Real and synthetic scenarios generated for the development, training, virtual testing and validation of CCAM systems

SYNERGIES

D5.1 Scenario Methodology Framework

Document Type	R – Document, Report
Document Number	D5.1
Primary Author(s)	Iskra Gasparic | AVL List GmbH
Document Version / Status	1.0 | Final Version
Distribution Level	PU (public)

Project Acronym	SYNERGIES
Project Website	www.synergies-ccam.eu
Project Coordinator	IDIADA AUTOMOTIVE TECHNOLOGY SA
Grant Agreement Number	101146542
Date of latest version of Annex I against which the assessment will be made	2024-05-27

Funded by the European Union

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.

CONTRIBUTORS

Name	Organization	Name	Organization
Sven Tarlowski	RWTH IKA	Elena Daskalaki	ICCS
Christoph Glasmacher	RWTH IKA	Erwin de Gelder	TNO
Joaquin J. Vilches Vilchez	CTAG	Rocio Otero Díaz	CTAG
Andreas Rauschert	BMW Group	Marc El Zeenny	IRT SystemX
Iskra Gasparic	AVL	Erwan Revert	IRT SystemX
Florian Klück	AVL	Didier Wright	IRT SystemX
Thomas Amler	IDIADA	Yavar Taghipour Azar	Vicomtech
Clement Val	CEESAR	Alejandro Hernández	Vicomtech
Ashfaq Farooqui	RISE	Lucas Dulewicz	VUFO
Thanh Bui	RISE	Konrad Reisinger	VUFO
Jorge Lorente Mallada	TME	Jerein Jeyachandran	UoW

FORMAL REVIEWERS

Name	Organization	Date
Olaf Op den Camp	TNO	2025-12-09
Jordi Pont	IDIADA	2026-01-29

DOCUMENT HISTORY

Revision	Date	Author | Organization	Description
0.8	2025-01-24	Iskra Gasparic | AVL	Project milestone (M4), Preliminary D5.1
0.9	2026-01-23	Iskra Gasparic | AVL	Final document revision
1.0	2026-01-29	Jordi Pont | IDIADA	Version ready for submission to EC

DOCUMENTATION OF SUBMISSION DELAY (OPTIONAL)

Planned date	Actual date
2025-11-30	2026-01-29

Justification	
Partner(s) causing the delay:	AVL
Justification:	All partner contributions to Task 5.1 were delivered on time. During the internal task-level review, several chapters were identified where further clarification, alignment, and technical refinement would significantly improve the overall quality and consistency of the deliverable. Implementing these improvements required additional coordination with contributing partners and careful integration of feedback, which could not be fully ensured within the original submission deadline. A short extension was therefore requested to ensure a higher-quality and more robust deliverable.
Effects on follow-up activities and other tasks:	No significant impact on follow-up tasks, milestones, KPIs or dependencies. The delay is limited in scope and duration and does not affect the overall WP5 timeline or interactions with other work packages.

Specific mitigation actions to catch up:	
Statement deliverable owner:	AVL. The deliverable owner confirms responsibility for the delay and commits to submitting the revised deliverable by the agreed new deadline (30/01/2026), ensuring full compliance with Annex I and SYNERGIES quality standards.
Statement task leader:	AVL. The task leader confirms that the delay is justified by quality-related improvements and that no negative impact on task objectives or dependencies is foreseen
Statement WP leader:	IKA. The WP leader acknowledges the delay and confirms that it does not affect the overall WP progress or planned deliverables.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 EXECUTIVE SUMMARY

Advanced Driver Assistance Systems (ADAS) and Automated Driving (AD) technologies are designed to operate in complex, dynamic, and often unpredictable traffic environments, making it essential to evaluate their performance across a wide range of relevant and representative situations. For that reason, the identification, extraction, enrichment, and generation of scenarios remain critical for creating test cases that support the verification and validation of such systems. Rather than relying only on physical testing or isolated functional checks, simulation-based scenario methodologies enable a systematic and scalable approach to evaluate how ADAS/AD systems respond to real-world challenges. These scenarios encapsulate both ordinary and edge-case situations, allowing detection of potential failures, and refinements of system behaviour under controlled yet diverse conditions. Within the SYNERGIES project, a structured methodology is developed to transform available knowledge and collected data into executable test scenarios. This methodology ensures that scenarios are representative of real-world traffic situations, but also of system specifications.

The process begins with the scenario concept, which defines the ontology and rules for scenario representation. This ontology captures relevant actors, infrastructure, manoeuvres, and interactions, forming the conceptual space of possible scenarios. It is informed by expert knowledge, literature, and real-world road interaction patterns. Building on this foundation, abstract and logical scenario definitions formalize the scenario space. Abstract scenarios offer high-level, declarative descriptions, while logical scenarios provide parameterized, machine-readable specifications. These definitions are enriched through the identification and processing of non-ordinary situations, such as rare incidents or near misses, which may reveal gaps in the ontology or new scenario types.

The SYNERGIES project applies this methodology to identify and extract scenarios from recorded project data and enrich them using external data sources. The data is then mapped to the ontology's semantic framework, bridging raw observations with structured definitions.

Before concrete instantiation, statistical methods can be used to estimate parameter distributions based on road conditions, driver behaviour statistics, and empirical observations. This ensures that generated scenarios reflect the variability and uncertainty inherent in real-world traffic systems. Alternatively, scenarios are sampled for systematic domain space coverage. Concrete scenarios are created out of logical scenarios by sampling parameter spaces and applying stochastic variability to get specific values, trajectories, and environmental conditions. The concrete scenarios are finally converted into test scenarios. These are formatted for compatibility with simulation environments, enabling systematic evaluation.

Finally, test scenarios are deployed into simulation platforms. Feedback from simulations is used to refine earlier stages, ensuring continuous improvement of the test scenario generation pipeline.

To support long-term relevance, the methodology also includes Scenarios of Tomorrow (SoT), which anticipate future changes in technology, infrastructure, and behaviour. This forward-looking approach ensures that the scenario framework remains robust and adaptable.

Within the project, interfaces for reading raw and annotated data, as well as for generating scenarios based on knowledge, are clearly defined. Interactions between methods and tools are coordinated to establish a seamless workflow that produces valuable and executable scenarios. The feasibility of these scenarios is demonstrated by uploading them to a Scenario Dataspace, developed and maintained within the project.

The goal is to standardize scenario generation and enrichment methods, leveraging insights from ongoing and past activities in Europe, Japan, and the US. This comprehensive, ontology-driven, and data-enriched methodology ensures that scenario generation is rigorous, scalable, and future-proof—supporting both current and emerging needs in ADAS/AD system development and validation.

Keywords: Scenario, Scenario Generation, Scenario Identification, Scenario Extraction, Scenario Enrichment

2 OBJECTIVES

The document describes the achievements within the SYNERGIES project related to defining and harmonizing the methodology for scenario identification, extraction, enrichment, and generation. Three main objectives have been identified, in accordance with objectives of WP5:

1. **Scenario Methodology**: Define a harmonized workflow for scenario identification, extraction, enrichment, and generation.
2. **Cross-project Integration**: Utilize knowledge and practices from ongoing and completed projects such as SUNRISE, Hi-Drive, VV Methods, SAKURA, and ADScene, leveraging insights from initiatives in the EU, Japan, and the US.
3. **Interoperability**: Create a flexible, modular, and globally applicable methodology that not only integrates existing knowledge but also lays the groundwork for future advancements in scenario-based research and testing.

The primary goal is to develop a harmonized and continuous tool chain for scenario identification, extraction, enrichment, and generation, ensuring compatibility with diverse data sources and alignment with established research and industrial tools. This methodology will integrate knowledge and practices from ongoing and completed projects. By building on these existing efforts, the tool chain will address the complexities of heterogeneous data sources while adhering to the requirements and target use cases. The results directly support the SYNERGIES Objective 1: “Deliver widely accessible scenarios”, by developing tools that enable identification and generation of scenarios that will be stored in a database accessible from the SYNERGIES platform. Methods for scenario identification, generation, and enrichment are enhanced by AI, in accordance with SYNERGIES Objective 5: “Enable a solid AI foundation for CCAM.”

Furthermore, the methodology is motivated by SYNERGIES Objective 3: “Enable the use of heterogenous and inclusive data sources for the generation of scenarios,” as the harmonized toolchain will be designed to extract meaningful scenarios from diverse sources, enhance them with relevant contextual information, and generate new, representative scenarios that support safety validation. As a direct consequence of the variety of generated scenarios, the activities will support the SYNERGIES Objective 2: “Maximize the usability and coverage of the scenarios” that will be of great value to different stakeholders. This process will align with international standards and best practices, ensuring compatibility with frameworks like ASAM OpenSCENARIO [1].

Ultimately, this task aims to create a flexible, modular, and globally applicable methodology. By fostering collaboration between stakeholders in the EU, Japan, and the US, the methodology will promote innovation and drive the adoption of harmonized tools in the global automated driving ecosystem.

3 DESCRIPTION OF WORK

3.1 Introduction

The development of robust and representative scenarios is a cornerstone of the SYNERGIES project's approach to testing and validating Connected, Cooperative, and Automated Mobility (CCAM) systems. As automated driving technologies evolve, the need for scalable and data-driven scenario generation methodologies that anticipate future applications and evolving requirements becomes increasingly critical. This chapter outlines the comprehensive framework developed within Work Package 5 (WP5), detailing the processes, tools, and concepts that enable the transformation of raw data and expert knowledge into simulation-ready scenarios.

This methodology is the result of a multi-step process that integrates insights from related European and international projects, defines a formal scenario concept for interoperability, and establishes a modular workflow for scenario identification, enrichment, and generation. To ensure that the methodology builds on existing knowledge and aligns with best practices, we began by reviewing related projects (see Chapter 3.2). that contributed valuable tools, concepts, and standards that informed the structure and requirements of the SYNERGIES framework.

Central to the methodology is the scenario concept (Chapter 3.5), which provides a unified and formalized way to describe scenarios across different levels of abstraction: functional, abstract, logical, and concrete. The concept proposed within this document enables consistent representation of traffic situations and supporting interoperability across tools and platforms.

Building on this foundation, the methodology addresses the full lifecycle of scenario creation:

- **Scenario Identification, Extraction, and Enrichment:** This phase involves detecting relevant traffic events from raw data sources, including sensor streams, traffic accident reconstructions, and AI-generated inputs. Techniques such as semantic compression, non-ordinary situation identification, and direct conversion of observations are used to extract meaningful scenarios and enrich them with contextual metadata, parameter distributions, and criticality metrics.
- **Scenario Generation:** Once scenarios are defined, they are instantiated into concrete test cases by generation from abstract scenarios and generation from logical scenarios. These processes rely on formal specification tools such as Traffic Sequence Charts (TSCs) and probabilistic sampling techniques to produce diverse, simulation-ready scenarios.
- **Scenarios of Tomorrow:** To ensure long-term relevance, the methodology includes a dedicated process for identifying and generating future-oriented scenarios. These scenarios anticipate emerging traffic patterns, new mobility actors, evolving human behaviours, and infrastructure changes. Human-in-the-loop (HiL) simulation and behavioural modelling are used to validate plausibility and refine definitions.

Throughout the development of this methodology, significant effort was placed on ensuring interoperability between components. Each stage of the process was designed with clearly defined input-output interfaces, enabling seamless data flow and tool integration. This modular and standards-aligned architecture ensures that the methodology is not only technically rigorous but also adaptable to future developments and compatible with existing simulation environments.

The following subchapters provide detailed descriptions of each component in the methodology, outlining the technical processes, data requirements, and implementation strategies that underpin the SYNERGIES scenario generation framework.

3.2 Related projects

The related projects contribute significantly to the SYNERGIES project, each enhancing different aspects of the methodology for scenario identification, extraction, enrichment, and generation. SUNRISE provides a federated scenario framework and tools for scenario validation, ensuring safety and performance standards are met. The scenario concept is defined in a way that leverages the mentioned projects and activities, such as HEADSTART, SUNRISE, PEGASUS, VV Methods, L3Pilot, and Hi-Drive, to define the common understanding of the concepts that are used throughout WP5. Hi-Drive contributes scenario concepts and development tools for highly automated driving, while VV Methods focuses on verification and validation methods, ensuring safety-compliant scenario generation. ADScene offers an industrial scenario library platform that supports scenario management and analysis, complementing the SYNERGIES project by enriching scenarios with diverse real-world conditions. Moreover, the expertise from ASAM working groups and ISO standards is of great importance, as the scenario methodology highly depends on the standards for both understanding common practices, as well as defining the outputs and data flow.

The experience from PEGASUS and VV Methods is utilized for formalizing the scenario criticality in the context of an Automated Driving System. Moreover, the layered scenario description that is introduced within these projects is used for adding metadata to scenarios. The ADScene project provides experience about vehicle data processing.

ADScene provides experience for general synthetic scenario generation methods, whereas L3PILOT project includes a specific use case of urban scenario definition, with the relevant parameters and domain ranges. The methods and concepts from PEGASUS and VV Methods support scenario generation in general as well. Together with experience from SUNRISE, the two projects support critical scenario generation methodology in particular. SUNRISE provides methods and strategies to validate synthetic scenarios against real-world scenario data.

Collectively, these projects enrich the SYNERGIES methodology by providing diverse tools, concepts, and standards to create a comprehensive and harmonized approach for scenario generation and validation. The participation of SYNERGIES partners in these projects is shown in Table 1.

Table 1: Overview of projects and standardization activities relevant for the SYNERGIES project, together with corresponding partners

PROJECT	HEADSTART [2]	SUNRISE [3]	L3PILOT [4]	HI-DRIVE [5]	PEGASUS [6] and VV METHODS	ADSCENE [7]	ISO3450x [8]	ASAM [9]
AVL		🟩						🟩
BMW			🟩	🟩	🟩		🟩	🟩
CEESAR						🟩		
CTAG				🟩				
DLR			🟩	🟩	🟩			🟩
ERTICO	🟩	🟩						
ICCS		🟩		🟩				🟩
IDIADA	🟩	🟩						
IDIADA DE	🟩	🟩						
IKA	🟩	🟩	🟩	🟩	🟩			🟩
IRT - SystemX						🟩		
IVEX								🟩

PSA			🟩	🟩		🟩		
Renault Group		🟩				🟩		
RISE	🟩	🟩						🟩
SIEMENS NL		🟩						🟩
TME	🟩	🟩	🟩	🟩				
TNO	🟩	🟩	🟩	🟩			🟩	
TUE								
UniGE								
UoWarwick		🟩						🟩
VICOMTECH	🟩	🟩						🟩
ViF	🟩	🟩						
VUFO								

3.3 Relation to other SYNERGIES activities

Deliverable 5.1 establishes the primary methodology for the activities within Work Package 5. The goal is to align the identification, generation and prediction of scenarios based on collected data, and the relation of the WP5 to other SYNERGIES activities is shown in Figure 1.

