered.

The ISMR block (in-service monitoring and reporting) employs the ODD description in order to verify whether the system is operating inside its ODD.

Besides the ODD description, various blocks also need **the external requirements** applying to the CCAM system. These requirements should reflect the required behavioural competences, regulations, rules of the road, safety objectives, standards and best practices. The requirements can be a source for creating scenarios, which is why the requirements are part of the first interface. Furthermore, it is important that the Query & Concretize block considers the requirements and outputs relevant test cases, as the goal of the SAF is to assure that the requirements are met.

Note that this process also establishes the means to measure compliance with the requirements for the tests cases. Not all requirements can be formulated using test validation criteria, which is why the requirements can also be communicated to the Test Evaluate block.

Lastly, the input to the ISMR block needs requirements to check whether system-level requirements are satisfied over the lifetime of the system. Note that requirements can be very different from system to system, so formalizing this might be challenging. For that reason, a standardized description format would be preferred.

The **CCAM system** is the main subject of the test cases, thus its technical specifications need to be provided to the Query & Concretize block. The CCAM system can also be a source for creating scenarios. For example, scenarios created using knowledge of the system architecture and fault analysis techniques such as systems-theoretic process analysis (also known as STPA). The CCAM system can be a physical prototype or virtual model of the actual system (or even a combination of both).

In case some variables need to be measured during the test execution – additional to the variables that are needed to verify the test objectives – this information can be provided to the Query & Concretize block.

Additionally, pass/fail criteria for successful test execution have to be provided to the Query & Concretize block, in order to identify when test cases have or have not successfully been executed. For example, if there are certain tolerances on speed values or lateral path deviations, these shall be included. When the test case is executed outside of the pass/fail criteria, then its execution would have to be deemed as unsuccessful from an execution point of view. These examples are related to both real world and test track test allocation, but other examples may apply to virtual test environment (for example when simulation output contains an error).

For all inputs, it is assumed that simulation models (other than those representing the CCAM system) and the simulation platform are part of the "Execute" component and, therefore, not provided externally. Hence, simulation models (other than that of the CCAM system) and the simulation platforms are not part of the listed interfaces.

# 4 DATA FRAMEWORK

This chapter describes the Data Framework. The direct link to the Data Framework page of the Interactive web-based Handbook is: https://ccam-sunrise-project.eu/data-framework/

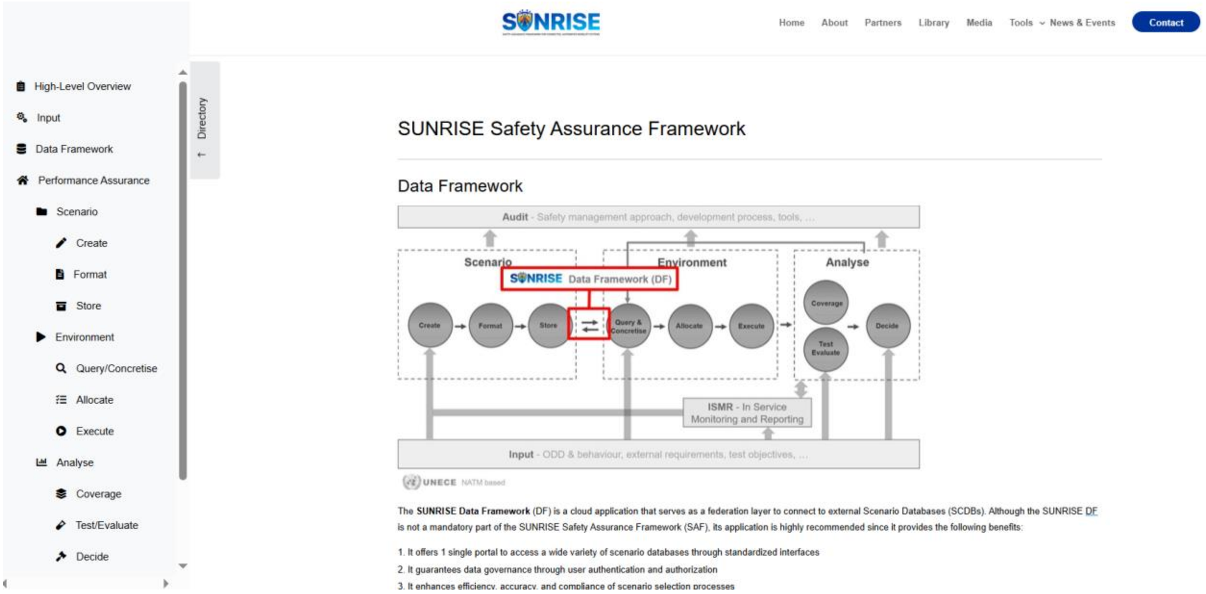


*Figure 7: Screenshot of the Data Framework page in the Handbook v1.0*

Figure 8 is an enlarged picture showing the position of the Sunrise Data Framework within the Safety Assurance Framework.

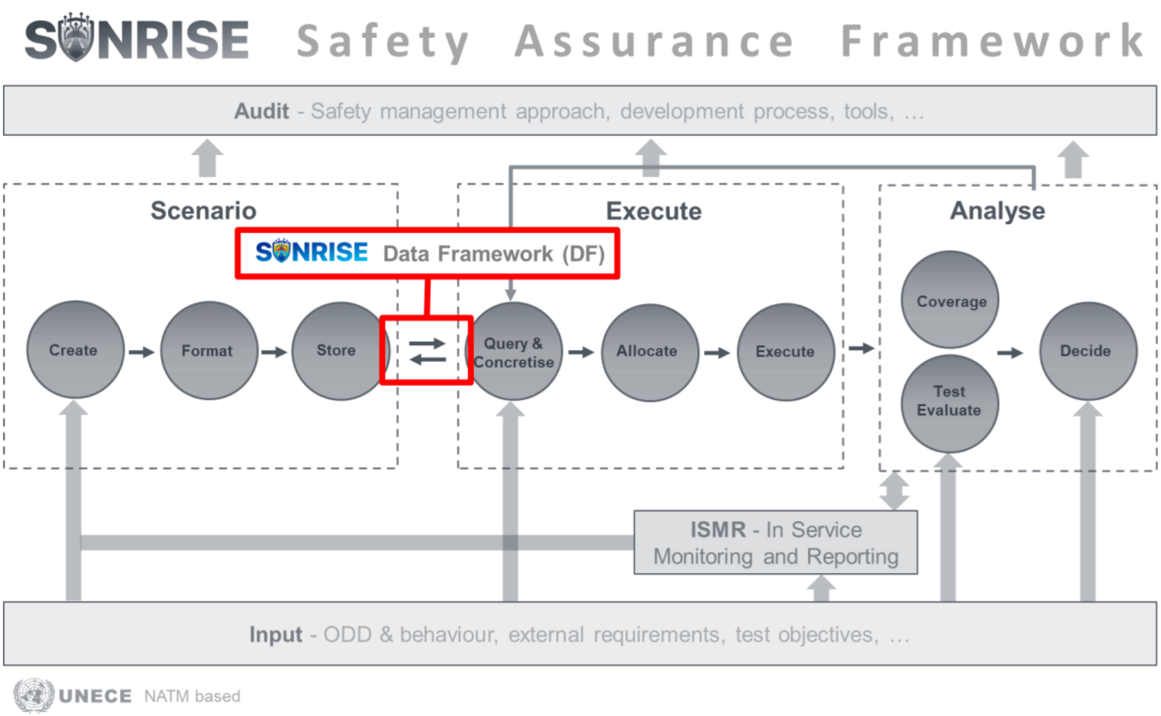


*Figure 8: Position of the Sunrise Data Framework within the Safety Assurance Framework*

The **SUNRISE Data Framework** (DF) is a cloud application that serves as a federation layer to connect to external Scenario Databases (SCDBs). Although the SUNRISE DF is not a mandatory part of the SUNRISE Safety Assurance Framework (SAF), its application is highly recommended since it provides the following benefits:

1. It offers 1 single portal to access a wide variety of scenario databases through standardized interfaces
2. It guarantees data governance through user authentication and authorization
3. It enhances efficiency, accuracy, and compliance of scenario selection processes
4. It centralizes scenarios for various stakeholders (like certifying entities, Cooperative, Connected Automated Mobility (CCAM) developers, test engineers)

The position of the SUNRISE DF within the SUNRISE SAF, is shown in the figure above. In this position, it can be observed as a tool to enhance overall CCAM validation efficiency through:

- Centralized scenarios from multiple databases through standardized interfaces
- Streamlined scenario selection enabled by advanced querying
- Improved compliance of selected scenarios with validation criteria
- Integration and compatibility with virtual, hybrid and physical test environments
- Sophisticated results tracking and feedback
- Future expansion of connected databases

# 5 PERFORMANCE ASSURANCE

This chapter describes the Performance Assurance. The direct link to the Performance Assurance page of the Interactive web-based Handbook is: [https://ccam-sunrise-project.eu/performance-assurance/](https://ccam-sunrise-project.eu/performance-assurance/)

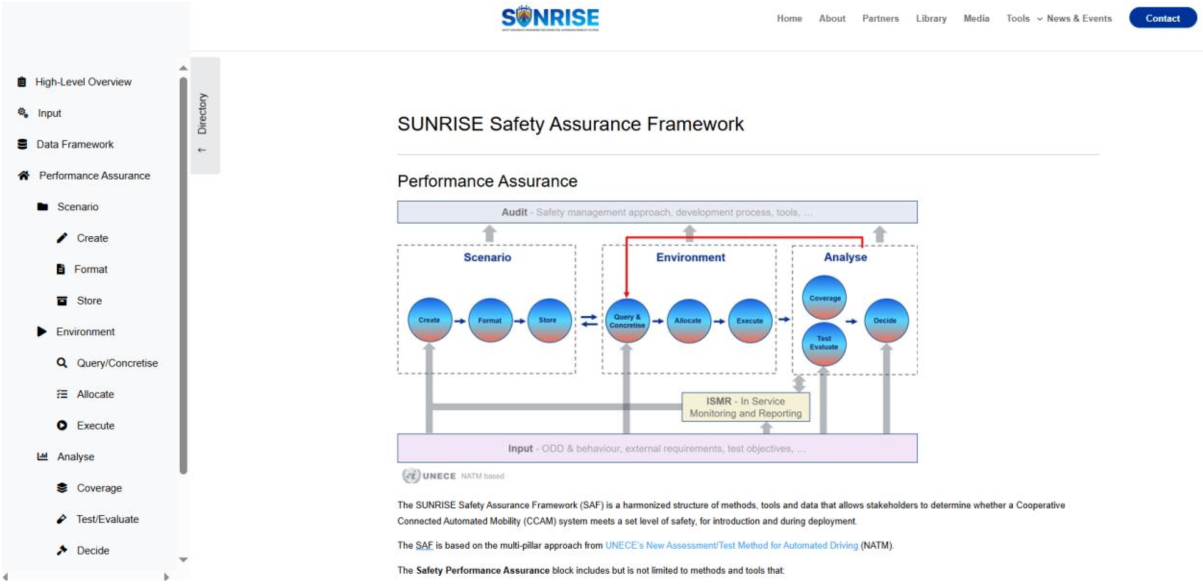


*Figure 9: Screenshot of the Performance Assurance page in the Handbook v1.0*

Figure 10 is an enlarged picture showing the SUNRISE SAF with expanded Performance Assurance pillar.

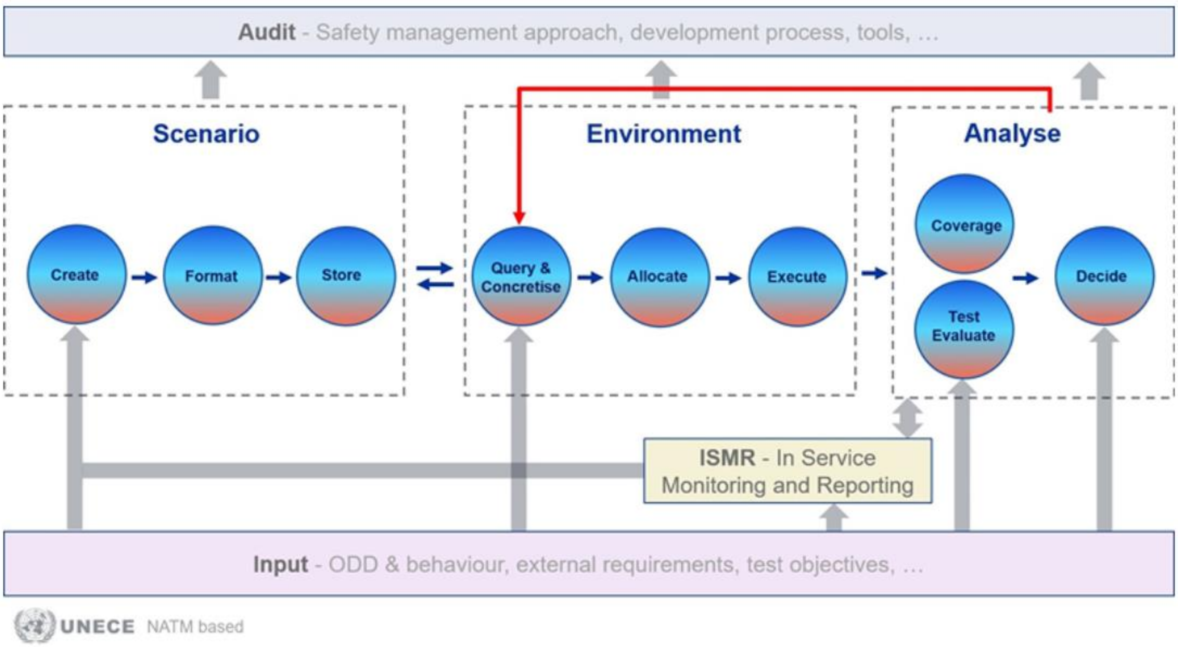


*Figure 10: SUNRISE SAF with expanded Performance Assurance pillar*

The SUNRISE Safety Assurance Framework (SAF) is a harmonized structure of methods, tools and data that allows stakeholders to determine whether a Cooperative Connected

Automated Mobility (CCAM) system meets a set level of safety, for introduction and during deployment.

The SAF is based on the multi-pillar approach from UNECE's New Assessment/Test Method for Automated Driving (NATM) [1].

The **Safety Performance Assurance** block includes but is not limited to methods and tools that:

- create, format, store and extract relevant scenarios
- allocate relevant scenarios to different test environments
- execute relevant scenarios in different test environments
- analyse and evaluate test results
- elaborate evidence-based safety argumentation and decisions

In the SUNRISE project, the focus is on Performance Assurance using a scenario-based approach and incorporating virtual, hybrid- and physical test environments. In addition, it features the SUNRISE Data Framework which allows users to obtain scenarios from multiple qualified scenario databases.

At a high level, the Safety Performance Assurance block comprises the Scenario, Environment, and Analyse sub-blocks. Essentially, this means that to conduct the safety assurance process, a set of scenarios is needed, an environment for executing these scenarios, and a process to analyse the execution results to determine whether the system is safe.

## 5.1 Scenario

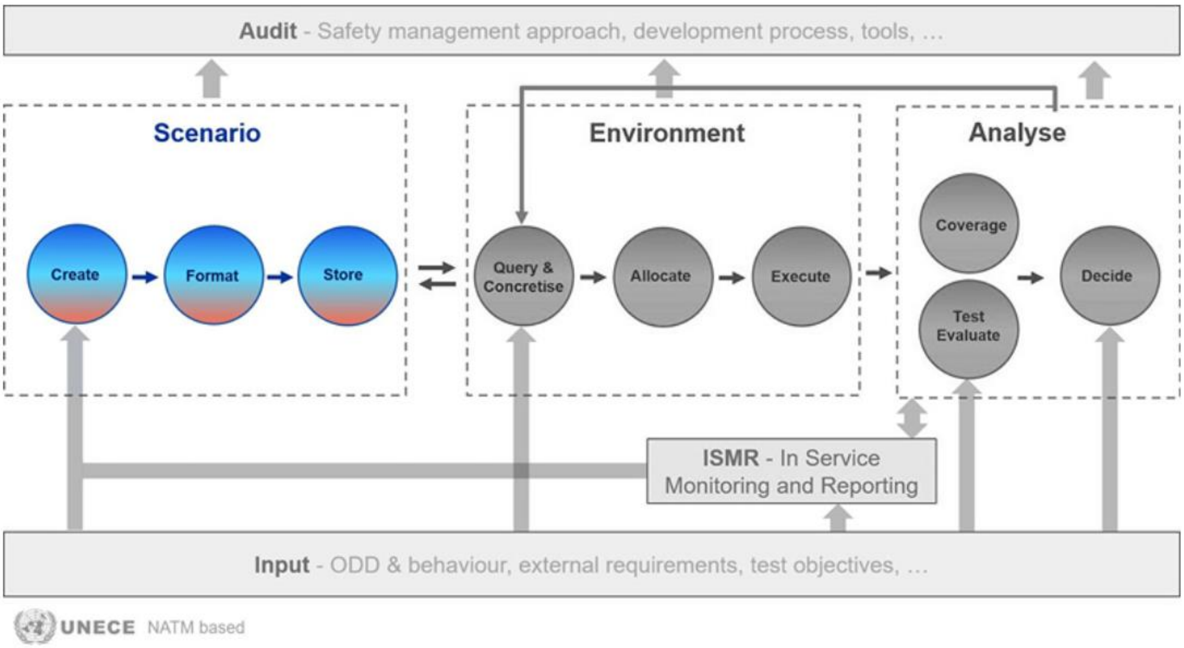


*Figure 11 : The Scenario block*

The **Scenario** block focuses on managing scenarios critical for safety assurance, encompassing three key processes: creation, formatting, and storage, see Figure 11.