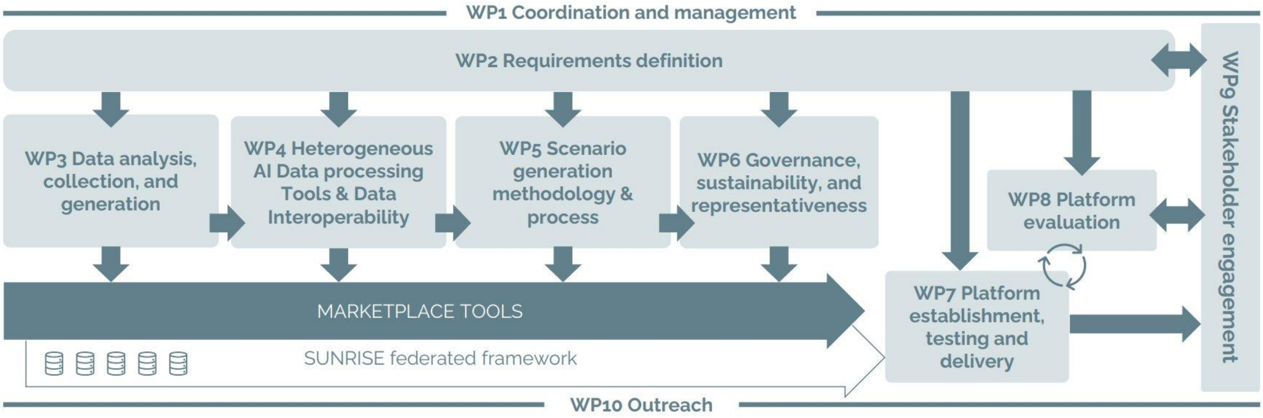


Figure 1: SYNERGIES Work Package overview

Within the project, data is collected and synthetically generated in Work Package 3, as well as being pre-processed using tools developed in Work Package 4. The information gained from the process steps in Work Package 5 is used in Work Package 6 to analyse those scenarios using metrics and tools developed there. To enable reproducibility of the scientific approaches, together with the generated scenarios, the tools developed in Work Package 5 will be published on the SYNERGIES marketplace, which is developed in Work Package 7. To ensure this collaboration of multiple work packages and stakeholders, Work Package 2 will define interoperability requirements to support the alignment over the work packages. These requirements are essential for all work packages to align, and Work Package 5 will demonstrate how these requirements are implemented.

Demonstrating the essential role of Work Package 5, this deliverable will present the overall methodology of this work package and explain how it achieves its defined role in relation to the other work packages and within the project.

3.4 Scenario generation methodology and process

The state-of-the-art methodology for scenario generation uses the standard executable scenario formats and corresponding metadata format, such as those specified by ASAM [9]. However, the standardization of higher-level scenario descriptions that would serve as an input to scenario generation tools still has not been achieved. To address this gap and ensure interoperability across the complete scenario-based testing workflow, the methodology developed within the SYNERGIES project is structured around a multi-layered scenario concept. This concept integrates static infrastructure elements (MOSAR) with dynamic traffic interactions (CRUISE). These layers are formalized using ontologies that define the actors, manoeuvres, environmental conditions, and their relationships that characterize traffic scenarios. The scenario concept and the ontology approach are further described in Chapter 3.5. This conceptual foundation supports the creation of abstract and logical scenarios, which are then instantiated into concrete scenarios and test scenarios using statistical modelling, semantic compression, and formal specification techniques. Scenarios can be generated using a wide range of data sources, such as recorded road-driving data, accident logs, drone data, or synthetic data. In addition, scenarios may be generated based on available knowledge about the ADAS/AD features, including their ODD.

3.4.1 Requirements

This chapter outlines the high-level requirements of the scenario methodology, which form the basis for defining the corresponding methodology components and their interactions. The requirements are listed in Table 2.

Table 2: Requirements on scenario methodology

ID	Topic	Requirement
#1	Input format	A consistent interface (file format) for data recording is required to support the diverse sources of data (recordings from test drives, surveillance cameras, drone-based recordings, accident data bases). It enables development of consistent data-processing algorithms.
#2	Coordinate system	The input data format should also specify the coordinate system. E.g., ASAM OSI specifies the global ground truth to be in a global cartesian coordinate system.
#3	Scenario output format	The scenario export format shall be standardized and transparent to enable broad use across companies and institutions. It shall support the relevant use cases of the project and be established and tested within the project context. It shall be supported by relevant tools. In addition, the provision of alternative output formats for different simulation tools shall be possible. Wherever feasible and applicable, scenarios made available on the platform shall use the ASAM OpenSCENARIO XML format.

#4	Common concept	The scenario descriptions for scenario identification and scenario generation should be based on a common set of concepts.
#5	Abstraction levels	The generated and identified scenarios can be classified as functional, abstract, logical and concrete.
#6	Data	Ground truth data shall be provided by WP3 for a subset of the data to enable scenario detection validation, training of AI-based models, and data-based scenario generation.
#7	Utilization of data	The method and tools shall be capable of making efficiently using input data and detected scenarios.
#8	Human understandable	The generated and identified scenarios should be human-understandable.
#9	Reproducibility	The methodology should enable reproducible generation and identification of scenarios.
#10	Traceability	The methodology should provide transparency regarding the input information from which scenarios are generated.
#11	Searchability	Relevant information should be provided to enable retrieval of a pool of a scenarios for a specific purpose (e.g. a given AD function or ODD).
#12	Tool interfaces	A technical interface, i.e. specifications for packaging developments and delivering them, needs to be specified (WP7), and Tasks 5.2 / 5.3 need to deliver their developments in accordance with these specifications.
#13	Tool orchestration	Traceability and job-orchestration tools shall be implemented within the platform (WP7) to enable execution of the tools developed in Task 5.2, possibly multiple times and with different versions.
#14	Data conversion tool	Tools shall be provided to convert object-level data (detected and tracked entities in the environment) into the semantic framework (T4.2), and to support manipulation of data within the abstract model (T4.3).
#15	Accident-based scenario creation tool	The methodology should enable manual creation of abstract scenarios and the direct creation of concrete scenarios in the case of accident-based scenarios.

#16	Test scenarios	The methodology should support the creation of test scenarios for use in simulation-based testing. The testing purpose and the system under test (SUT) may vary depending on stakeholder needs, which may influence the required level and type of scenario information.
#17	Redundancy	The methodology shall prevent the generation of redundant scenarios.
#18	Scenario relevance	The methodology should be capable of generating relevant (test) scenarios (e.g. challenging situations based on accident data) for a specific SUT (e.g. AD functions, collective perception testing).
#19	Quality	The methodology should ensure that generated scenarios meet a minimum quality level appropriate for their intended purpose. Each scenario should be clearly and unambiguously defined to avoid multiple interpretations.
#20	Standardized scenario input / working base	The methodology should enable seamless integration with various simulation platforms by defining common parameters (e.g. vehicle types, road layouts, weather conditions) and consistent units, allowing data to be shared and reused without extensive reformatting.
#21	Integration of existing Human-in-the-Loop simulation toolchains	Integration and compatibility between the SYNERGIES platform and existing toolchains should be established to the greatest extent possible.

3.4.2 Scenario methodology workflow summary

Within the SYNERGIES project, partners have created a workflow of the scenario methodology process, that is shown in Figure 2. The workflow outlines how Scenario Source Data, ontology-based abstractions, and scenario generation processes are combined into a coherent and interoperable framework. By linking data-driven analysis with logical, abstract, and concrete scenario representations, the methodology ensures consistency and traceability across different levels of abstraction.

The closed-loop structure of the workflow enables continuous refinement through simulation results, knowledge extraction, and ontology evolution, while the explicit consideration of Scenarios of Tomorrow ensures that the methodology remains applicable to future developments. Together, these elements establish a robust foundation for systematic, scalable, and forward-looking scenario-based evaluation. All the workflow components are described in more detail within this document.

The following sections guide the reader through the workflow by detailing the associated data flows, abstraction levels, and feedback mechanisms. For clarity, the text should be read in conjunction with Figure 2.

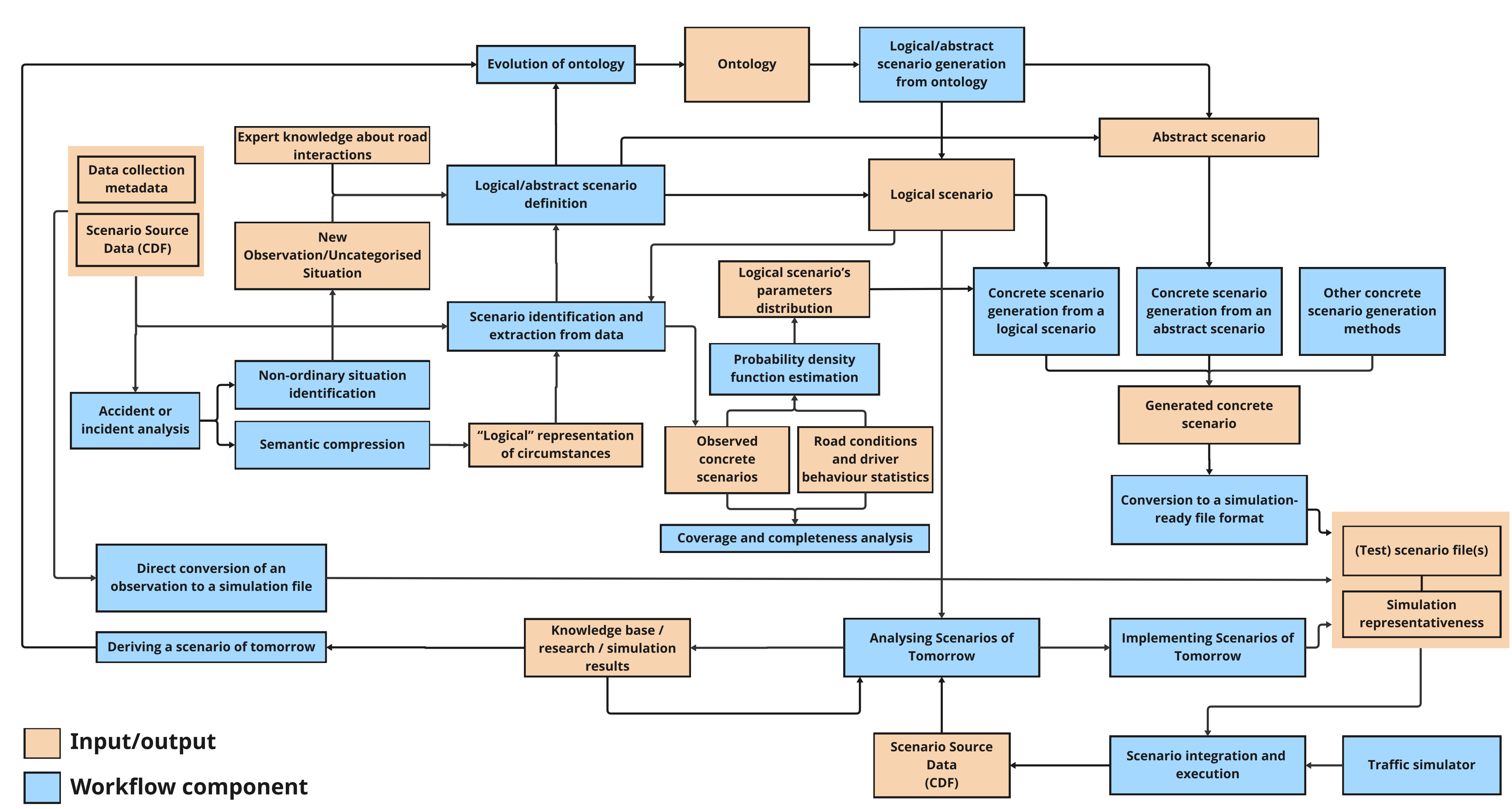


Figure 2: Scenario generation methodology integrating multiple components, ranging from ontology-based conceptualization to a simulation-ready scenario

To ensure interoperability of different components, the methodology must ensure efficient information exchange through common formats and data models. Within this deliverable, the SYNERGIES partners have defined a scenario concept, which is described in more detail in Chapter 3.5. The concept is based on a standardized naming of different scenario elements and properties of all actors, which is described within an ontology. The ontology defines a scenario space, representing any possible combinations of infrastructure elements and actor behaviours. With the extension of the scenario space and the applications, the Ontology itself is continuously updated through an evolution of ontology, incorporating newly identified concepts.

The workflow integrates three sections, each described in more detail within this document: Scenario identification, extraction, enrichment (Chapter 3.6), Scenario generation (Chapter 3.7) and Scenarios of Tomorrow (Chapter 3.8).

Scenario identification, extraction, enrichment

Transforming raw scenario source data into executable test cases requires a structured pipeline. The process relies on **Scenario Source Data (SSD)** as input. With the same motivation as for the scenario concept, the format has been established within SYNERGIES for storing SSD, referred to as the **Common Data Format (CDF)**. This harmonization ensures that methods for data processing can be consistently applied to information collected from different sources and data collection setups. The SSD describes the spatio-temporal properties of driving situations, including geometry, state, and kinematics of all actors and the road infrastructure (Chapter 3.4.3). The SSD is retrieved from data collection campaigns, by **accident or incident analysis**, based on knowledge-based inputs or generated using AI models. As part of the workflow, the SSD can be directly used in simulation by **direct conversion of an observation to a simulation file**, in which case the simulation files define explicit trajectories for the scenario participants. Alternatively, the SSD is further processed for deeper understanding of the circumstances.

Using data mining approaches, **non-ordinary situation identification** is performed to detect near misses or other rare events. The distinction between ordinary and non-ordinary is not always possible during the analysis, especially if the System Under Test (SUT) is not defined. If a **new observation/uncategorised situation** is identified and cannot yet be mapped to an existing logical or abstract scenario, a new scenario needs to be created through **logical/abstract scenario definition**, supported by **expert knowledge about road interactions**.