1. **Create** involves generating scenarios based on data-driven or knowledge-driven approaches. Scenarios are derived using methodologies like StreetWise, which extracts real-world scenarios and statistics from driving data, or database like Safety Pool™, which leverage various scenario generation methods. These processes occur within individual scenario databases (SCDBs), reflecting database-specific requirements and use cases. While scenario creation is handled independently by SCDB providers, it is integral to the overarching safety assurance framework.
2. **Format** structures scenarios into appropriate representations for effective communication and downstream testing. Multiple levels of abstraction are employed to cater to diverse stakeholders, ranging from human-readable functional and abstract-level scenarios to machine-readable logical and concrete-level scenarios. Common formats like ASAM OpenSCENARIO, OpenDRIVE, and BSI Flex 1889 ensure interoperability and standardization.
3. **Store** consolidates formatted scenarios within SCDBs, enabling seamless access and integration. The SUNRISE project introduces the SUNRISE Data Framework to link multiple SCDBs, requiring database owners to adhere to shared formats and abstraction levels to connect. This federated approach ensures unified access while maintaining flexibility for individual database management.

Together, these processes enable a structured, standardized, and scalable approach to scenario management for CCAM system safety assurance.

### 5.1.1 Create

The Create block entails acquiring the necessary data and knowledge to create scenarios, while Format involves structuring the scenario using for example a scenario description language such as ASAM OpenSCENARIO XML [2] in combination with ASAM OpenDRIVE [3], or a schema based on the StreetWise domain model [4] or MetaScenario [5]. Finally, Store entails storing the formatted scenario in a searchable scenario database (SCDB). This component also includes the SUNRISE Data Framework for accessing individual scenario databases.

Please note that the processes described in the Create and Format blocks, take place within each SCDBs. Within the Store block, the SUNRISE Data Framework is introduced to link individual SCDBs into the SUNRISE Data Framework, providing that these SCDBs meet the requirements identified and developed in the SUNRISE project.

Since the scenario creation takes place in individual SCDBs, the information provided here is for reference. However, from safety assurance point of view, the scenario creation process is an integral part. As mentioned earlier, scenarios can be created using two different approaches: data-based, and knowledge-based [6], however this does not restrict the total number of created scenarios as this is specific to the databases and also the use cases.

Taking the Safety PoolTM scenario database [7] as an example, which University of Warwick has been developing and maintaining, it contains scenarios generated using eight different methods which belong to the two approaches. StreetWise [8] is a methodology developed by TNO to use driving data to extract real-world scenarios, determine the statistics of these scenarios, and use the scenarios for assessing automated driving systems. To support the research and development of StreetWise, it comes with a concept database that contains real-world scenarios. The EU ADS Act [9], FRAV VMAD [10], CertCAV [11] also mention scenario creation approaches from both data-based and knowledge-based; SAKURA [12] mainly uses data-based scenario creation approach; Streetwise [13] also incorporate real world data combined with sampling methods to create scenarios; VVM [14] documents a data based scenario generation approach from drone database.

### 5.1.2 Format

Once the scenarios are created, the next step is to **Format** them in a way that represents the scenario effectively. Please note that since this activity takes place at individual Scenario Databases (SCDBs), there is no specific requirements on the format at this stage (as such requirement will be incorporated at later processes). The following information is for references only. Given the multiple stakeholders involved in scenario-based testing, including regulators, research engineers, test engineers, system engineers, and the public, there is a need to use multiple levels of abstraction for scenario formatting. There are four levels of scenario abstraction: *functional level scenarios, abstract level scenarios, logical level scenarios,* and *concrete level scenarios*. Each level emphasises different properties; for example, functional and abstract levels may focus on human readability, while logical and concrete levels may focus on machine readability. The importance of using common formats is important, such as ASAM OpenSCENARIO and OpenDRIVE formats for logical and concrete representations, and BSI Flex 1889 for abstract-level representation.

### 5.1.3 Store

After scenarios are created and formatted, the next step is to **Store** them in a scenario database for downstream testing activities. The SUNRISE Safety Assurance Framework (SAF) proposes a common SUNRISE Data Framework (DF) for accessing multiple scenario databases. Requirements have been set out for these databases to connect to the SUNRISE DF. Scenario creation and formatting is the responsibility of individual database owners. However, it is also their responsibility to ensure that these requirements are met to connect their databases to the SUNRISE DF, such as using a common scenario format and incorporating multiple abstraction levels.

## 5.2 Environment

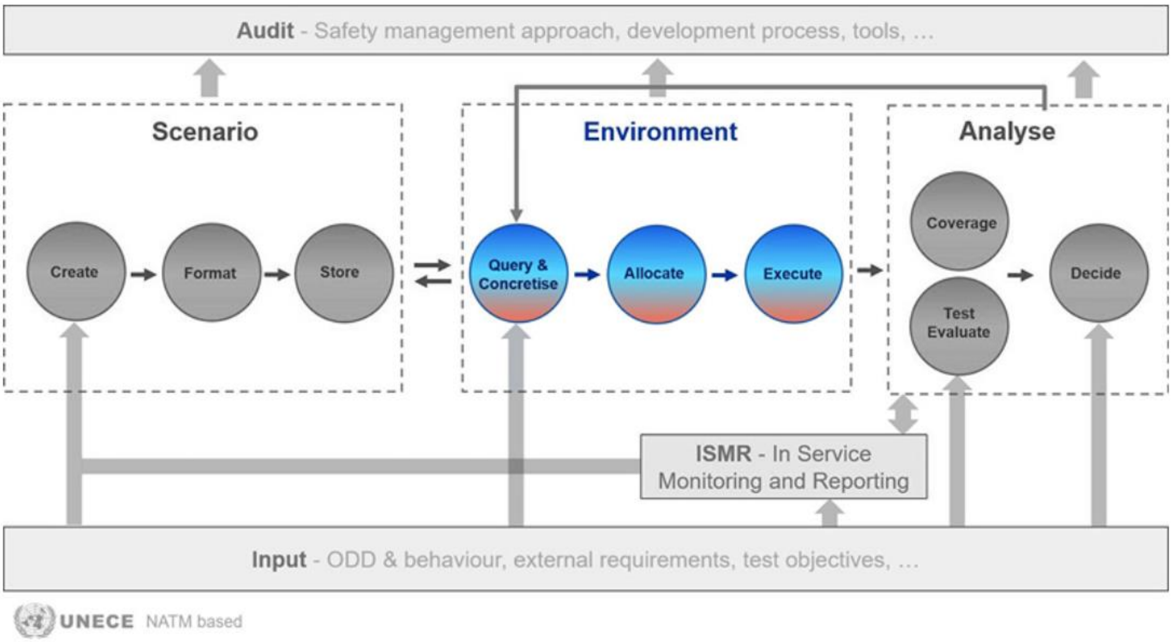


*Figure 12 : The Environment block*

The **Environment** block of the SUNRISE Safety Assurance Framework (SAF) operationalises test scenarios through three stages: querying and concretising, allocating test environments, and executing scenarios, see Figure 12.

1. **Query & Concretise**
   Scenarios are retrieved from the SUNRISE Data Framework (DF) using test objectives, Operational Design Domain (ODD), behaviour, and external requirements. Logical scenarios, defined by parameter ranges, are concretised into specific values to create concrete test scenarios. These are then combined with test objectives to make them test-ready. Queries follow the OpenLABEL format, ensuring consistency across databases. Sampling methodologies support exploring parameter spaces, estimating safety measures, or identifying failure points.
2. **Allocate**
   Test scenarios are matched to appropriate test environments, from virtual simulations to controlled real-world settings. Test case requirements, such as scenario details and pass/fail criteria, are compared against the capabilities of available test instances like Hardware-in-the-Loop or proving grounds. A virtual-first approach prioritises lower-fidelity simulations initially, with iterative reallocations to higher-fidelity environments as needed. External factors, such as safety overrides, are documented for assessment.
3. **Execute**
   Allocated test scenarios are executed in the designated environments, with relevant data recorded for analysis. Feedback loops enable refining scenarios and optimising test objectives.

This component ensures a structured, efficient, and adaptive process for scenario testing within the safety assurance framework.

### 5.2.1 Query/Concretise

The **Query & Concretise** block takes input from the Input block, which contains Operational Design Domain (ODD), behaviour, external requirements, and test objectives. It then passes these requirements to the SUNRISE Data Framework (DF) as a query and retrieves scenarios from the individual scenario databases. The scenarios returned from the SUNRISE DF to the Query & Concretise block could be either of logical or concrete scenario levels, meaning that all parameters are defined using value ranges, allowing for an infinite number of concrete scenarios to be derived from a single logical scenario. The next step is to concretise these parameter ranges into specific values and to combine these concrete scenarios with the test objectives. Once combined, the scenarios are allocated within an execution environment.

Text below provides a workflow of the *Query & Concretize* block.

1. Since the scenario creation and formatting occurs in individual Scenario Databases (SCDB), the Input block information containing the test objectives, ODD & behaviour requirements and external requirements, will be fed directly into the Query & Concretise block.
2. These external requirements will then be used to query into individual SCDBs via the SUNRISE federated layer, and logical or concrete scenarios will be then returned. It is at this step that scenarios hosted within SCDBs become test scenarios, because it is associated with the intended testing purposes upon the retrieval from the scenario databases.
3. If the returned scenarios are at the logical level (i.e., parameters are described in ranges)
    1. the first function of the block will create concrete scenario with concrete parameter values.
    2. the second function of the block will then combine the concrete test scenario with further test objectives.
4. If the returned scenarios are already concrete scenarios, then this block will skip the creation of the concrete scenario step, but combining it with test objectives.
5. The block will send the concrete test scenarios with their test objectives to the environment allocation block for test execution.
6. During or post execution, the Analyse block will then feedback the test outcome back to this block, the next set of concrete parameter combinations can be created. An example of such process could be incorporating an optimisation algorithm, with the intention of explore a testing objective, e.g., to explore the failure points within a logical scenario.

Logical or concrete scenarios can be retrieved from databases connected to the SUNRISE Data Framework through queries. These queries are constructed using tags recorded in the OpenLABEL format, which adheres to a harmonized ontology developed within the SUNRISE project. This approach ensures a unified understanding of all elements and their interrelationships across the connected databases.

To derive concrete scenarios from the logical ones for testing purposes, several sampling methodologies have been developed. These methodologies facilitate the discretization of the continuous parameter space and enable the selection of specific samples (concrete scenarios) within this space. These samples are chosen to estimate the distribution of a safety measure across the parameter space. Alternatively, these methodologies can be applied to optimize for other testing objectives, such as identifying the pass/fail boundary within the parameter space or identifying parameter subspaces.

### 5.2.2 Allocate

The SUNRISE Safety Assurance Framework is test environment-agnostic, allowing scenarios to be executed in a range of test environments, from fully virtual to hybrid environments (such as Hardware-in-the-Loop) to controlled physical environments (such as proving grounds). Within the **Allocate** block, a test environment allocation workflow is applied (as detailed below), which will be implemented in case studies within the broader Safety Assurance Framework. Once a scenario is allocated to a test environment (also called test instance), the scenario is executed, and the corresponding data is recorded.

The initial allocation process involves two key inputs: test case information and test instance capabilities. Test cases include scenario descriptions, expected behavior of the system under test (SUT), and pass/fail criteria. Specific requirements are extracted from these test cases. The second input consists of available test instances, such as virtual testing, X-in-the-Loop (XiL), and proving ground testing, with field testing excluded due to its uncontrollable nature.

The process begins with comparing test case requirements to the capabilities of test instances using a structured approach. This includes analyzing aspects like scenery elements, environmental conditions, dynamic elements, and test criteria. Once a suitable test instance is identified, the test case is allocated to it.

A virtual simulation-first approach is prioritized for safety and efficiency. Test cases suitable for virtual simulation are executed using the lowest-fidelity simulation capable of meeting requirements, to maximize throughput. After execution, results are reviewed to determine if further testing on higher-fidelity test instances is required. This iterative process may include reallocations to ensure the necessary test coverage and accuracy.

The process includes provisions for external influence, such as road authorities overriding allocations or proving ground operators refusing tests due to safety concerns. These decisions must be documented for the final assessment. Additionally, the Allocate block includes initial reallocation steps, resulting from the Analyse block.

### 5.2.3 Execute

In the **Execute** block, the test execution might happen in a virtual, hybrid or physical test environment, depending on the test instance resulting from the Allocate block. The allocated test cases form the input for this block. How the tests are carried out is not explicitly specified by the Safety Assurance Framework (SAF) and is the responsibility of the entity performing the tests. If the Allocate block has been applied correctly, it is guaranteed that the selected test instance is capable of performing the tests. In the case of virtual testing, the SAF recommends a harmonised approach. For that, the SUNRISE project developed a harmonised V&V simulation framework, which can be used for virtual validation of Cooperative Connected Automated Mobility (CCAM) systems but is not mandatory.

As shown in Figure 13, the **SUNRISE harmonised V&V simulation framework** consists of a so-called base layer consisting of 4 interconnected subsystems, namely:

- Subject Vehicle – Sensors (the sensors installed in the vehicle)
- Subject Vehicle – Automated Driving (AD) function (the behavioural competencies of the vehicle)
- Subject Vehicle – Vehicle Dynamics
- Environment (in which the vehicle operates)

In this approach, the base layer is the core element that can be harmonised, because these 4 subsystems are essential for all simulations. That is the reason why it is possible to use standardised interfaces between these subsystems. The framework can be extended by the user in 4 dedicated dimensions related to the target Operational Design Domain (ODD), the vehicle Sensor set-up, the Software architecture and the Hardware architecture.