In parallel, the SSD is transformed into the logical-level data through **semantic compression**. During this step, information such as road geometry and vehicle positions is replaced with semantic descriptions using the ontology (e.g., "vehicle on the lane" or "vehicle in front of ego"). The result is a **"logical" representation of circumstances**, which reflects the source data at a higher level of abstraction. Based on this representation, **scenario identification and extraction from data** is performed to find corresponding occurrences. Techniques such as manual annotation, anomaly detection, and metric-based hazard identification are used to extract meaningful scenarios, for example, those containing specific ODD elements. The process results in observed concrete scenarios, which serve as data-driven foundation for further analysis. The target of data collection is to obtain diverse samples of possible road situations. To assess whether this goal has been achieved, **coverage and completeness analysis** is performed. Its results can be used to evaluate the achieved ODD coverage, interpret testing results, and plan future data collection activities.

Together with **road conditions and driver behaviour statistics** (e.g., number of rainy days per year, average speed on specific types of roads depending on presence of rain, etc.), the observed parameter values are used for **probability density function estimation** for a given logical scenario. The probability density function therefore combines general statistics and actual observations. This function is used for the logical scenario's parameter distribution, which can be used, together with other sampling techniques, for generating concrete scenarios. At this point, the data-based workflow connects to the broader scenario generation methodology described below (see Chapter 3.7 for more detail).

The described pipeline, moving from raw observations to logical families and concrete test cases, guarantees that SSD inputs are transformed into scenarios that are realistic, diverse, and directly usable for safety assurance. The process is once again visualized in Figure 3.

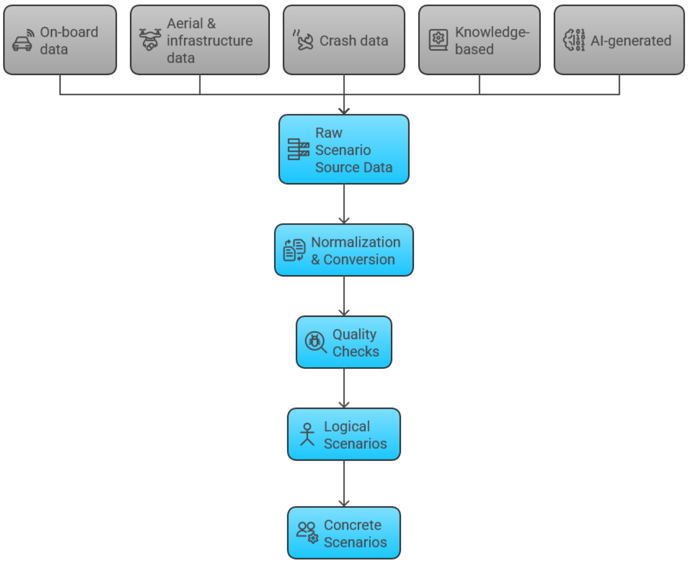


Figure 3: Workflow from raw data to logical and concrete scenarios

Scenario generation

The scenario generation process starts with a **logical scenario** or **abstract scenario** as input. These scenarios may be created based on real-world data, as described above, or defined based on other sources from the ontology such as standards, legislation, expert knowledge, ODD definitions, system requirements, etc. This is illustrated in the workflow as "**Logical/abstract scenario generation from ontology**". Both pats result in the same input format, following the common scenario concept, and are therefore handled identically by the scenario generation process.

An **abstract scenario** provides a formalized, declarative description of a traffic scenario at a high level of abstraction. Because of that, it allows flexibility in selecting specific parameter values during **concrete scenario generation from an abstract scenario**. Concrete scenarios are generated using methods such as constraint solving or Monte Carlo sampling, ensuring that the declared relations within the abstract scenario are fulfilled.

When performing **concrete scenario generation from a logical scenario**, the logical scenario already defines the parameter space, since it is the unambiguous, machine-readable definition. Therefore, any plausible combination of parameter values that satisfies the logical description can be used to generate a concrete scenario, either by applying different sampling methods, or by using the logical scenario's parameter distribution estimated based on real data.

Alternatively, **Other concrete scenario generation methods** may be applied, for example, approaches based on AI models that generate concrete scenarios without explicitly formalizing scenarios at higher abstraction levels (abstract/logical).

In all three cases, the result is a **generated concrete scenario**. This scenario then undergoes **conversion to a simulation-ready file format** to prepare the scenario for execution. The result of this process is one or more scenario or test scenario (short: (test) scenario) file(s), possibly enriched with information on **simulation representativeness**.

Scenarios of Tomorrow

To ensure that the scenario methodology not only addresses current applications but also anticipates future needs, **analysing Scenarios of Tomorrow** is performed. This analysis produces insights into required scenario enhancements, including changes in logical scenario parameter distributions, behaviour parameter distributions, and predictions of emerging scenarios along with their risk estimation. The discoveries are documented in a **knowledge base/research/simulation result** and are provided as recommendations and feedback to standards and regulatory bodies.

Once a Scenario of Tomorrow has been defined, **implementing Scenarios of Tomorrow** is carried out by generating a concrete scenario based on its description. The resulting scenario is prepared together with a **traffic simulator** through **scenario integration and execution** and executed in, for example, a human-in-the-loop driving simulator. As a result of simulation, new SSD becomes available, closing the loop and enabling further analysis. Therefore, the entire described SoT workflow serves as the foundation for **deriving a Scenario of Tomorrow**, achieved by adapting existing scenarios and by deriving completely new scenarios based on accumulated knowledge and simulation results. The Scenarios of Tomorrow will often require ontology extensions, since the target is to describe scenarios that are not yet frequently considered.

3.4.3 Scenario Source Data

As explained in Chapter 3.3, the scenario methodology partially relies on processed data from Work Package 4. Before describing the methodology in more detail, this chapter provides an overview of the WP4 data used as input to the complete methodology, as well as of the format in which it is provided.

Scenario Source Data (SSD) provides the raw material from which SYNERGIES constructs events, logical scenarios, and executable test cases. Without trustworthy SSD, scenario-based testing cannot achieve safety assurance. Four complementary paradigms form the basis of modern SSD practice:

i. Data-driven datasets from vehicles, infrastructure, and aerial sensing
ii. Crash and incident reconstructions
iii. Knowledge-based assets such as ontologies and standards
iv. AI-augmented sources leveraging generative models

A unifying requirement across all paradigms is traceability and interoperability. In SYNERGIES, SSD is provided using a common format, OMEGA-PRIME, which itself largely relies on existing standards, namely OpenDRIVE [10] for static road geometry and Open Simulation Interface [11] ground-truth messages for road users' trajectories, packaged in an .mcap file.

Additionally, other standards are used for other categories of data, such as OpenSCENARIO [1] for executable behaviour, and OpenLABEL [12] for tagging and metadata. Operational Design Domain (ODD) attributes—defined in ISO 34503, PAS 1883, and ASAM OpenODD [13] —act as the filtering and categorization backbone.

Data-driven SSD

Empirical corpora form the foundation of most scenario libraries. Multimodal on-board datasets such as KITTI, nuScenes, Argoverse-2, and Waymo Open Motion provide synchronized streams of video, LiDAR, radar, and vehicle state data aligned with high-definition maps. These corpora allow researchers to mine naturalistic interactions, for instance, yielding cut-in events where a merging driver forces an automated vehicle to brake, or sequences where occlusions challenge perception.

In addition to data collected from sensors on vehicles, valuable scenario information can be obtained from cameras placed on roadside infrastructure or from drones observing traffic from above. This aerial view makes it possible to record the movement of every vehicle with minimal visual obstruction, given favourable weather conditions. For example, the **highD** [14] dataset captures highway traffic in Germany, while the **inD** [15] dataset focuses on urban intersections. Both provide detailed, natural driving behaviour that can be analysed to understand how vehicles interact in real situations. The **INTERACTION** [16] dataset extends this approach internationally and includes detailed map information using the Lanelet format, allowing researchers to link vehicle movements to road layouts and traffic rules. Such aerial datasets are especially useful for studying how drivers negotiate space with one another, for example, how they decide when to overtake or cross at intersections, which helps build realistic and human-like scenarios for automated-vehicle testing.

A practical challenge lies in map normalization. Many datasets are released in proprietary formats (e.g., Lanelet2 [17] or bespoke HD maps). To integrate them into the SYNERGIES CDF, conversion pipelines (Lanelet2 ↔ OpenDRIVE) are required. Such conversions are imperfect: lane connectivity or traffic-rule fidelity may degrade, necessitating quality assurance gates. For example, converting an Argoverse intersection to OpenDRIVE may produce missing priority rules, which must be checked against local traffic codes before using them.

Crash/incident SSD

Real-world collisions, while tragic, provide important edge cases for scenario libraries. Assuming the required information on the crashes can be retrieved at sufficient accuracy and reliability, occurrences of crashes and their characteristics can be *directly* catalogued as parametrized scenarios. The scenarios can be stored at multiple description levels (abstract, logical, concrete).

Crashes can also be entirely reconstructed in simulation using tools such as PC-Crash, which allows validating the plausibility of the reconstruction. Such reconstructions are now commonly shared in association with accident databases (for instance, as Pre-Crash Matrices (PCM), as constructed in GIDAS or through the IGLAD consortium). These reconstructions encode trajectories, velocities, and contact geometries in the seconds leading to impact. This corresponds to the same kind of data that can be represented (and hence converted to) the SYNERGIES CDF. These can either be used as specific datasets that can be used in the same way as any SSD, to generate new parametrized scenarios, but an additional benefit is that they can be replayed in simulation to directly assess whether a specific function could have mitigated or avoided the crash. In this way, crash-derived SSD directly feeds into efficacy studies for safety functions and ensures that safety cases cover historically validated hazards.

Knowledge-based SSD

Not all scenarios arise from empirical data; many are defined by rules, standards, or ontologies. These sources provide scaffolding that ensures consistency, interoperability, and coverage.

- Standards as anchors. ISO 34502 [18] sets out a framework for scenario-based safety evaluation, while ISO 34503 [19] and PAS 1883 define hierarchical ODD taxonomies (scenery, environment, dynamic elements). ASAM OpenODD translates these into machine-readable forms.
- Executable scenario formats. OpenSCENARIO 2.0 DSL [20] allows logical scenarios with parameter ranges, constraints, and coverage goals; its XML 1.x counterpart [1] instantiates concrete test cases. OpenDRIVE encodes road networks, and OSI defines ground-truth and sensor-view interfaces. OpenLABEL [12] standardizes metadata and tagging, enabling linkage of ODD attributes, behaviours, and evaluation metrics.
- Ontologies. SAE J3164 provides a lexicon of manoeuvres and behaviours, and OpenXOntology [21] ensures semantic consistency across the ASAM OpenX family [9].

To illustrate, a functional scenario such as *"vehicle approaching a pedestrian crossing in rain at night"* can be codified as a logical scenario in OSC2 DSL (with parameters for pedestrian speed, rainfall intensity, and road lighting), constrained by ISO ODD definitions, and stored with OpenLABEL tags (weather: rain, lighting: low, actor: pedestrian).

Community repositories such as Safety Pool™ further support scenario reuse by allowing regulators, developers, and researchers to exchange standardized, ODD-tagged scenarios.

AI/LLM-augmented SSD

A rapidly emerging frontier in scenario generation is the use of large language models (LLMs) and other foundation models to transform unstructured information into simulation-ready assets. Unlike traditional pipelines, which rely on manual authoring or labour-intensive mining of logged data, LLMs can ingest heterogeneous sources—accident reports, textual requirements, natural language prompts, even images or ODD specifications—and output executable scenarios in formats such as OpenSCENARIO. This capability makes them particularly valuable for filling gaps in coverage, accelerating scenario authoring, and exploring rare or adversarial events that are difficult to capture empirically.

Several research efforts demonstrate the breadth of these approaches. *Text2Scenario* and its derivatives show that free-form descriptions can be translated directly into OpenSCENARIO representations, substantially reducing the authoring overhead for engineers. Building on this, *Txt2Sce* [22] extends the pipeline by taking a single textual accident report, extracting its semantic structure, and then generating whole families of derived scenarios through systematic variation. In a different vein, *Chat2Scenario* bridges human intent with real datasets: a user specifies a functional description such as "a cut-in on a wet motorway at night," the system retrieves matching episodes from naturalistic logs, and exports them as executable scenarios, thereby combining the fidelity of real data with the flexibility of LLM interpretation. More recently, multi-modal approaches such as *Traffic Composer* [23] combine textual descriptions with contextual imagery, by extracting an *intermediate representation* (IR) from both modalities (textual IR and visual IR) before generating executable scenarios, to produce richly parameterized, map-specific scenarios in CARLA and other simulators, while ODD-to-simulation pipelines demonstrate that entire scenario suites can be generated directly from formal ODD specifications (see Figure 4). In parallel, adversarial methods use LLM reasoning to push scenarios beyond the nominal distribution of datasets—introducing, for example, unsafe pedestrian behaviours or extreme traffic interactions—to probe the limits of automated driving functions.

The promise of these methods is matched by their risks. Without safeguards, generated scenarios may contain physically impossible dynamics, violate traffic laws, or fall outside the target ODD. SYNERGIES therefore can treat AI-augmented SSD as a **controlled pipeline** rather than a free-form generator. Each scenario must first be aligned with ODD tags expressed in OpenODD or PAS 1883, ensuring that outputs are regionally valid and traceable. Generated trajectories and behaviours are automatically checked against physical limits, map connectivity, and traffic rules before being accepted. Scenario representations are preferred in OpenSCENARIO, which allows parameter ranges, constraints, and coverage goals to be formally declared; only then are concrete XML instantiations exported for execution. Provenance metadata is recorded at every step—including the original text, prompt, dataset references, and model version—so that AI-derived scenarios remain auditable and compatible with licensing requirements.

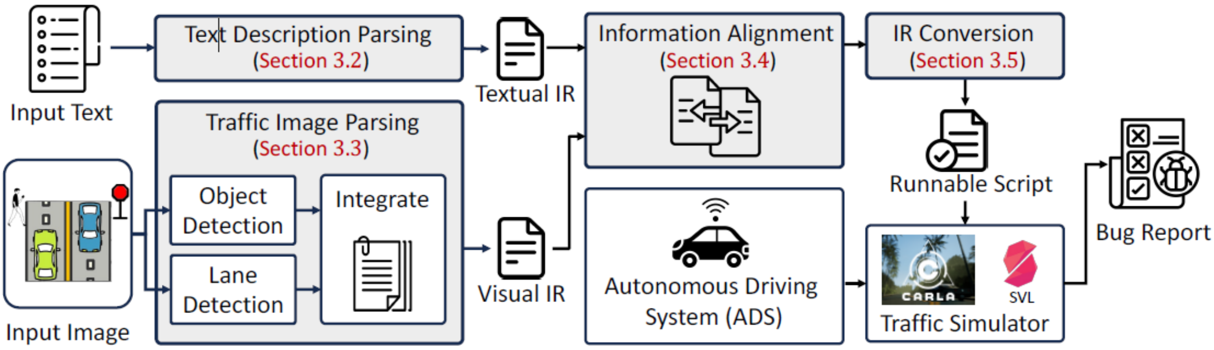


Figure 4: Overview of Traffic Composer [24], IR stands for Intermediate Representation

By integrating such pipelines, SYNERGIES leverages the productivity of LLMs while enforcing rigorous safeguards. The outcome is a class of scenario sources that are fast to produce, diverse in coverage, traceable, and defensible within a safety-assurance argument. Far from replacing empirical or knowledge-based sources, AI-augmented SSD complements them by populating the long tail of rare, adversarial, or region-specific events that would otherwise remain absent from the scenario library.