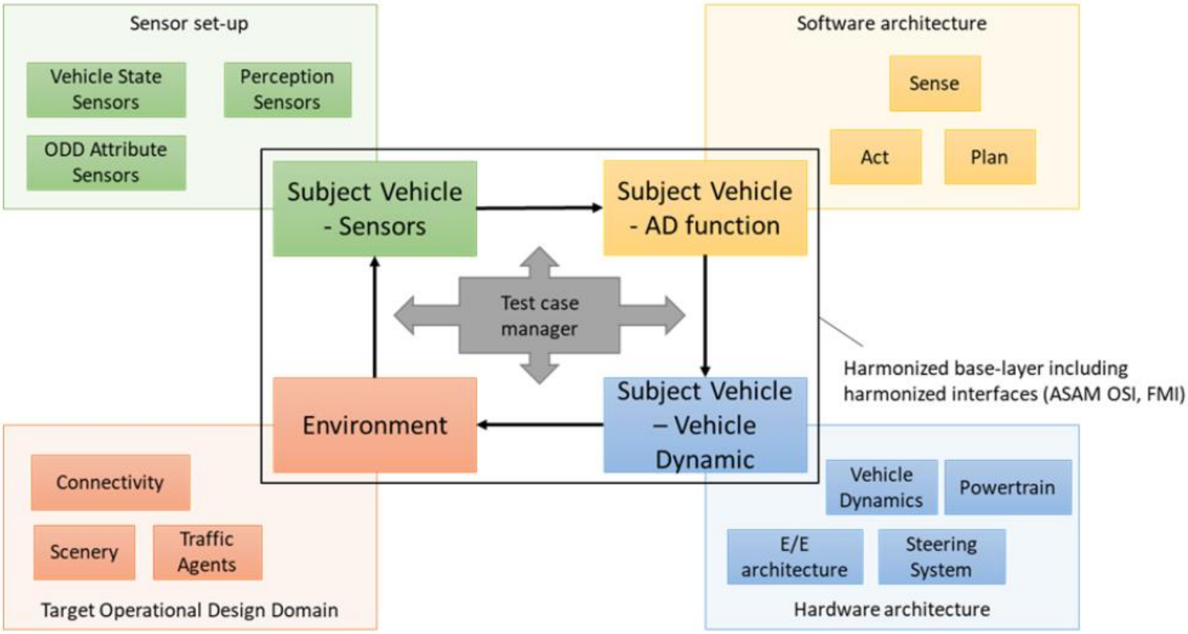


*Figure 13: the SUNRISE harmonised V&V simulation framework*

## 5.3 Analyse

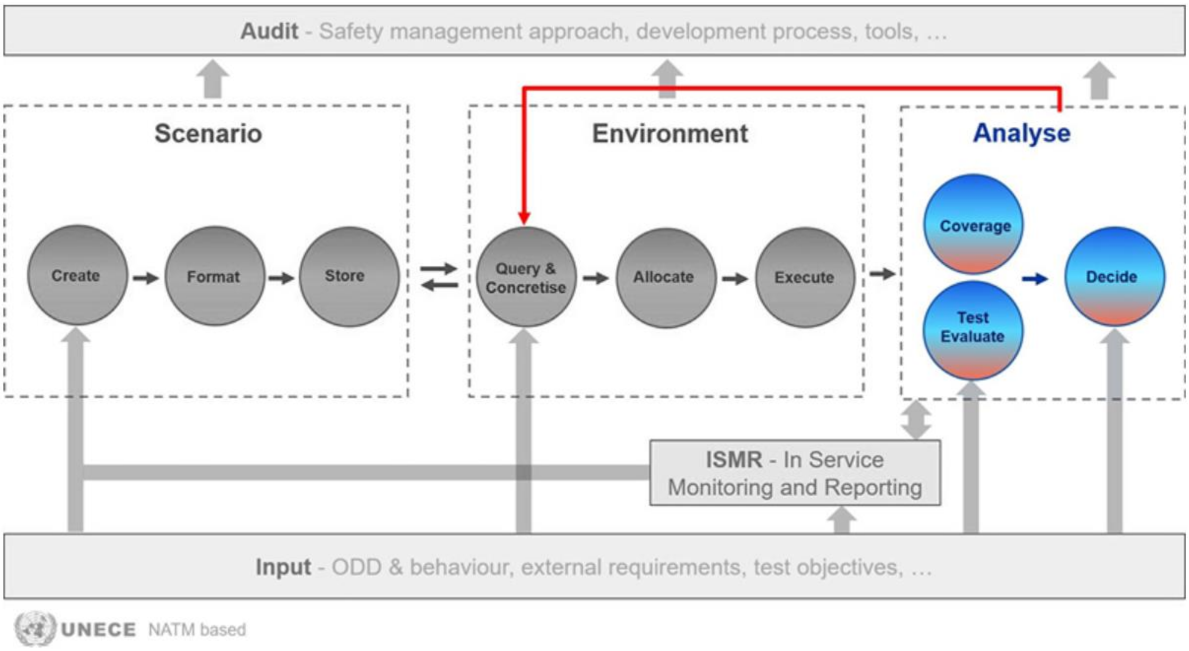


*Figure 14: The Analyse block*

The **Analyse** component evaluates test results to assess the system's safety through three interconnected stages: coverage analysis, test evaluation, and decision-making, see Figure 14.

4. **Coverage**
   This stage examines test coverage from multiple perspectives, including parameter ranges in logical scenarios and Operational Design Domain (ODD) features. Iterative feedback loops identify conditions where the system fails pass/fail criteria, enabling new concrete scenarios to be generated within the parameter space. ODD coverage is particularly critical, ensuring the system's operational boundaries are thoroughly tested using diverse scenarios.
5. **Test Evaluate**
   Individual test executions are assessed to determine whether the system meets safety requirements, such as maintaining speed limits or avoiding collisions. Results from both test evaluation and coverage analysis feed into iterative refinements of scenarios until coverage thresholds are achieved.
6. **Decide**
   Once sufficient coverage is reached, combined results from the previous stages are synthesised to produce the overall safety assurance outcome for the system.

The Analyse component ensures systematic evaluation, iterative refinement, and comprehensive safety assurance of the tested system.

### 5.3.1 Coverage

After execution, the data is analysed as part of the safety assessment of the system. There are two main blocks in this analysis: Coverage and Test Evaluate. The **Coverage** block examines the test from multiple perspectives and derives a combined coverage outcome. These perspectives might include analysing coverage of the parameter value ranges in the logical scenario using individual concrete parameter values. Based on the analysis outcome, an iterative process derives new concrete scenarios within given logical scenario's parameter ranges to identify failure conditions (e.g., parameter combinations that lead to system failure). This iterative feedback loop is indicated by the red feedback arrow.

Another aspect of Coverage Analysis is Operational Design Domain (ODD) analysis. Here, the accumulated ODD features covered by the set of test scenarios are examined to determine if enough of the system's ODD is covered. ODD is the focus because it defines the operational boundary within which the system is expected to operate safely. Thus, it is essential to thoroughly explore this claimed boundary using a diverse set of scenarios. These are just two examples that form the coverage concept, and the SUNRISE project will continue to explore this further in later stages.

### 5.3.2 Test/Evaluate

The **Test Evaluate** block assesses each test execution to determine whether the system has passed or failed. For instance, did the system stay within an upper speed limit, or did it avoid collisions? Both the Coverage and Test Evaluate blocks contribute to the overall Analysis block, and the results are used to select further concrete parameters within the original scenario's parameter ranges. After several iterations, once the coverage threshold is reached, the combined coverage and test evaluation results will feed into the Decide block, producing the overall safety assurance outcome for the system.

### 5.3.3 Decide

Inputs for the **Decide** block will be updated and included later from the SUNRISE deliverable D2.3, due in August 2025.

# 6 AUDIT

This chapter describes the Audit. The direct link to the Audit page of the Interactive web-based Handbook is: https://ccam-sunrise-project.eu/audit/

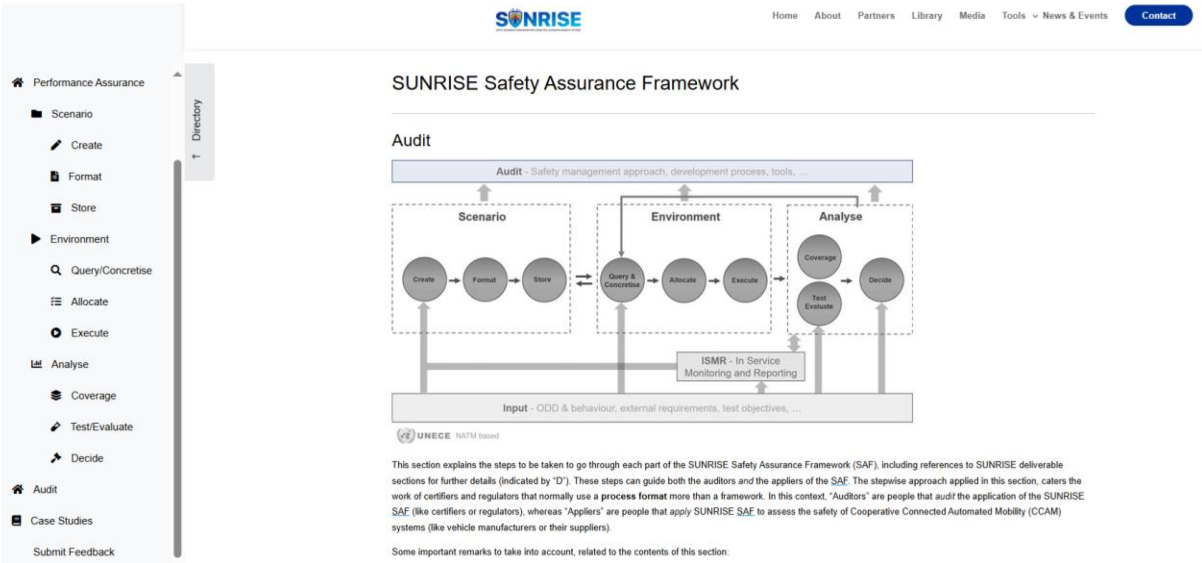


*Figure 15: Screenshot of the Audit page in the Handbook v1.0*

Figure 16 is an enlarged picture showing the Audit block.

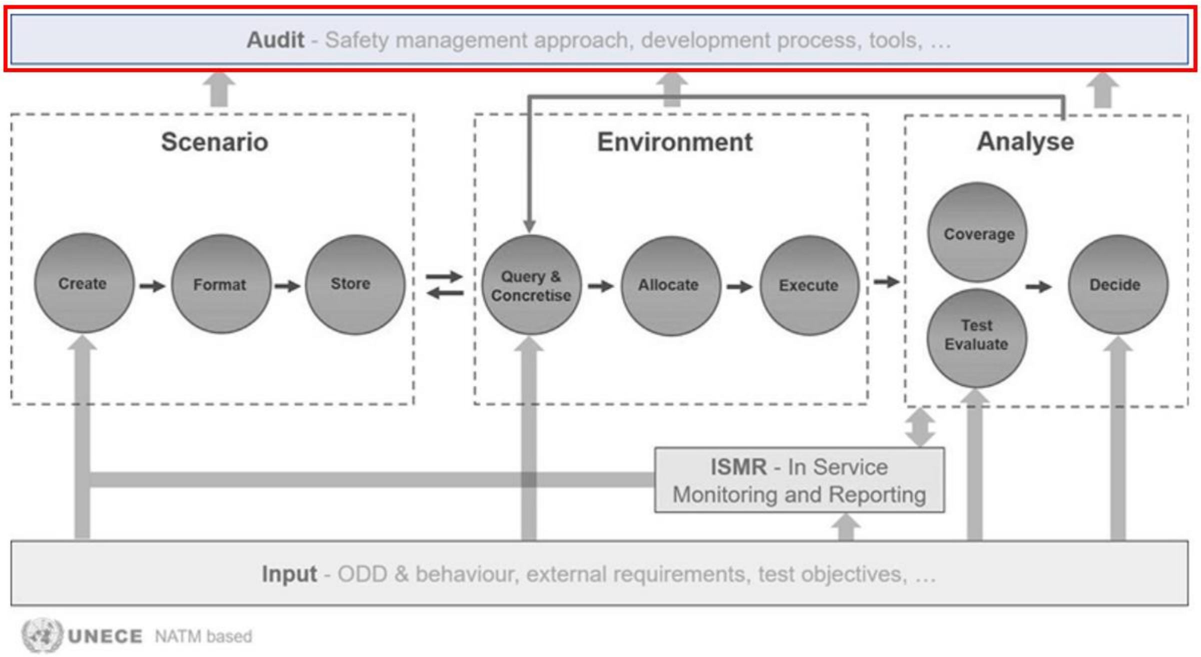


*Figure 16 : Audit*

This section explains the steps to be taken to go through each part of the SUNRISE Safety Assurance Framework (SAF), including references to SUNRISE deliverable sections for further details (indicated by "D"). These steps can guide both the auditors and the appliers of the SAF. The stepwise approach applied in this section, caters the work of certifiers and regulators that normally use a **process format** more than a framework. In this context, “Auditors” are people that *audit* the application of the SUNRISE SAF (like certifiers or regulators), whereas “Appliers” are people that *apply* SUNRISE SAF to assess the safety of Cooperative Connected Automated Mobility (CCAM) systems (like vehicle manufacturers or their suppliers).

Some important remarks to consider, related to the contents of this section:

1. This section might be modified and will be expanded with additional steps, based on contents of deliverables that were not available yet at the time of writing this section.
2. Auditing of CCAM safety assurance procedures (like the SUNRISE SAF), might involve aspects going beyond the steps described in this section, including (but not limited to) audits on the internal procedures for design, development, testing and overall safety management at CCAM technology developers (like vehicle manufacturer and their suppliers). Although not denying their possible importance, these aspects are not treated here due to scope limitations of the SUNRISE project.

## Scenario

By following the **steps** indicated below, a user of the SUNRISE SAF can **audit** the contents of the **Scenario block**. These steps aim to audit the overall characteristics and properties of the SCDB, and do not consider the actual content of individual test cases (which is part of the Execute block). Note that the Scenario sub-blocks (Create, Format and Store) are the responsibility of the scenario database owner and therefore not further elaborated in the SUNRISE SAF. For that same reason, auditing steps will not be specified for individual sub-blocks and apply only to the overarching Scenario block.

1. **Check Standardization Compliance**
   A. Verify that the scenario databases follow standardized formats for storing scenarios (D3.2 Section 7.1)
   B. Confirm alignment with data exchange and testing standards (D6.1 Section 2.6)
2. **Verify Source Documentation**
   A. Check that scenarios in the database include source information (D3.2 Section 4.2.3, Requirement A.3)
   B. Validate that sources are properly documented with data or expert knowledge origin (D3.2 Section 4.2.3, 8.1.1)
3. **Evaluate Database Extension Capabilities**
   A. Verify the database supports extensible parameter lists (D3.2 Section 5.5.2, Requirement H.2)
   B. Ensure the database can accommodate new parameters. For example when standards or protocols are updated. (D3.2 Section 5.5.2, 8.2.4)
4. **Verify Database Search Functionality**
   A. Confirm the database supports searching/querying using tags (D3.2 Section 4.2.2, Requirement A.2)

5. **Review Data Accuracy** (D5.3 Section 4.2.2)
    A. Assess the correctness of data entered into the scenario database
    B. Cross-reference with known benchmarks or references
    C. Verify that scenarios reflect real-world conditions accurately
6. **Review Data Consistency** (D5.3 Section 4.2.2)
    A. Check for uniformity across all scenarios in:
        - Units of measurement
        - Data formats
        - Terminologies used
        - Standardized naming conventions
7. **Check Data Freshness** (D5.3 Section 4.2.2)
    A. Check how up to date the scenarios are
    B. Verify current relevance of the database content
    C. Check last update timestamps
8. **Check Number of Scenarios** (D5.3 Section 4.2.2)
    A. Count total distinct scenarios available
    B. Obtain overview of database comprehensiveness
    C. Check scenario quantity metrics
9. **Check Covered Kilometres** (D5.3 Section 4.2.2)
    A. Check on how many Km's the scenario database is based
    B. Assess span and scale of included scenarios
    C. Quantify geographic coverage
10. **Verify Scenario Distribution** (D5.3 Section 4.2.2)
    A. Analyse breakdown of scenarios by:
        - Geographic regions
        - Road types
        - Weather conditions
        - Time of day
        - Traffic density
11. **Verify Scenario Complexity** (D5.3 Section 4.2.2)
    A. Assess difficulty levels across scenarios considering:
        - Number of vehicles involved
        - Presence of pedestrians
        - Road complexity
        - Environmental conditions

# Environment

## Query & Concretize

By following the **steps** indicated below, a user of the SUNRISE SAF can **audit** the contents of the **Query & Concretize block**, thereby ensuring that resulting test cases are of high quality, relevant and comprehensive for the intended testing purposes:

1. **Review the inputs** to the COTSATO (COncretizing Test Scenarios and Associating Test Objectives) process (D3.2 Section 7.3)

   A. Verify that the ODD description is provided and follows the format guidelines in ISO 34503
   B. Check that the system requirements are clearly defined
   C. Ensure the SUT (System Under Test) is properly specified
   D. Confirm that variables to be measured during test execution are listed
   E. Validate that pass/fail criteria for successful test execution are provided
2. **Examine the query** used to fetch scenarios from the scenario storage:
   A. Ensure the query is well-formulated and aligns with the requirements specified in D3.2 Section 7.4
3. **Verify the scenarios** retrieved from the scenario storage:
   A. Check that the scenario concept complies with the requirements outlined in D3.2 Section 4
   B. Confirm that the scenario parameters meet the requirements outlined in D3.2 Section 5
   C. Validate that the parameter spaces adhere to the requirements in D3.2 Section 6
4. **Review the test cases** generated by the COTSATO process (D3.2 Section 7.5)
   A. Ensure each test case includes a test scenario, metrics, validation criteria, and pass/fail criteria
   B. Depending on the purpose, verify that the metrics cover aspects such as safety, functional performance, HMI, operational performance, reliability, and scenario validation
5. **Check the mapping** of requirements to test cases (D3.2 Section 7.5)
   A. Confirm that a clear mapping exists between system requirements and the generated test cases
6. **Review the metrics on the collection of test scenarios** (D3.2 Section 7.5)
   A. Verify that information about representativity and source of test scenarios is provided
7. **Evaluate Individual Scenario Quality**
   A. Check the testing purpose metrics to ensure scenarios are relevant for the intended testing (D5.3 Section 3.1.1)
   B. Assess scenario exposure and probability to verify if scenarios represent realistic situations (D5.3 Section 3.1.2)
   C. Verify scenario description quality including completeness and unambiguity (D5.3 Section 3.1.3)
   D. Validate scenario consistency in terms of semantic and format consistency (D5.3 Section 3.1.4)
   E. Check scenario processability to ensure scenarios can be executed in the intended test environments (D5.3 Section 3.1.5)
8. **Evaluate Multiple Scenario Quality**
   A. Assess diversity and similarity between scenarios to avoid redundant testing (D5.3 Section 3.2.1)
   B. Verify scenario coverage to ensure comprehensive testing of the Operational Design Domain (ODD) (D5.3 Section 3.2.3)
   C. Check completeness of data across the scenario set (D5.3 Section 3.2.4)
9. **If Using Multiple Scenario Databases**

A. Use the Scenario Relatedness Index (SRI) described in D5.3 Section 5 to:
    - Check for identification and filtering out of redundant scenarios
    - Verify adequate coverage
    - Verify test efficiency optimization

## Allocate

By following the **steps** indicated below, a user of the SUNRISE SAF can **audit** the contents of the **Allocate block**, thereby ensuring that the allocation process was carried out correctly and comprehensively according to the guidelines provided by the SAF.