While AI- and LLM-based approaches significantly enhance productivity and coverage in scenario generation, their use introduces specific risks that must be addressed to ensure safety, validity, and regulatory acceptability. Within the SYNERGIES methodology, the following mitigation principles are applied:

- **Hallucination and plausibility risks**. Automatically generated scenarios may include physically implausible dynamics or unrealistic behaviours. Mitigation is achieved through constraint checking against physical limits, traffic rules, and ontology-based consistency checks prior to acceptance.
- **ODD and legal validity**. Generated scenarios may inadvertently violate regional traffic regulations or fall outside the intended Operational Design Domain. All AI-derived scenarios are therefore filtered and tagged using formal ODD definitions (e.g. OpenODD, ISO 34503, PAS 1883) to ensure contextual validity.
- **Traceability and auditability**. To support safety arguments and regulatory review, full provenance metadata is recorded, including source prompts, datasets, model versions, and post-processing steps. This ensures transparency and reproducibility of AI-generated scenarios.
- **Quality assurance and acceptance control**. AI-generated scenarios are treated as candidate Scenario Source Data and must pass the same validation, enrichment, and

integration steps as empirically derived or knowledge-based scenarios before being used in testing.

By embedding AI-based generation within a controlled, standards-aligned workflow, the SYNERGIES methodology leverages the strengths of foundation models while maintaining the level of rigor required for safety-critical CCAM validation.

3.5 Scenario concept

As outlined in Sections 3.1 and 3.4, the methodology developed in WP5 is built upon a unified driving scenario concept as a formalised framework for describing scenarios across different abstraction levels: functional, abstract, logical, and concrete. According to [25], a driving scenario concept describes the following aspects:

1. Which driving scenarios (scenario categories) exist?
2. How are these driving scenarios defined?
3. How are these driving scenarios related?

A review of existing scenario concepts, including the afore mentioned work as well as other related projects and database initiatives, is given in [27]. The scenario concept of SUNRISE is described in [28]. Reviewing these references revealed that no single "ideal" concept exists. Instead, there is a wide variety of scenario concepts developed for different purposes. Developing a new one from scratch that fits all purposes was considered unrealistic and error-prone. Instead, the work package partners aimed to minimize repeated work by agreeing on parts of the existing scenario concepts that satisfy relevant requirements in Table 2Table 2: Requirements on scenario methodology and the objectives described in the following paragraphs. As a result, the work package partners concluded that the scope of WP5 could be most effectively addressed by a scenario concept that retains concepts from MOSAR [29] and CRUISE [25], see Figure 6.

The rest of this section is structured as follows. The following paragraphs describe the role of the proposed scenario concept, the motivations behind its selection, and an overview of its structure. These motivations are structured along the topics of interoperability, suitability in WP5, and maturity, structure and compliance with existing standards. The underlying ontology, its role and relevant standards are described in Subsection 3.5.1. Subsection 3.5.2 describes the definition of logical and abstract scenarios (see requirement #5 in Table 2) including further details about MOSAR and CRUISE.

Role of the proposed scenario concept

Before justifying the selected building blocks, the role of the scenario concept in WP5 needs to be clarified. The scenario concept in WP5 is shall not be confused with the scenario concept of the SYNERGIES platform (which is, e.g., used for scenario search). The scenario concept in WP5 (as proposed in this deliverable) is specifically tailored to the methodology depicted in Figure 2 and to the WP5 topics described the following sections, namely scenario identification, extraction, and enrichment in Section 3.6, scenario generation in Section 3.7 and Scenarios of Tomorrow in Section 3.8. The scenario concept of the SYNERGIES platform on the other hand, is anticipated to be a further development of the SUNRISE scenario concept [28], however, it is not fully specified at the time of writing this deliverable. These two possibly different scenario concepts must nonetheless be closely related to be useful in practise. As a minimum requirement, the scenario concept in WP5 must be compatible with the scenario concept of the SYNERGIES platform. Since the scenario concept of the SYNERGIES platform is expected to build upon the one in SUNRISE, and since SUNRISE developed its scenario concept to enable interoperability between the ADScene database (using MOSAR) and scenario.center (using CRUISE) via the SUNRISE Data Framework [30], this requirement is not expected to cause significant complications. Nonetheless, the WP5 team should be informed if the proposed scenario concept turns out to be incompatible with or contradictory to any project tasks in other work packages. As the opposite extreme, the scenario concept in WP5 may be fully integrated into the scenario concept of the SYNERGIES platform. This ideal case is encouraged in this deliverable because it would eliminate compatibility issues between the two concepts and allow scenarios in the SYNERGIES dataspace to be searchable using a common ontology.

Interoperability within the methodology

Within WP5, the primary goal of agreeing on a shared scenario description is to enhance interoperability and collaboration through unified definitions of the concepts and terminology used in the methodology. By promoting a common understanding of relevant terms and concepts among all partners, this shared scenario concept reduces the risk of misunderstandings and facilitates effective collaboration. In the context of WP5, the particular emphasis is placed on the interoperability of workflows, which are represented as blue boxes in Figure 2 and are interconnected via directional pathways. Figure 5 illustrates a minimal example of such interconnected workflows from the whole methodology.

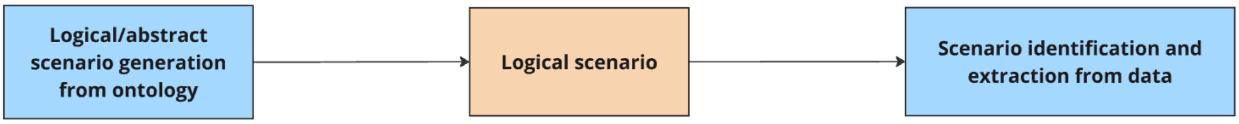


Figure 5: Minimal example of connected workflows from Figure 2

Here, the workflows "Logical/Abstract Scenarios Generation from Ontology" and "Scenario Identification & Extraction from Data" are linked via the "Logical Scenario" interface (shown as an orange box in the figure), which is an output of the former workflow and input to the latter. Ideally, the scenario concept enables the standardization of all interfaces in Figure 2, eliminating the need for data conversion between workflows. This would allow specific tasks (e.g. sequences of workflows) to be realized using multiple combinations of tools from the Marketplace without requiring implementation-specific adaptations, see requirement #12 in Table 2Table 2: Requirements on scenario methodology. Since the methodology includes scenario identification and scenario generation, this objective also addresses requirement #4. In addition, future platform users, particularly those in provider and user roles for scenarios and tools as detailed in Table 4 of Deliverable 2.2 [31]), are expected to benefit from a shared scenario concept. It enables scenario and tool providers to specify their content and outputs in a standardized manner, allowing scenario and tool users to efficiently search for and identify the assets for their specific requirements.

It should be noted that the driving-scenario concept proposed here does not exclude the use of tools for scenario identification, extraction, or generation that use a different underlying scenario concept from SYNERGIES. For example, output scenarios produced by a scenario generation tool(chain) developed for a specific scenario database can still be shared via the SYNERGIES dataspace, provided they are compliant with the format and requirements of the platform. Moreover, standardized interfaces facilitate the development of data-conversion mechanisms, such that tools developed in WP5 can be combined with other existing external tools. In particular, if an existing framework covers a broader set of scenarios than the proposed concept, or if an uncovered scenario is detected (e.g. using the method of Section 3.6.4) contributions to extend the proposed scenario concept and close such gaps are encouraged.

Suitability for the topics of WP5

The second objective is to establish a scenario concept that is tailored towards the topics addressed by the tasks in the WP5, namely scenario identification and extraction, scenario generation, and Scenarios of Tomorrow. Initial work in the other WP5 tasks revealed that several partners prefer a declarative framework for describing scenarios, which ideally allows complex scenarios to be decomposed into simple building blocks and simple scenarios to be composed into more complex ones in a structured way. A declarative scenario description enables a high-level view of the scenario content without going into fine-grained details. This facilitates human understanding (see requirement #8 in Table 2) on an abstract (or logical) level and is well-suited for scenario identification, because it enables searching for instances that match predefined patterns in the static and dynamic parts of datasets. This approach avoids the need to develop dedicated code or tool from scratch for detecting each new complex scenario. In addition, it facilitates identifying scenarios that share similar building blocks which helps to prevent the regeneration of existing scenarios, see requirement #17 in Table 2. The composition of simple scenarios into complicates ones is also well-suited for scenario generation, however, additional considerations may be necessary to obtain a realistic and meaningful scenarios. This topic is addressed in [32] for example, see also the description of a parametrization of a composed scenario in Section 3.5.2. Consequently, this objective is in line with requirement #4 in Table 2.

Maturity, structure and compliance with existing standards

The third objective is to setup a mature, yet extensible scenario concept that complies to existing standards and employs a widely used underlying structure. By building upon mature concepts that have proven their applicability, e.g., in previous projects or database initiatives, we aim to reduce sources for errors such as inconsistencies, ambiguities etc. Additionally, if the scenario concept complies to existing standards and employs a widely used underlying structure, most stakeholders will already be used to the majority of the underlaying concepts and definitions which may foster the acceptance of the proposed scenario concept and facilitate its application in a certain task. More explicitly, this objective can be summarised as follows: *The scenario concept should rely:*

- *on an ontology to describe infrastructure and road users' interactions. This ontology may be changed and/or evolve over time,*
- *as much as possible on proven concepts (i.e., concepts that have been successfully implemented in previous projects),*
- *as much as possible on existing standards, see also Section 3.1.8 in Deliverable 2.1* [33].

Overview of the scenario concept in WP5

As mentioned at the beginning of this section, the proposed scenario concept in WP5 reuses concepts of MOSAR and CRUISE. This selection is not the only possibility to cover the tasks of the work package. To motivate the selection of CRUISE, notice that it was developed in VVM project for similar purposes as in this work package and its approach to describe the dynamic part of scenarios in terms of pre-defined concepts and base scenarios (see Figure 14) enables decomposing given scenarios into simpler building blocks and creating complicated scenarios as compositions of simpler building blocks (see [32]). It was considered a mature scenario concept that fulfils the second and third objective above (see also subsections 3.5.1 and 3.5.2). Another candidate would have been the closely related scenario concept of the EC-funded project Hi-Drive [34] which is also based on 6-layer model by [35]. However, CRUISE is used in scenario.center which was connected to the SUNRISE Data Framework [30]. Thus, by the paragraph on the role of the scenario concept in WP5 above, it will likely be compatible with the scenario concept of the SYNERGIES platform. This point and the framework for setting up parametrizations (see Section 3.5.2) were considered advantages over the scenario concept of Hi-Drive. Consequently, CRUISE was selected for the description of the dynamic part (layer 4 of the 6-layer model) of scenarios. Regarding the static part of scenarios (layer 1, 2, and 3) and environmental conditions (layer 5), the approach of MOSAR, see Section 3.5.2, seemed to be better suited to the tasks of the work package. It is a mature concept that is used in the industrial scenario library ADScene which was also connected to the SUNRISE Data Framework [30]. It provides a declarative approach that relies on an extensible ontology, and its visual approach facilitates human understanding, see Section 3.5.2. Layer 6 (Digital Information) is exceptional and not specified in detail yet. The work package partners agreed that setting up a scenario concept for this layer (especially for communication) is difficult without a specific purpose. In contrast to predecessor projects like HEADSTART and SUNRISE, such a purpose is not detailed within SYNERGIES at the time of writing this deliverable. Thus, the details of layer 6 in the scenario concept of WP5 are left open for future work. Notice, that the SSD format omega-prime [36] supports digital information such as V2X. Thus, the scenario concept can be extended to describe layer 6 based on the methodology in Figure 2.

In summary, the scenario concept in WP5 uses a combination of concepts from MOSAR and CRUISE to describe scenarios along five layers of the 6-layer model: MOSAR for the static part and environmental conditions (layers 1, 2, 3, and 5), and CRUISE for the dynamic part (layer 4), see Figure 6. Notice that the base scenarios in CRUISE are specified without reference to a specific infrastructure. Thereby, a certain based scenario can be mapped on any part of the infrastructure where the base scenario can possibly be performed. For example, a vehicle approaching a traffic jam can be detected in any lane of a multi-lane highway. To achieve full compatibility between the parts from MOSAR and CRUISE, this mapping should be specified precisely. However, defining such a procedure this is still work in progress so that the mapping between base scenarios and infrastructure is done inside the tools.

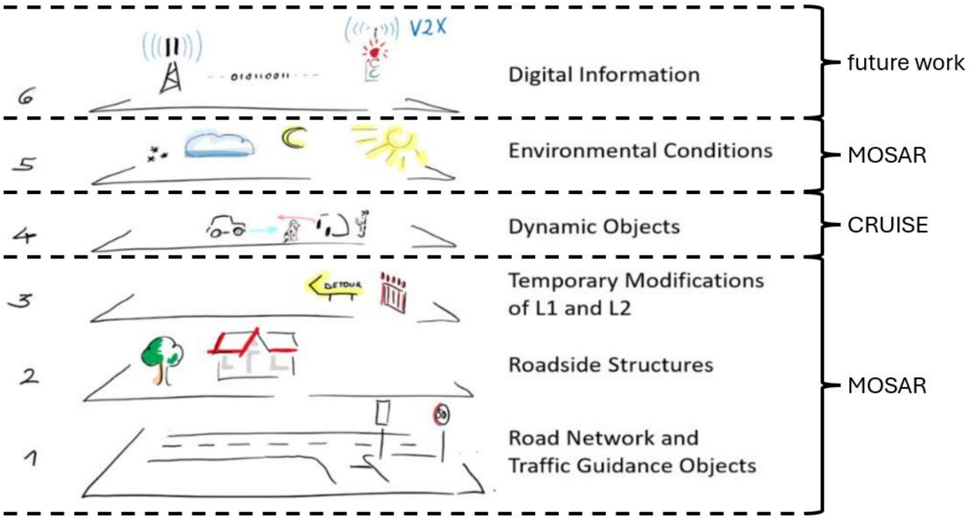


Figure 6: Partition of the 6-Layers Model in [25] in the scenario concept of WP5 (adjusted image from [32])

To improve clarity and accessibility for non-expert readers and reviewers, Table X provides an overview of how the different scenario abstraction levels used in the SYNERGIES methodology relate to typical stakeholders and tools. This mapping illustrates how each abstraction level supports different roles within the CCAM ecosystem while remaining part of a coherent, interoperable workflow.