1. **Review the comparison** of test case requirements with test instance capabilities:
    - Ensure that the structure outlined in D3.3 Section 3 was followed, which includes scenery elements, environment conditions, dynamic elements, and test criteria (D3.3 Section 3.3).
    - Verify that the metrics described in D3.3 Section 4.3 were applied for the comparison.
2. **Check the decision-making process** for test case allocation:
    - Confirm that the process outlined in D3.3 Section 4.5 and D3.3 Figure 27 was followed.
    - Verify that the "virtual simulation first" approach was applied as described in D3.3 Section 4.2.
3. **Examine the documentation** of the allocation results:
    - Review the allocation matrix or table as described in D3.3 Section 4.6 and exemplified in D3.3 Figure 28.
    - Ensure that scenarios that could not be allocated or were not sufficiently tested are properly flagged and reported to the "Coverage" block of the "Analyse" part of the SAF.
4. **Verify the consideration of various metrics:**
    - Check that both functional and non-functional metrics were considered, as described in D3.3 Sections 4.3 and 4.4.
    - Confirm that safety was prioritized in the decision-making process (D3.3 Section 4.5).
5. **Review the reallocation process:**
    - Ensure that the iterative allocation to higher-fidelity test instances, when necessary, was performed as described in D3.3 Section 4.5.
    - Verify that the reasons for reallocation decisions were properly documented (D3.3 Section 4.6).
6. **Check for special circumstances:**
    - Review if any deviations from the general methodology were made due to special circumstances, and if so, ensure they were properly justified (D3.3 Section 4.5).
7. **Verify the completeness of documentation:**
    - Ensure that all steps of the decision-making process, including reasons for decisions, were documented and returned to the SAF (D3.3 Section 4.6).

    - Check for the presence of a tree structure containing all metrics and results of the comparison to all test instances (D3.3 Section 4.6).
8. **Review the consideration of safety standards:**
    - Verify that safety standards such as SOTIF were considered in the allocation process, particularly for identifying potentially triggering conditions or functional insufficiencies of the SUT (D3.3 Section 4.5).

## Execute

By following the **steps** indicated below, a user of the SUNRISE SAF can **audit** the contents of the **Execute block**, thereby ensuring that whatever simulation framework was used, it meets the essential requirements and produces reliable, validated results. It's important to note that while using the SUNRISE harmonized V&V simulation framework is recommended for better interoperability, modularity and scalability, it is not mandatory for the SUNRISE SAF.

1. **Verify that the simulation framework used aligns with the harmonized V&V simulation framework** described in D4.4 Section 4.5, which includes:
    A. Checking that the base layer contains the four core interconnected subsystems:
        - Subject Vehicle - Vehicle Dynamics
        - Subject Vehicle - Sensors
        - Subject Vehicle - AD function
        - Environment
    B. Confirming the framework uses standardized interfaces between subsystems, particularly ASAM OSI as detailed in D4.4 Section 4.4
2. **Validate the data formats used align with recommended standards** (D4.4 Section 4.3):
    A. ASAM OpenSCENARIO for scenario descriptions
    B. ASAM OpenDRIVE for road networks
    C. ASAM OpenLABEL for sensor data and scenario tagging
3. **Assess simulation model validation**:
    A. Verify that correlation analysis between virtual simulation and physical tests was performed (D4.1 Section 3.6, D4.1 Section 1.1, D4.2 Section 8.1 - R1.1_14)
    B. Confirm robustness and representativeness of virtual validation framework (D4.2 Section 8.1 - R1.2_10)
    C. Check that model quality metrics meet defined thresholds (D4.1 Section 3.6)
    D. Review the simulation model validation test report (D4.1 Section 3.6, D4.3 Section 4.4)
4. **Evaluate validation metrics and Key Performance Indicators (KPIs):**
    A. Review test case validation metrics (D4.1 Section 3.1)
    B. Verify that requirements from protocols, standards and regulations were used where applicable. For example from Euro NCAP or GSR (D4.2 Section 8.1 - R1.1_01, R1.2_01, R3.1_01)

C. Check that SOTIF requirements were addressed per D4.2 Section 4.1, including:
    - Risk quantification for scenarios, triggering conditions and ODD boundaries (D4.2 Table 12 - R10.1.19, R10.1.20, R10.1.21)
    - Validation results for known unsafe scenarios (D4.2 Table 12 - R10.1.20.1)
    - Validation results for discovered unknown unsafe scenarios (D4.2 Table 12 - R10.1.18.1)
    - Assessment of residual risk (D4.2 Table 12 - R10.1.6)

5. **Check test case execution results:**
    A. Check that all executed test cases generated desired results.
    B. Confirm test coverage metrics have been generated. For example:
        - Check EURO NCAP and GSR compliance metrics (D4.2 Section 8.1 - R1.1_01, R1.2_01, R3.1_01)
        - Verify sensor validation metrics were applied (D4.2 Section 8.1 - R1.1_02)
        - Review correlation coefficients between simulation and physical test results (D4.2 Section 8.2 - R2.1_49)
    C. Confirm that test results include both virtual and physical validation data where applicable (D4.2 Section 8.1 - R1.1_14)
    D. Verify that executed simulations correspond to the requests from scenario manager (D4.2 Section 8.1 - R1.1_25)
    E. Verify that test results are correctly documented and stored (D4.2 Section 8.1 - R1.1_27)

# Analyse

## Coverage

By following the **steps** indicated below, a user of the SUNRISE SAF can **audit** the contents of the **Coverage block**, thereby ensuring that the combined set of test case results sufficiently covers the ODD and the parameter value ranges:

1. **Verify Scenario Coverage** (D5.3 Section 4.2.3):
    A. Check if the coverage analysis includes all four types of coverage metrics described by de Gelder et al. (D5.3 Reference [59]):
        - Tag-based coverage: Verify that scenarios cover all relevant ODD aspects through specific tags
        - Time-based coverage: Confirm all timestamps in driving data are represented
        - Actor-based coverage: Ensure all relevant actors are included in at least one scenario
        - Actor-over-time-based coverage: Check that relevant actors are included throughout their period of importance
2. **Verify Parameter Coverage** (D5.3 Section 4.2.3):
    A. According to Laurent et al. (D5.3 Reference [60]), verify that:

- Parameters influencing ADS decision-making are adequately tested
- Multiple simulations with different parameter values have been run
- Changes in parameters lead to statistically significant differences in outcomes (if they should)
- Key metrics like path deviation and safety (minimum distance to objects) are considered

## Test Evaluate

*Further inputs for this section will result from the unfinished deliverable D3.5.*

## Decide

*Further inputs for this section will result from the unfinished deliverable D2.3.*

# 7 CASE STUDIES

This chapter describes the Case Studies. The direct link to the Case Studies page of the Interactive web-based Handbook is: https://ccam-sunrise-project.eu/case-studies-hb/