Table 3 Mapping of scenario abstraction levels to users and tools

Scenario abstraction level	Purpose in methodology	Typical users	Typical tools / representations
Functional scenario	High-level description of traffic situations and safety-relevant use cases	Policy makers, regulators, system architects, researchers	Natural language descriptions, safety concepts, use-case definitions, ODD descriptions
Abstract scenario	Formalised, qualitative definition of scenario families and interactions	Researchers, method developers, safety engineers	Ontology-based descriptions, MOSAR concepts, CRUISE base scenarios, graphical scenario editors
Logical scenario	Parameterised, machine-readable scenario definitions enabling systematic variation	Scenario engineers, tool developers, data scientists	ASAM OpenSCENARIO 2.0 (DSL), logical constraints, parameter distributions, semantic models
Concrete scenario	Fully instantiated scenarios with fixed parameter values	Simulation engineers, validation engineers	ASAM OpenSCENARIO XML, OpenDRIVE, OSI, simulator-specific formats
Test scenario	Executable scenario used for testing a specific system under test	OEMs, Tier-1s, test engineers, technical services	Simulation environments, HiL/SiL toolchains, KPI evaluation tools

3.5.1 Ontology

Ontologies provide a structured representation of knowledge within a specific domain by defining concepts, their attributes, and the relationships between them. They serve as a unifying framework to describe road users, their manoeuvres, the infrastructure, and the logical relationships between them. Ontologies are designed to be machine-interpretable and support reasoning, making them suitable for systematic scenario definition, identification, and generation. They enable consistency across scenario description levels and facilitate scenario generation for the verification and validation of automated functions.

The Association for Standardisation of Automation and Measuring Systems (ASAM) has advanced this approach through ASAM OpenXOntology which provides fundamental definitions, properties, and relationships of concepts relevant to Automated Driving Systems (ADSs). Such ontologies underpin interoperability of terms and concepts across scenario description levels.

The ontology serves three main purposes within the scenario methodology framework:

1. A shared semantic framework: It ensures that the same terms are used consistently across scenario abstraction levels (abstract, logical, concrete).
2. Scenario building block: It defines the actors, infrastructure, manoeuvres, and environmental properties that are relevant to constructing scenarios. Any scenario (whether extracted from data or generated synthetically) must be expressible within the ontology.
3. Evolving knowledge model: It evolves over time as new types of scenarios emerge that cannot be described with the existing ontology, e.g., when new actors or rare traffic events are discovered.

As shown in Figure 6, the ontology developed within SYNERGIES is designed to capture the layers of the well-established 6-layer model for the scenario concept. This ontology, anchored to existing standards and taxonomies (e.g. ISO 3450x, BSI Flex 1891), can be mapped to the 6-layer model. Figure 7 (i) shows the hierarchical taxonomy as illustrated in ISO 34503 [19], for specifying operating conditions that enable the definition of an ODD of an ADS. Figure 7 (ii) shows the hierarchical taxonomy of behaviours as specified in BSI FLEX 1891 [38] for an ADS and other road actors.

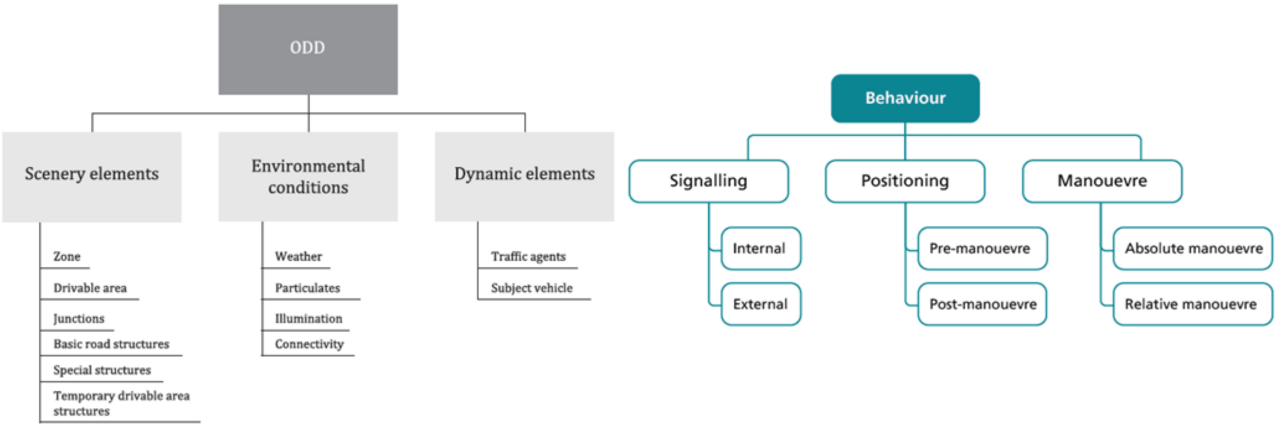


Figure 7: (i) Top level taxonomy with ODD attributes [19] (ii) Hierarchy of behaviour classifications [38]

Figure 8 is an extract from the ASAM OpenXOntology concept. The example illustrates how scenario attributes and their hierarchical relationships are formally defined using the OWL (Web Ontology Language). Such a structured representation enables the ontology to describe the complete scenario space in a machine-interpretable and semantically consistent manner.

```
@prefix : <http://ontology.asam.net/ontologies/OpenXOntology.owl#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@base <http://ontology.asam.net/ontologies/OpenXOntology.owl> .

<http://ontology.asam.net/ontologies/OpenXOntology.owl> rdf:type owl:Ontology ;
                                                         owl:versionIRI <http://ontology.asam.net/ontologies/OpenXOntology.owl/1.0.0> ;
                                                         rdfs:comment "Parent commit (prerelease info only!): 1e655fa506e00981628317cf4f248235f8f38863" ;
                                                         rdfs:label "ASAM OpenXOntology" ;
                                                         owl:versionInfo "Version 1.0.0" ,
                                                                         "license: This ontology is the copyrighted property of ASAM e.V. Any use is limited to the
                                                                         scope described in the license terms (https://www.asam.net/license). In alteration to the
                                                                         regular license terms, ASAM allows unrestricted distribution of this standard. §2 (1) of
                                                                         ASAM’s regular license terms is therefore substituted by the following clause: 'The licensor
                                                                         grants everyone a basic, non-exclusive and unlimited license to use the standard ASAM
                                                                         OpenXOntology.'" .


###  http://swrl.stanford.edu/ontologies/3.3/swrla.owl#isRuleEnabled
<http://swrl.stanford.edu/ontologies/3.3/swrla.owl#isRuleEnabled> rdf:type owl:AnnotationProperty .

###  http://ontology.asam.net/ontologies/Core#aggregatedInto
<http://ontology.asam.net/ontologies/Core#aggregatedInto> rdf:type owl:ObjectProperty ,
                                                                    owl:AsymmetricProperty ;
                                                           rdfs:domain <http://ontology.asam.net/ontologies/Core#SpatioTemporalExtent> ;
                                                           rdfs:range <http://ontology.asam.net/ontologies/Core#SpatioTemporalExtent> ;
                                                           rdfs:comment "DEF: A relationship type where a SpatioTemporalExtent may be aggregated into one or more
                                                           others. This object property has the same meaning as the class Aggregation, but a different representation." ,
                                                                        "HQDM aggregated_into" .


###  http://ontology.asam.net/ontologies/Core#appliesTo
<http://ontology.asam.net/ontologies/Core#appliesTo> rdf:type owl:ObjectProperty ;
                                                      rdfs:comment "DEF: This relation is used to describe that a specification or regularity applies to a particular
                                                      object. For example, this relation can be used to describe which lanes a speed limit sign applies to." .
```

Figure 8: Snippet of ASAM OpenXOntology (ASAM)

Beyond static classes and properties, an ontology can support logical rules (for example, in SWRL – the Semantic Web Rule Language) to express scenario constraints, safety conditions, or regulatory rules. Rules written in SWRL are used in conjunction with the ontology to automatically reason about and infer relationships among scenario objects [39]. For example, consider a rule derived from the UK Highway Code:

“If a vehicle is in a lane which has a solid line as the right-side marking, then the vehicle must not change lane right and it must not turn right.”

This rule can be enforced within an ontology as follows:

car(c) ∧ laneType(l) ∧ solidLine(sl) ∧ isOn(c,l) ∧ turnRight(tr) ∧ changeLaneRight(lcr) ∧ hasRightLaneMarking(l,sl) → mustNot(c,lcr) ∧ mustNot(c,tr)

Here, each *predicate* in the rule, such as car(c), laneType(l), solidLine(sl), changeLaneRight(lcr), and turnRight(tr), imposes a *type* constraint, ensuring that the variables used as parameters correspond to instances of the respective *classes* defined in the ontology. For instance, the predicate car(c) requires that variable c must be an instance of the class “car”. When all constituent parts of the scenario exist, the predicates hasRightLaneMarking(l,sl) and isOn(c,l) are used to establish the relationships between the variables. These relationships define the scenario condition.

The safety condition expressed in the rule states that the car c must not perform either a lane change right or a turn right manoeuvre. In this formulation, the scenario’s ODD conditions (e.g., lane type, markings, actor type) determine the applicable safety constraint (namely, that the car should not perform an unsafe manoeuvre).

Ontology tools such as Protégé support the use of SWRL rules to augment the domain ontology. Reasoning engines such as Pellet and HermiT are then used to reason about the ontology in the context of the rules. Given the ontology, representing knowledge about the domain, along with the rules (expressing safety or performance requirements), it then becomes possible to use an automated reasoner to infer a logical scenario.

By capturing the various layers of a scenario concept, the ontology establishes a comprehensive representation of the domain. Scenarios can then be instantiated from this ontology by combining static scenery, dynamic actors, and environmental parameters into logical structures, which can be further parameterized and transformed into scenarios.

3.5.2 Logical/abstract scenario definition

The SYNERGIES methodology relies heavily on logical and abstract scenarios, which serve as a crucial bridge between raw or annotated traffic data and the creation of detailed, simulation-ready test scenarios. These scenarios provide the high-level specifications that describe traffic situations in an abstracted, systematic form, enabling the aggregation of observed cases and the formalization of expert knowledge while supporting the diversity found in real-world traffic.

Abstract scenarios offer a high-level, conceptual representation of families of traffic situations. They outline the involved entities, their spatial and temporal relationships, and the behavioural principles that characterize the situation. They establish the qualitative criteria used to determine whether an observed situation corresponds to a concrete instance of that scenario type. For example, an abstract scenario may describe a situation where "a vehicle enters the lane ahead of the ego vehicle from an adjacent lane," without prescribing specific parameter values such as the precise path, speed, or timing of the manoeuvre. This approach captures the fundamental meaning and purpose of the interaction, allowing for a broad spectrum of possible implementations.

Logical scenarios expand on this framework by incorporating parameterization. They define an abstract structure enriched with parameters that can be fixed or variable. Fixed parameters represent unchanging elements, such as road layout or the number and roles of actors involved. Variable parameters, on the other hand, are defined by ranges or (multivariate) probability distributions, encompassing factors like initial speed, vehicle spacing, weather, or traffic volume. In some instances, parameters may be correlated to reflect real-world conditions (e.g. due to mathematical constraints or causal relations), such as adjusting headway based on speed. Mathematically, a logical scenario establishes a (constrained) multi-dimensional parameter space, enabling the generation of a large variety of concrete scenarios.

Positioned at this stage, this combined framework formalizes and generalizes traffic situations, ensuring the resulting definitions are reusable in different contexts, interoperable across systems, and suitable for systematic analysis and extraction. In the SYNERGIES context, logical scenarios provide the unambiguous machine-readable definition of a scenario by defining the parameter space of the actors and their interactions within a structured environment (road information and weather information). Abstract scenarios support this process by defining the qualitative families of situations that guide logical scenario construction and scenario extraction. Because an abstract scenario is not fixed to a single sequence of events and actions, but allows a wide variety of them, abstract scenarios have a lower specification complexity than logical scenarios [40]. Hence, abstract scenarios allow for a more compact description of the scenario space.

There are two approaches to define and create such scenarios:

1. **"Bottom-Up"**: Abstract and logical scenario definition is based on many sources:
    a. Expert Knowledge about road interactions
    b. New Observation/Uncategorized situations, which extend the list of known logical scenarios when no corresponding logical scenario was available

    c. Non-Ordinary situations using data mining approaches, near misses or other rare situations that are observed and categorized (logical scenarios only)
    d. Additional scenario source data (e.g. NDS, instrumented vehicles, road-side observations, accident reconstruction, Generative AI, simulation) may be used to enrich parameter ranges and distributions for logical scenarios (e.g. using methods like hybrid graphs, see Glasmacher et al. 2023) and to identify potential new abstract scenario families.

2. **"Top-Down"**: An ontology can define the qualitative structure and constraints of scenarios, enabling the creation of logical scenarios and, in some frameworks, abstract scenario classes. Empirical data can then be used to assign parameter values or probabilistic distributions.

The abstract / logical scenario describes a situation where the initial state of the main actors is known (position and dynamics), and one or more events or actions occur, which define the triggering conditions of these events/actions. Properties that are necessary to define a scenario include:

- The origin of the scenario (derived from empirical data, expert knowledge, or an ontology) as an important metadata that informs the parameterization and integration of logical scenarios
- The infrastructure where the scenario takes place should be described, including road length, type, width, curvature, and other relevant properties
- Actors can have physical properties, types, and dynamic capabilities
- The sequence of events and actions, sometimes referred to as a storyboard, describes the kinematics, relations, and other specific parameters relevant to each action
- The environment, which can include weather conditions, lighting condition, road conditions, and traffic

Scenario structure is visualized in Figure 9. The elements of the scenario (Infrastructure, Actors, etc.) are based on an external ontology that can be enriched.

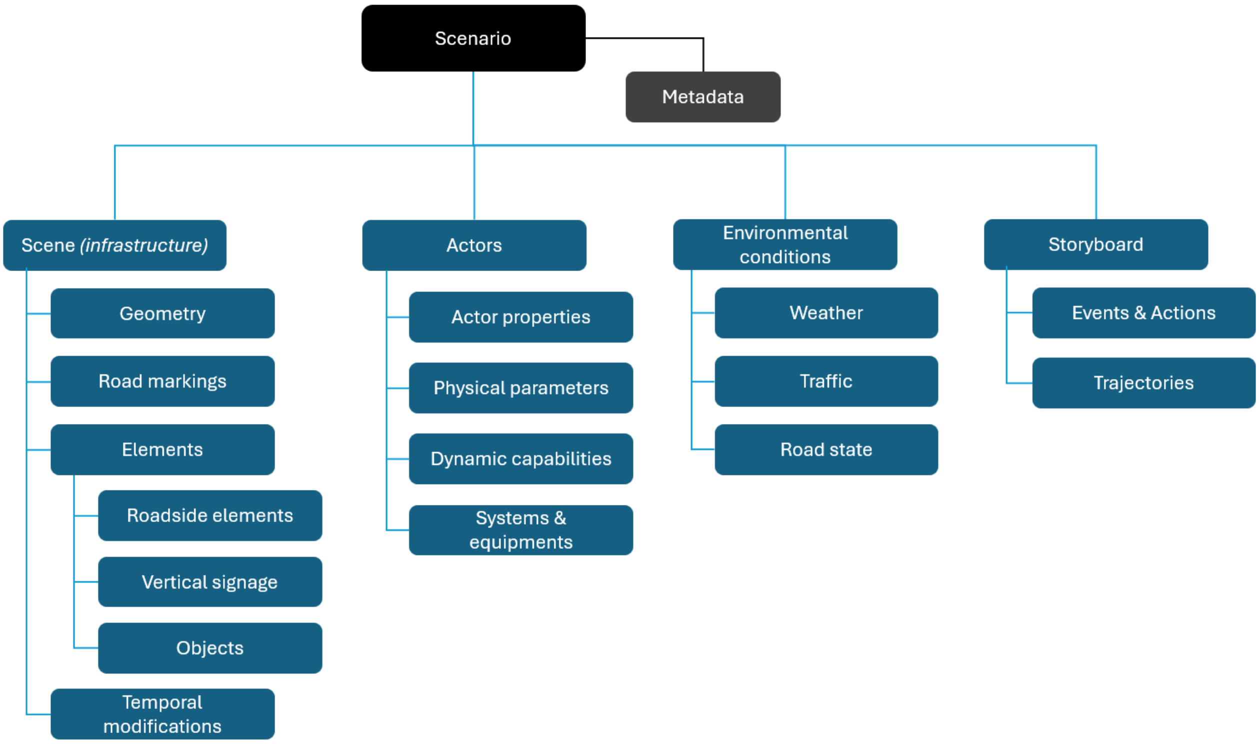
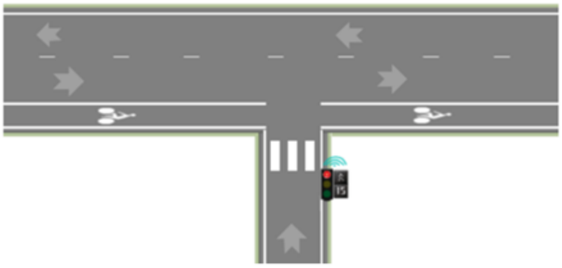


Figure 9: Scenario structure

Another important aspect of the abstract and logical scenario definitions is the way their parameters are represented. Abstract scenarios' parameters are represented at a higher level, using qualitative or symbolic values derived from a defined ontology, whereas logical scenarios must include parameters that can vary. Logical scenarios' parameters are represented in more detail, introducing ranges of values, (multivariate) distributions, relations or constraints between parameters, and concrete values where appropriate.

Infrastructure

The infrastructure class defines the unchanging environment where a scenario takes place, referring to both the road layout and any road furniture placed on it. For example, one infrastructure description is shown in Figure 10.



Bidirectional main road with two traffic lanes and a cycle lane on the right. A T-intersection connects to a one-way road with a single traffic lane. A connected traffic light with a pedestrian crossing signal controls a pedestrian crossing near the intersection.

Figure 10: Infrastructure description example

This concept is designed for a simple, high-level overview of the terrain and roads with two main goals. It enables the placing of actors in specific areas and defines important features that might impact the scenarios that occur there.

The infrastructure is defined as follows:

- The infrastructure class starts with basic information like a name, description, and an image

- The road is broken down into *Road Segments*, which are consistent parts of the road. Each new change in road conditions (like a shift in geometry or signage) marks the start of a new *Road Segment*
- These Road Segments are further divided into Strips, showing how each segment is structured along its length
- Static objects, such as parts of road infrastructure and signs, are placed within this layout

Following this logic, an infrastructure representing a road with separate carriageways and an exit would be broken down as follows (see Figure 11):

1. The road would be segmented into three main parts:
    a. The first segment would represent the current section of the road with separate carriageways
    b. The second segment would represent the exit
    c. A third segment might be necessary to depict the ramp

   Each segment corresponds to a distinct portion of the road, reflecting changes in design or function, such as moving from a main road to an exit or ramp.

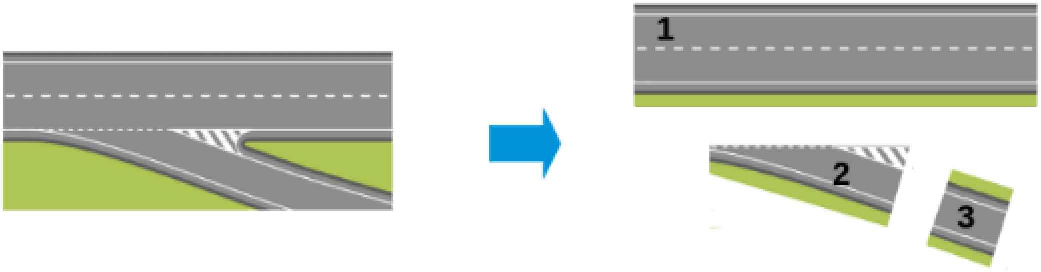


Figure 11: Infrastructure composition

2. The segment of the current section would be divided into several strips, as shown in Figure 12:
    a. One for the road separation area, one for the traffic lanes, one for the emergency area, and one to represent the edge of the pavement
    b. The exit segment would be broken into multiple strips, one on the right edge of the road with a zebra crossing, one for the main lane, and another for the left edge of the road

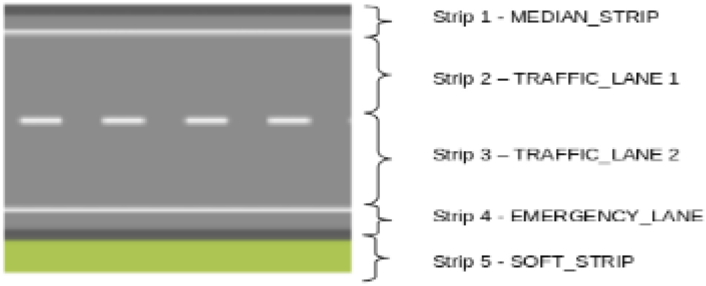


Figure 12: Infrastructure decomposition

Infrastructure elements include all objects placed on the infrastructure, represented through a flexible data structure capable of describing many element types:

- Road equipment (barriers, gantries, metal barriers, etc.)
- Objects (cones, trees, road obstacles)
- Temporary features (potholes or oil stains)

Because these elements are very different from each other, the data model only includes basic details for each one, such as a label, description, image, and size, using the infrastructure elements class.

Each created segment can be joined by specifying a contact point between the segments, as shown in Figure 13 (side [segment side: START; END; LEFT; RIGHT], position [START; CENTER; END], and the angle of segment between both [ROW; T; WIDE_Y; SHARP_Y]).

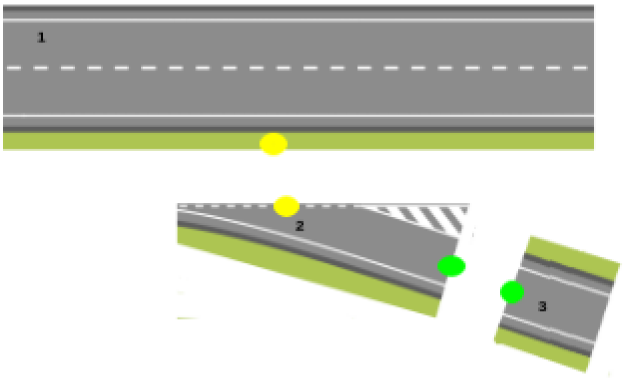


Figure 13: Road segments junction. The yellow and green circles represent contact points between the road segments.

Actors

Actors represent all the dynamic entities in the scenario, including the ego vehicle, other vehicles, cyclists, pedestrians, and animals. Because these actors can differ significantly, the actor class includes only the basic information: name, type, description, image, and how the actor appears in scene vignettes.

Additional optional attribute groups can be specified to describe actors in more detail:

- Physical parameters: characteristics such as size, weight, or colour
- Dynamic capabilities: properties like maximum acceleration or maximum speed
- Vehicle capabilities: vehicle-specific traits like maximum seating capacity
- Occupants: information about the driver or number of passengers
- Transportation usage: the intended or actual usage category assigned to the actor

Environmental conditions

Environmental information, such as weather, lighting conditions, traffic, and road conditions, can be included as a set of attributes for each scene in a scenario. These attribute sets are optional, and the scenario setup specifies which of them are used:

- Weather: conditions like brightness, rain, wind, snow, hail, fog, or smoke at each stage of the scenario
- Traffic: traffic conditions, including whether traffic is flowing smoothly or congested, and may also capture phenomena like traffic waves or accordion traffic

- Road state: the state of the road surface, for example dry, wet, flooded, or icy

When these attribute sets are linked to a scenario, their values can be adjusted for each scene stage in the “Environment” section of the storyboard.

Dynamic elements

Describing the relationships between multiple road users is a challenging task. Within the project, the dynamic representation is based on hybrid graph models that integrate spatial, temporal, and interaction dimensions, as described in [25]. To efficiently represent possible traffic patterns, base scenarios are used. These base scenarios describe either the movement of the ego vehicle or, in case of interactions, the relationship between the ego vehicle and one object road user. These base scenarios are derived at an abstract level using an ontology, as seen in Figure 14. Compared to scenario description languages, this systematic approach helps reduce the number of abstract scenarios, enabling efficient implementation and data utilization for scenario generation from data. Furthermore, it gives the opportunity to systematically add new concepts as needed to account for the potential new requirements.

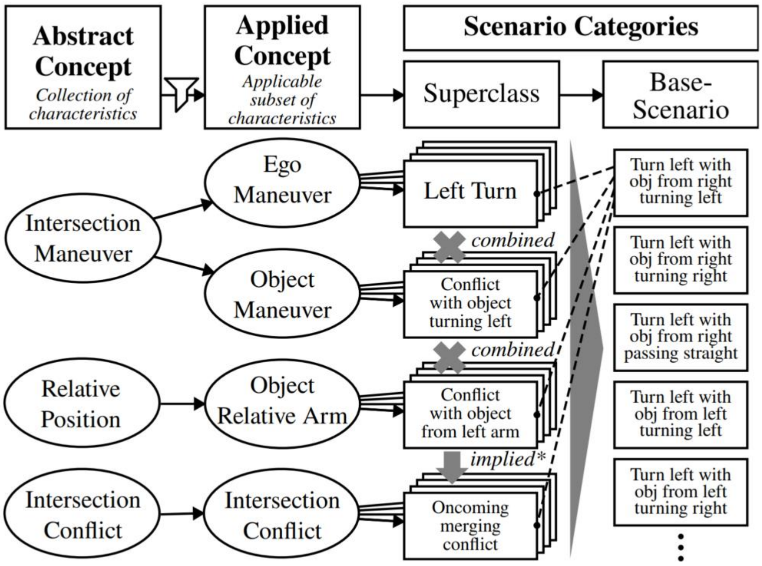


Figure 14: Derivation of base scenarios [41]

With these base scenarios, more complex traffic can be described combining base scenarios (see scenarios

Compared with assembling scenarios from individual elements, as is common in scenario description languages [20], this approach brings the benefit of requiring a smaller number of scenarios. This reduction allows more efficient aggregation of individual scenarios while still providing the flexibility needed to describe traffic situations comprehensively. Figure 15 visualizes how base scenarios are combined.

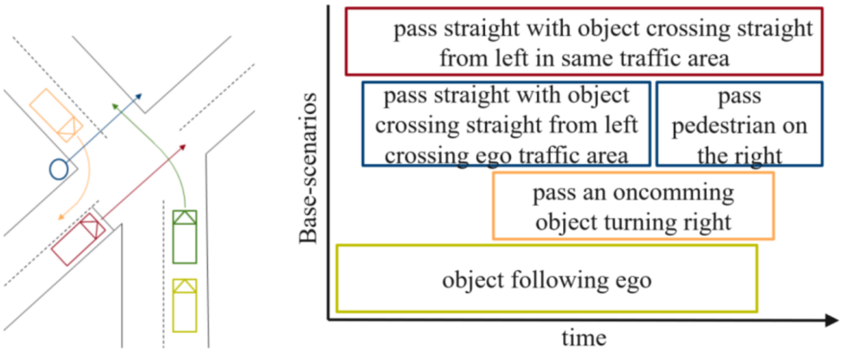


Figure 15: Combination of base scenarios [26]

Based on these base scenarios, parameters are defined. Whenever possible, these parameters are specified for each applied concept and propagated through the ontology along with upcoming emergent parameters, deterministic constraints (incl. both equality and inequality constraints) and causal relations for an efficient and explainable scenario generation [41]. Since different methods may require different information, the parameters defined from the common set of abstract scenarios are linked to different parametrizations. Along with the given relations, a probabilistic ontology is set up as part of the scenario concept like [42]. Compared with maintaining multiple independent parametrizations, this has the benefit of storing parameters more efficiently and ensuring consistency more easily. The hierarchical structure and systematic formulation of the dynamic scenario elements help to setup a creation of an argumentation structure to assess the completeness of the concept [43].

3.6 Scenario identification, extraction, enrichment

The identification, extraction, and enrichment of scenarios is a process through which scenarios are created from pre-existing Scenario Source Data (SSD). In SYNERGIES, SSD is standardised into the OMEGA Prime format. This format encodes detailed descriptions of relevant infrastructure elements (including geometry and state) and the types and kinematics of traffic participants as they evolve over time. SSD can originate from empirical observations collected by instrumented vehicles or roadside equipment, or it can be generated synthetically. OMEGA Prime is source-agnostic, and all data exchanged between WP3 and WP5 within SYNERGIES is assumed to be in this format.