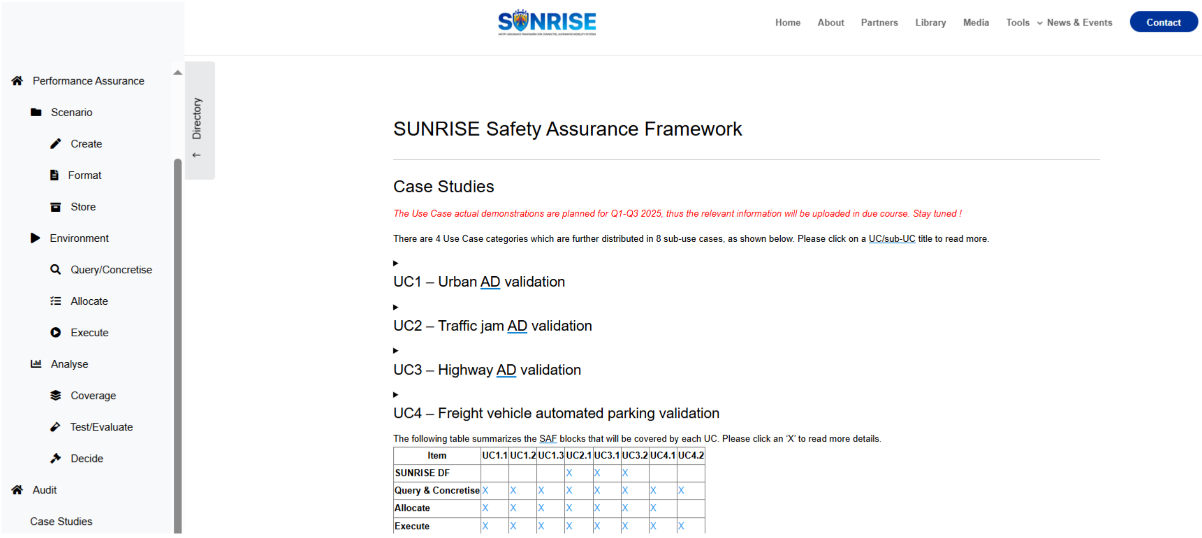


*Figure 17: Screenshot of the Case Studies page in the Handbook v1.0*

To facilitate the understanding of SAF users, the handbook will be enriched with Use Case demonstration results obtained in Work Package 7 (WP7). This is expected to happen between March and May 2025.

# 8 SUBMIT FEEDBACK

This chapter describes the Submit Feedback. The direct link to the Submit Feedback page of the Interactive web-based Handbook is: https://ccam-sunrise-project.eu/submit-feedback/

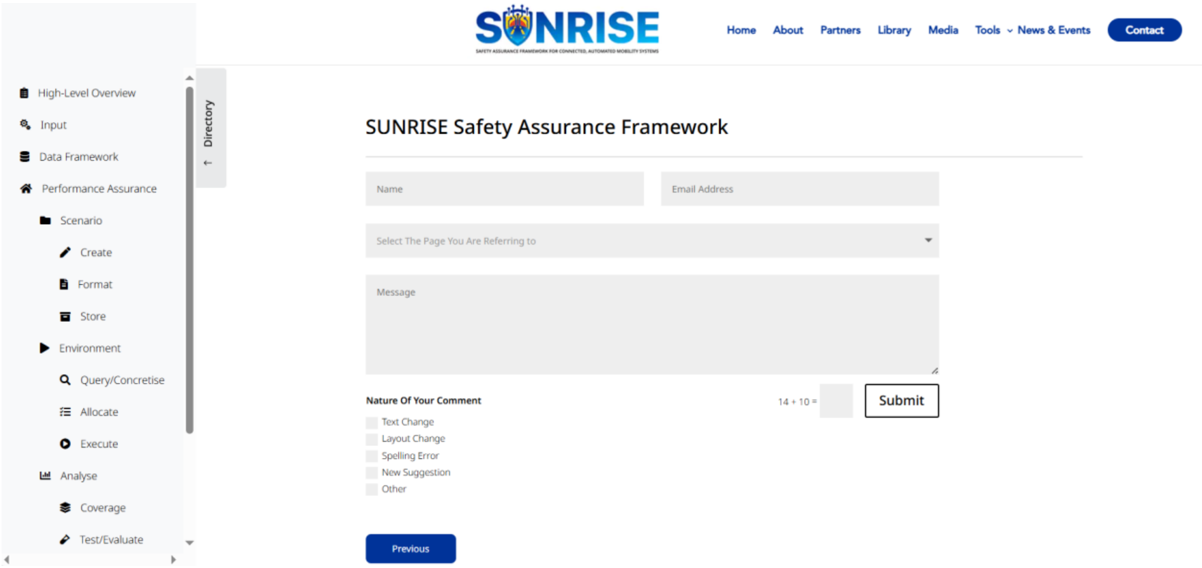


*Figure 18: Screenshot of the Submit Feedback page in the Handbook v1.0*

Users have the opportunity to provide comments and share their feedback on the content of the Handbook v1.0 by utilizing the 'Submit Feedback' form. This feature is included to encourage user engagement and ensure continuous improvement of the Handbook.

# 9 CONCLUSIONS

This deliverable presents the first version (version 1.0) of the Web-based Interactive Handbook for the SUNRISE Safety Assurance Framework (SAF). By using the Handbook both naïve and skilled user will get guidance how to use the SAF.

The first version (1.0) of the Handbook is based on the existing draft SAF (as developed in task T2.2) and is anticipated to be updated during the remaining project time, with new versions of the Handbook, until the Final Event in June 2025, and a last update at the end of the project in August 2025. After the project end, the Handbook might be transferred and updated in successor projects.

Input to the Handbook is mainly done via Task 2.2, where the SAF is being developed. The different blocks in the SAF are developed with technical input from the rest of the technical WPs (WP3 – Method, WP4 - Toolchain, WP5 – Ontology, and WP6 – Data framework).

The SAF Handbook described in this deliverable will later be enriched with Use Case demonstration results obtained in Work Package 7 (WP7).

# 10 REFERENCES

[1] UNECE (11 Apr 2023). (GRVA) – New Assessment/Test Method for Automated Driving (NATM) Guidelines for Validating Automated Driving System (ADS). Retrieved on 29 Jan 2025 from https://unece.org/transport/documents/2023/04/working-documents/grva-new-assessmenttest-method-automated-driving-natm

[2] ASAM (21 Nov 2024). ASAM OpenSCENARIO XML. Retrieved on 29 Jan 2025 from https://www.asam.net/standards/detail/openscenario-xml/

[3] ASAM (21 Nov 2024). ASAM OpenDRIVE. Retrieved on 29 Jan 2025 from https://www.asam.net/standards/detail/opendrive/

[4] de Gelder, E., Paardekooper, J.-P., Khabbaz Saberi, A., Elrofai, H., Op den Camp, O., Kraines, S., Ploeg, J., and De Schutter, B. (2022). Towards an Ontology for Scenario Definition for the Assessment of Automated Vehicles: An Object-Oriented Framework. IEEE Transactions on Intelligent Vehicles, 7(2), p. 300-314.

[5] Chang, C., Cao, D., Chen, L., Su, K., Su, K., Su, Y., Wang, F-Y., Wang, J., Wang, P., Wei, J., Wu, G., Wu, X., Xu, H., Zheng, N., Li, L. (2022). MetaScenario: A Framework for Driving Scenario Data Description, Storage and Indexing. IEEE Transactions on Intelligent Vehicles, p. 1-21

[6] Zhang, X., Khastgir, S., Asgari, H., and Jennings, P. (2021). Test Framework for Automatic Test Case Generation and Execution Aimed at Developing Trustworthy AVs from Both Verifiability and Certifiability Aspects. Paper presented at the IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 2021, pp. 312-319, doi: 10.1109/ITSC48978.2021.9564542.

[7] Safety Pool (n.d.). The Global Initiative For Certifiable AV Safety. Retrieved on 29 Jan 2025 from https://www.safetypool.ai/

[8] TNO (28 May 2024). Scenario-Based Safety Assessment of Automated Driving Systems. Retrieved on 29 Jan 2025 from https://www.tno.nl/en/newsroom/papers/scenario-based-safety-assessment/

[9] European Commission (5 Aug 2022). Automated cars – technical specifications. Retrieved on 29 Jan 2025 from: https://ec.europa.eu/info/law/better-regulation/have-your-say/initiatives/12152-Automated-cars-technical-specifications_en

[10] John Creamer and Francois Guichard (22 Mar 2024). FRAV/VMAD Integration. Retrieved on 29 Jan 2025 from: https://wiki.unece.org/pages/viewpage.action?pageId=224002635

[11] Connected Places Catapult (10 Jun 2021). Assuring the safety of connected and autonomous vehicles. Retrieved on 29 Jan 2025 from: https://cp.catapult.org.uk/report/assuring-the-safety-of-connected-and-autonomous-vehicles/

[12] SAKURA (n.d.). Project Info. Retrieved on 29 Jan 2025 from https://www.sakura-prj.go.jp/project_info/

[13] TNO (n.d.). StreetWise: scenario based safety assessment for automated driving. Retrieved on 29 Jan 2025 from https://www.tno.nl/en/digital/smart-traffic-transport/safety-assessment-automated-driving/streetwise/

[14] VVM (n.d.). Our vision. Retrieved on 29 Jan 2025 from https://www.vvm-projekt.de/en/