Scenarios derived from SSD may be *concrete* instances that map to predefined abstract and logical scenarios. Abstract scenarios define the conditions that an observation needs to satisfy to be classified as an instance of a scenario (identification), while logical scenarios specify how that instance should be parameterised (extraction). Identifying concrete instances in SSD can be achieved through explicit algorithms tailored to each abstract scenario, formal interpretation of scenario definitions against source data, supervised machine learning, or, in some cases, manual labelling. Most approaches benefit from an intermediate semantic compression step, where SSD's raw representations of the real world (precise geometry and trajectories) are transformed into compact, structured representations that preserve the essential roles, interactions, and context of road users by mapping them to higher-level traffic concepts shared with abstract scenario definitions.

SSD can also reveal new *abstract* and *logical* scenarios when data exposes recurrent or atypical situations that are not yet catalogued. Data mining techniques can detect near-misses or other unusual events that merit formalisation as new scenarios. A complementary method is completeness analysis, where unlabelled portions of SSD are analysed to determine which abstract scenario best represents those conditions and how that scenario should be defined.

A particularly important SSD source is real-world crash data, which captures current critical situations relevant to automated vehicle safety and mitigation. Crash records are not available directly in OMEGA Prime; however, accident reconstruction from police reports and other evidence can produce OMEGA Prime SSD, or crashes can be modelled directly at the abstract and logical levels to create scenario descriptions.

Finally, although the primary goal is to build collections of scenarios that model the conditions present in a dataset or ODD, directly replaying a specific observation in simulation also has high value—especially for incidents and crashes—because it allows one to evaluate whether a system under test could have prevented or mitigated the event.

This chapter describes these data-driven approaches in the following sections:

a) **Scenario identification and extraction from data**, in which we describe multiple methods to identify and extract concrete scenarios from SSD and the role of semantic compression in mapping raw data to higher-level traffic concepts.

b) **Non ordinary situations identification:** data mining techniques for discovering atypical situations, including near misses, that motivate new abstract and logical scenarios.

c) **Accident or incident analysis:** challenges and methods for creating scenarios from accident data.

d) **Direct conversion of an observation to a simulation file:** using raw SSD for replay to evaluate mitigation benefits and perception/fusion performance.

e) **Completeness analysis:** methods for analysing unlabelled SSD to identify and define new abstract scenarios.

Table 4 summarizes requirements for scenario identification, extraction, and enrichment that were derived before developing the methodologies described within this chapter.

Requirements

Table 4: Requirements for Scenario identification, extraction, enrichment methodology

ID	Requirement Topic	Requirement
Rq1	Identification & extraction input format	A consistent interface (file format) for SSD is required to support the manifold sources of data (recordings from test drives, surveillance cameras, drone-based recordings, accident data bases). Every data provided for Identification & Extraction must use that same format.
Rq2	Identification & extraction input format	The common format should contain all the necessary information for scenario detection and enrichment. This includes traffic participants, their properties and their trajectory, as well as infrastructure description.
Rq3	Identification & extraction output format	The format used to describe a concrete scenario shall be simple, standardized and transparent. It should also maintain the link between that specific concrete scenario and the abstract and logical scenarios that define its type and content.

Rq4	Common concept	The scenario description for scenario identification and test scenario generation should share the same scenario concept.
Rq5	Common concept	The scenario concept should also include a practical semantic framework, that allows the definition of abstract scenarios as well as the semantic compression of data into a common logical scenario model.
Rq6	Common concept	Scenarios to be identified and extracted from data must be defined formally in accordance with the common scenario concept.

3.6.1 Scenario identification and extraction from data

Scenario identification and extraction from real-world data involves analysing large volumes of driving data to detect situations in the real world that are relevant for safety and performance assessment. By systematically extracting these scenarios, engineers can build comprehensive test suites that reflect the variability of real-world driving. The workflow will depend on the scenario source data (SSD) used. While the task of scenario identification and extraction is usually associated with data-driven datasets, alternative approaches may be required.

For data-driven datasets, the workflow typically starts with preprocessing sensor and vehicle data, including object detection and environment annotation. Afterward, manoeuvre detection algorithms (e.g., detection of a cut-in) and rule-based filters (e.g., “during night-time, a pedestrian appears at a crosswalk, and the ego vehicle speed exceeds 40 km/h”) are applied to extract and label specific data segments according to the defined specifications. Event detection algorithms can automatically scan and group large datasets, while rule-based filters allow for the manual specification of logical constraints to identify scenarios that match certain engineering criteria. Besides the automated algorithms and filters, manual identification and extraction of scenarios can also be performed as a standalone methodology or used to support the extension of the above-described algorithms and filters.

The process of scenario identification and extraction from data presents a range of technical and methodological challenges that must be addressed to ensure the robustness, scalability, and relevance of the resulting scenarios. One of the primary challenges lies in the heterogeneity of scenario source data (SSD). The SYNERGIES methodology must accommodate a wide variety of data types, including sensor-based recordings, crash reconstructions, knowledge-based inputs, and AI-generated content. Each of these sources differs in structure, resolution, completeness, and semantic richness. As a result, the identification and extraction workflows must be adaptable and modular, capable of handling both structured and unstructured data while maintaining consistency in output formats.

Another significant consideration is scalability, driven by the large volume of data collected from real-world driving, simulations, and crash databases. Automated tools must be capable of processing large datasets efficiently, without compromising the fidelity of extracted scenarios. This requires robust preprocessing pipelines, efficient data filtering mechanisms, and scalable storage and retrieval systems. Moreover, the integration of manual review processes must be balanced against the need for throughput and repeatability.

Criticality detection is also a central concern when a SUT is specified. Identifying which situations are safety-relevant or performance-critical requires the definition and application of appropriate metrics. These may include surrogate safety measures such as time-to-collision, minimum distance, or deceleration rates, as well as more complex indicators derived from behavioural models or risk assessments. The selection of these metrics must be aligned with the intended use cases and the capabilities of the automated driving systems under evaluation.

In addition, the interpretability and traceability of extracted scenarios are crucial for validation and regulatory acceptance. Each scenario must be linked back to its source data, with clear documentation of the extraction logic, assumptions, and parameterization. This traceability ensures that scenarios can be audited, reproduced, and refined as needed.

Together, these challenges underscore the need for a flexible, modular, and standards-aligned methodology. By addressing data heterogeneity, ensuring scalability, and incorporating expert validation, the SYNERGIES approach to scenario identification and extraction lays a solid foundation for the generation of meaningful, diverse, and simulation-ready scenarios. The remainder of this chapter provides a detailed overview of the various approaches and methodologies applied within SYNERGIES for scenario identification and extraction. Specifically, the following sections describe the use of structured scenario concepts and tagging, the identification of scenarios from crash datasets, automated scenario mining and filtering techniques, as well as manual and hybrid approaches.

3.6.2 Semantic compression

Semantic compression transforms spatio-temporal Scenario Source Data (SSD) into a compact, logical-level representation aligned with the project ontology. Its goal is not only to reduce data volume but also to preserve and express the semantic relationships that describe how actors interact with the environment—for example, *"vehicle on a lane"*, *"vehicle in front of…"*, or *"pedestrian crossing the road"*.

This stage bridges raw sensor data and the Logical Representation of Circumstances, preparing inputs for scenario identification and generation. By replacing detailed geometry and low-level signals with ontology-based relations, semantic compression makes data standardized, searchable, and interoperable across tools and partners.

In the SYNERGIES workflow, heterogeneous SSD—trajectories, map geometry, and time-aligned signals—are anchored to OpenDRIVE and OSI references. OpenLABEL annotations bind events, roles, and relations to these structures, ensuring a consistent semantic layer. Typical reductions reach one to two orders of magnitude while retaining all behaviourally relevant content.

Three core principles guide the process:

1. **Semantic fidelity**: preserving relationships and roles that define traffic meaning (e.g., yielding, merging, following).
2. **Geometric and temporal consistency**: simplifying data within bounded spatial and temporal tolerances, maintaining lane topology and key instants such as minimum time-to-collision.
3. **Ontology alignment**: representing all entities and interactions through standardized semantic categories.

Compression follows semantics-aware retention rules, like semantic-aware video compression for automotive perception [44]: essential elements—vehicles, lanes, pedestrians—are retained at high fidelity, while non-influential background data are reduced.

Comparable approaches include ScenarioNet (Li, 2023) and scenario.center (Schuldes, 2024), which abstract recordings into structured entities; Karunakaran et al. (2022), which estract lane-change scenarios in OpenDRIVE/OpenSCENARIO; and Chat2Scenario (2024), that extends this abstraction via language-based and LLM-assisted methods.

Through these steps, semantic compression forms the bridge between raw SSD and logical, ontology-driven data, enabling efficient scenario mining, reuse, and a reduction in validation costs.

3.6.3 Automated scenario mining and filtering

As outlined in the preceding section, there are many methodologies that can be employed to identify and extract scenarios from data. However, when dealing with large-scale datasets or continuous data streams, it becomes essential to design scalable and flexible processing pipelines for scenario extraction. Combined with a knowledge-based approach to scenario definition, it is possible to robustly extract scenarios from complex data sources. The objective of this process is to establish a methodology for scenario identification and extraction from large-scale data. The foundation for this approach lies in the SSD format, the six-layer model, and a set of predefined scenario variants (or rather base scenarios) as defined in 3.5 to be identified and extracted.

The knowledge-based component derives scenario hypotheses from the digital map provided by the SSD format based on the abstract scenario definitions. These hypotheses are generated based on potential interactions between traffic participants, which in turn are inferred from their routes within the digital map provided in the SSD format. Based on this set of scenario hypotheses, specific processing chains are configured for each scenario. These pipelines are designed to extract scenario instances from the data and store them persistently in a suitable format (e.g. Open Source TASI format [45] or ASAM OpenSCENARIO [1]).

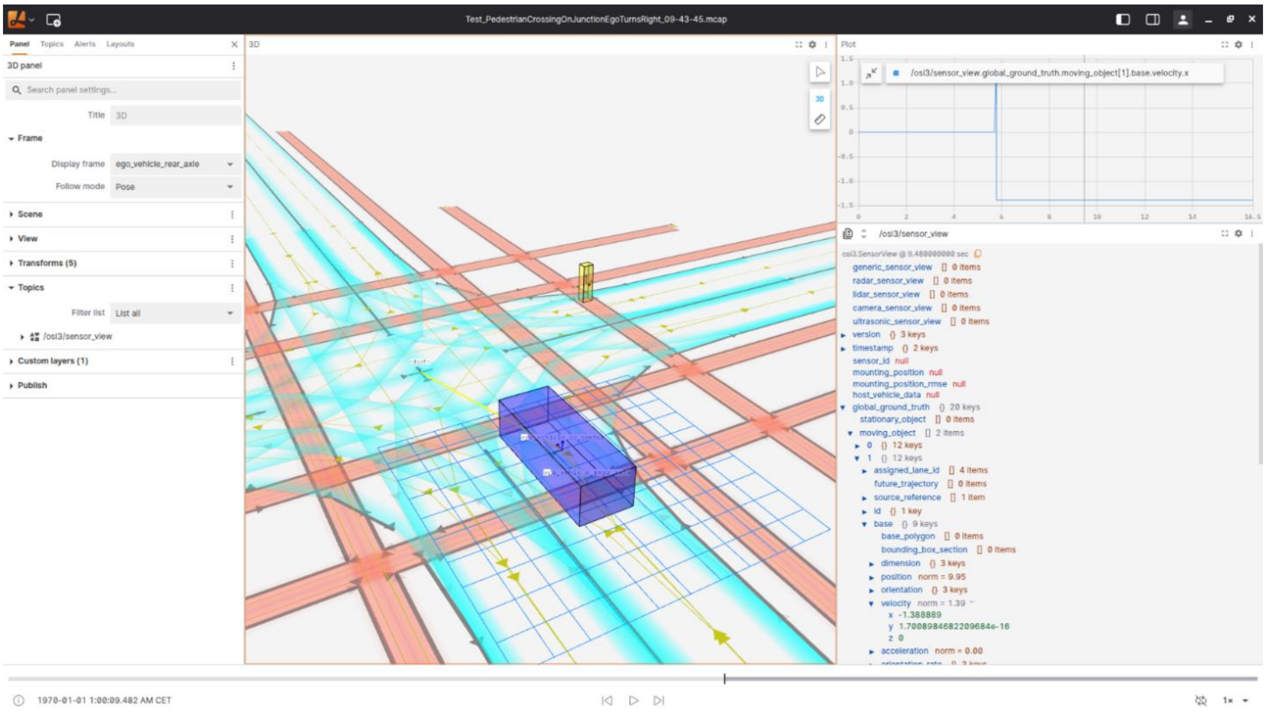
Each chain consists of multiple steps aimed at assigning traffic participants to their corresponding scenarios and enriching the data by retrieving or estimating missing information. Since real-world data often contain noise and incomplete information, preprocessing is a crucial part of this procedure. This includes filtering out noise or assigning participants to lanes in the digital map when such information is not already available. The exact configuration of each processing chain depends on the scenario being extracted, with the six-layer model serving as the conceptual foundation for scenario definition.

The extracted scenario instances can be used to define logical scenarios, for follow-up analysis (e.g. in terms of situation criticality or behaviour of traffic participants) or to generate test scenarios.

3.6.4 Non-ordinary situation identification

The objective of this process is to capture rare, critical, or unexpected situations in addition to detections by standard methods, which are essential for improving scenario coverage and validation robustness. There will be cases where automatic scenario identification and extraction tools cannot recognize logical or abstract scenarios in the provided Scenario Source Data (SSD) or where extended periods occur without any identified logical or abstract scenarios. This can be either due to a failure of the scenario extraction tooling or because no scenario has been defined yet to cover this case. Time periods in which an object trajectory in the SSD is classified as out of distribution by an anomaly detector will also be flagged for further analysis. These time periods can then be analysed further for the identification of non-ordinary situations, improvement of identification and extraction tools, or revision of existing logical scenarios.

One possibility is the manual identification of non-ordinary situations by visualizing the SSD with the tool Lichtblick, as seen in Figure 16. The 3D panel in the centre illustrates what happens in the SSD as if viewed through a camera. This is very useful for visually identifying non-ordinary situations manually. The needed timestamps for the scenario extraction can also be taken from the "current time" display label at the bottom left.



Title 3D

Frame

Display frame ego_vehicle_rear_axle

Follow mode Pose

Topics

Filter list List all

/osi3/sensor_view

```
osi3.SensorView @ 9.480000000 sec
  generic_sensor_view  [] 0 items
  radar_sensor_view  [] 0 items
  lidar_sensor_view  [] 0 items
  camera_sensor_view  [] 0 items
  ultrasonic_sensor_view  [] 0 items
▸ version  {} 3 keys
▸ timestamp  {} 2 keys
  sensor_id null
  mounting_position null
  mounting_position_rmse null
  host_vehicle_data null
▾ global_ground_truth  {} 20 keys
    stationary_object  [] 0 items
  ▾ moving_object  [] 2 items
    ▸ 0  {} 12 keys
    ▾ 1  {} 12 keys
      ▸ assigned_lane_id  [] 4 items
        future_trajectory  [] 0 items
      ▸ source_reference  [] 1 item
      ▸ id  {} 1 key
      ▾ base  {} 9 keys
          base_polygon  [] 0 items
          bounding_box_section  [] 0 items
        ▸ dimension  {} 3 keys
        ▸ position  norm = 9.95
        ▸ orientation  {} 3 keys
        ▾ velocity  norm = 1.39
            x -1.388889
            y 1.7008984682209684e-16
            z 0
        ▸ acceleration  norm = 0.00
```

1970-01-01 1:00:09.482 AM CET

Figure 16: Example of visualizing scenario source data

Another possibility is to use metrics to systematically detect hazardous situations within driving data. The goal is to monitor risk signals in these data streams and then analyse situations to determine the relevance of uncovered sequences and derive new scenarios. This process is particularly useful when scenario source data comes from simulations or real-world drives.

For example, ego-centred metrics, which assess criticality directly for the ego vehicle based on its actions and reactions to a target vehicle, or region-centred metrics, which measure overall criticality by considering all actors present in the ego vehicle's region of interest (ROI), can be combined [2].

Figure 17: Example of a global severity score computed from simulation data

If these values of cross critical thresholds in sequences are not yet covered by existing scenarios, they reveal specific combinations of environmental factors that could lead to a hazardous event. Whether such a sequence is formalized as a new scenario is determined through expert analysis, which considers its novelty with respect to the existing scenarios.

After a scenario has been identified and extracted, the scenario generation process (see Chapter 3.7) can continue in two different ways:

a) The identified scenario can be mapped to an existing logical/abstract scenario. This means the automated scenario identification and extraction should be extended to detect these kinds of situations automatically in the future.
b) The identified scenario cannot be mapped to an existing logical/abstract scenario. This means, that a new logical/abstract scenario must be created for future automatic detection.

3.6.5 Accident or incident analysis and extraction from data

One way to extract scenarios is to use data from real-life crashes. This information is particularly valuable for validating ADAS and AD functionalities, as it reflects critical traffic situations. Modern assistance systems in vehicles are designed to prevent these situations or at least mitigate their consequences.

Crash data is merged into corresponding databases. However, their content varies depending on the motivation behind the data collection. They therefore offer different advantages, but at the same time are subject to restrictions in terms of their usability.

The most obvious source of crash data is police crash statistics. These form the basis for official statistics and provide an almost complete picture of real-life crashes. Representativeness is therefore a key advantage of this data. However, it offers only limited technical details. In general, there is no reconstruction of the incident, and essential information on the dynamics of the entities in the scenario is missing. In addition, the quality of the data depends heavily on the police unit collecting it and its experience in the field [46].

So-called in-depth investigations can be considered as a counterpart to police data (Figure 18: Essential steps of in-depth crash investigation). These provide detailed insight into the course of the crash scenario. Reconstruction is primarily useful for this. It provides so-called "ground truth" data, which is suitable as a basis for further analysis. Nevertheless, in-depth investigations are not limited to the technical aspects of a crash (dynamics, deformations, energies) but also consider the physical consequences of the crash for the people involved. Due to their data quality, such crash statistics are a central source for test scenarios in the development process of new driving assistance systems. Examples include the databases of the German In-Depth Crash Study (GIDAS) and the Crash Investigation Sampling System (CISS). The Initiative for the Global Harmonization of Crash Data (IGLAD) pursues a broader approach. Here, data from several in-depth crash investigations carried out worldwide are brought together to create the largest possible basis for investigations [V4SAFETY – D4.1].

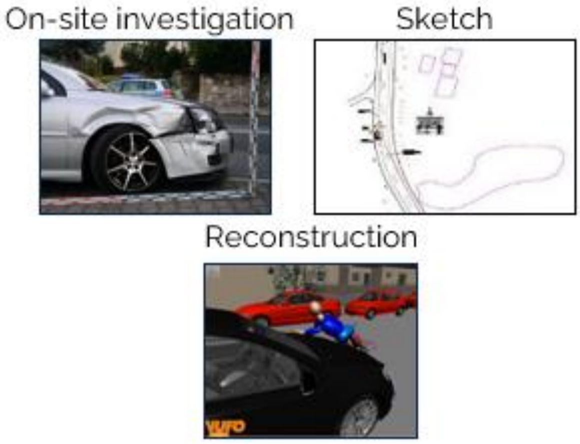


Figure 18: Essential steps of in-depth crash investigation

Data from event data recorders (EDRs) is also becoming increasingly relevant in this context. These are becoming more widespread in the vehicle fleet, not least due to legislative requirements. They provide insight into sensor values recorded by the vehicle immediately before, during, and after a collision. This data can be used in crash reconstructions and reduces the uncertainty of their results. However, the information contained therein is too limited to be used as the sole source for describing a scenario. EDR data can be incorporated into in-depth investigations and thus improve their quality [V4SAFETY – D4.1].

In addition to the data sources stated so far, hospital and insurance data should also be mentioned. These have a stronger medical or macroeconomic focus and are less concerned with technical aspects. They are therefore of little importance to the present considerations. That said, medical data can also serve as one of the possible data sources for in-depth investigations [V4SAFETY – D4.1].

When considering the identification of scenarios from crash data, a special feature becomes apparent. In contrast to, for example, stationary camera-based traffic observations, there is no need to identify suitable events. The very nature of a crash means that each individual case already describes a scenario.

The main task is then to convert the information from selected crash data into scenarios that can be used for the specification and validation/testing of ADAS, including simulation.

The required information can be divided into two groups: dynamic data and static environmental information. The latter is used to describe the environment of a scenario. This includes road layouts, objects, lane markings, and weather conditions. The driving paths of those involved, defined by trajectories, describe the dynamic component of a crash scenario.

The key environmental data is depicted in a crash sketch, which is created by combining photographs of the crash site and its measurements. It also forms the basis for the crash reconstruction. Considering the secured traces and the physical characteristics of those involved the course of the crash is reproduced as accurately as possible using appropriate software (e.g., PC-Crash). The result is trajectories and a definition of the speed and acceleration profiles. This also provides parameters such as the initial speed and collision speed. The results of the reconstructions can in turn be stored in a structured database or converted into an appropriate format, which enables their use in simulations. One example of this is the Pre-Crash Matrix (PCM). As a freely available format, it can be used for various in-depth investigations, and PCM files exist for both the GIDAS database and the international IGLAD data.

The information available after reconstruction is sufficient to reproduce the scenario in a (virtual) test environment. This requires conversion into an appropriate format (OpenSCENARIO). As a first intermediate step, the SYNERGIES project defined a uniform format for "ground truth" data, such as that generated during crash reconstruction or camera-based traffic observations: OMEGA Prime (Common Data Format – CDF). It essentially consists of two components: an ASAM OSI (Open Simulation Interface) and an ASAM OpenDRIVE file. OSI describes the dynamics, and OpenDRIVE describes the static elements. A central task in SYNERGIES will be to develop methodologies that can be used to describe the available "ground truth" data in OMEGA Prime. This step is necessary to make the data homogeneous with other source data in the SYNERGIES project. However, a further step is necessary to use the crash scenarios in a test environment. Many applications for testing ADAS rely on the ASAM OpenSCENARIO format for the dynamic description of a scenario. Therefore, another project goal is to develop a corresponding tool for converting OMEGA Prime data (i.e. OSI) to OpenSCENARIO.

In addition, it is possible to add descriptive tags to the crash scenarios. The information for this is extracted from the crash data itself. This ensures that application-specific scenarios can be searched for easily. The SYNERGIES project will use the ASAM OpenLABEL format for such metadata.

Finally, some examples of previous work that has been done to generate test scenarios for ADS with crash datasets include using Stats19 data [47] and GIDAS data (Babisch et al, 2023). While Stats19 [48] data contains police-reported information on road crashes, GIDAS [49] data offers detailed in-depth data on crashes in Germany where at least one person was injured. In both cases, the information used from the crash datasets has been categorical information which can help to understand the main conditions (temporal, spatial or criticality phenomena), that could be associated with a crash involving personal injury.

Another example related to the use of trajectory information from crash datasets is that previous work has also been done where TASC data has been mapped to AD test scenarios, based on the interaction of the participants involved and their kinematic behaviour (Urban et al, 2020). Similarly, TASC data and GIDAS-PCM data have also been used to identify potential edge cases for Automated Driving (Hi-Drive, 2025). In these situations, the analysis of the overall situation and the interaction of the participants and their kinematic behaviour is what is considered instead of the perception information that is used in the data driven SSD approaches.

3.6.6 Direct conversion of an observation into a simulation file

The concept of direct conversion of an observation into a simulation file describes the seamless transformation of real-world traffic scenes into simulation-ready formats. By bypassing extensive manual reconstruction, this approach ensures that observed road user behaviours can be transferred quickly and authentically into virtual environments for analysis, scenario generation, and system validation.

OpenSCENARIO supports the description of the movement of the involved actors in the scenarios as a trajectory. These trajectories can be derived from observed trajectories in the form of either logged GPS traces or points in a specified local Cartesian coordinate system, which are derived from stationary observations. Additionally, there is the possibility to link these trajectories to an environment, which is represented in an OpenDRIVE file.

3.6.7 Coverage and completeness analysis

When using a set of scenarios to evaluate the safety of an ADS, this collection should encompass the ADS's ODD. Consequently, research has focused on developing coverage metrics to assess how well a scenario set addresses an ODD. For a summary of coverage metrics, refer to [50].

This section specifically addresses the coverage of logical scenario sets. Two perspectives can be considered here. First, on a more abstract level, the logical scenarios should represent the types of situations and conditions an ADS might face within its ODD. For instance, if a junction is included in the ODD, there should be a logical scenario featuring a junction with traffic approaching from all valid directions. Second, on a parameter level, the parameters and their ranges should reflect real-world possibilities.

To evaluate whether a set of logical scenarios adequately covers an ODD, one may use concrete scenarios derived from real-world observations. If a concrete scenario does not match any existing logical scenario, the logical scenario set can be deemed incomplete, as this set does not reflect all concrete scenarios. However, the opposite is not necessarily true: even if every concrete scenario matches with a logical scenario, the logical scenario set might still be incomplete because the concrete scenarios may not represent all possible variations.

Matching a concrete scenario to a logical scenario is not straightforward. Using all details from a concrete scenario might mean no logical scenario corresponds to it. For example, a highway scenario with dense traffic could involve dozens of vehicles visible to the ego vehicle, while no logical scenario might account for that many vehicles. Additionally, logical scenarios often simplify time-dependent state variables; for example, a vehicle's speed might be assumed constant or modelled with a specific function, which may differ from reality. Therefore, two factors must be considered when determining if a concrete scenario corresponds to a logical scenario:

1. The logical scenario must include all relevant elements of the concrete scenario. For example, not every vehicle in a dense traffic scenario needs to be represented, as long as those affecting the ego vehicle's behaviour in a relevant way are included. In case the ego vehicle is equipped with an SUT, e.g., an ADS, whether an element is relevant (i.e., affecting the behaviour of the SUT) depends on the SUT itself.
2. If the logical scenario uses models to represent time-dependent state variables, these models must adequately capture the changes in these variables over time as seen in the concrete scenario. For example, if a traffic participant's speed varies significantly, a

model assuming constant speed may be insufficient. Whether a model sufficiently captures temporal changes also depends on the ADS in the ego vehicle, see [51].

Concrete measures that take these factors into account are developed in SYNERGIES and will be documented in the forthcoming deliverable D6.1.

If the logical scenario set does not fully encompass all concrete scenarios, two options exist: creating a new logical scenario or modifying [52] an existing one. For instance, parameter ranges might be adjusted, or a traffic participant added. When revising a logical scenario, coverage analysis should be repeated because changes might cause previously matching concrete scenarios to no longer correspond to any logical scenario.

3.6.8 Considerations

Within the broader methodology, the identification and extraction of scenarios from data form the cornerstone of the **"Bottom-Up"** approach, introduced in 3.5.2. This step is particularly challenging because it must accommodate a wide variety of data sources, ranging from ordinary driving records to incident and accident reports, and translate them into scenarios that may be concrete, logical, or abstract. While numerous approaches exist to perform this task, they differ in terms of the data they rely on, the level of abstraction they target, and the techniques they employ.

What emerges from this diversity is not a single prescriptive method, but rather the recognition that scenario identification and extraction must be understood as a **family of approaches**. Each has its own trade-offs in terms of effort, scalability, and precision, and each reflects the priorities of the stakeholders/partners who developed them. Against this backdrop, our effort has been to provide a general framework, a kind of **map** in which different tools and practices can be situated, allowing stakeholders to recognize their own contributions within a shared structure. This framework defines processes, such as non-ordinary situations identification, accident or incident analysis and coding, semantic compression, and scenario identification and extraction from data, as well as interfaces, such as the OMEGA-PRIME format or abstract scenario definitions, that provide multiple pathways between processes.

Ideally, existing standards such as **ASAM OpenSCENARIO**, **OpenDRIVE**, and the **ISO 3450x series** would make it possible to define all interfaces within this framework and to clearly delimit the scope of each specific approach and tool. However, this is not the case. While these standards provide important building blocks, their accumulation does not yet amount to a full methodology. At the same time, our own methodology still falls short of fully standardizing and unambiguously defining all interfaces. The result is nonetheless a step forward: a common reference point that helps relate disparate efforts, fosters mutual understanding, and lays the groundwork for more coherent methodologies and more efficient collaborations in the future.

3.7 Scenario generation

Whereas the previous chapter outlines the identification process of scenarios based on a given scenario concept, this is not sufficient in terms of a comprehensive scenario database for two reasons: first, scenarios may be of interest which are not directly observed in data but are important for a database. Second, identification is not enough, since scenarios must be useable for testing in the context of scenario-based testing. To address both, within SYNERGIES scenarios are generated. Thereby, the two objectives of creating additional scenarios and creating test scenarios must be distinguished. Methods can either contribute to one or both objectives. Within SYNERGIES, both are tackled, and it will be outlined for specific tools how they are contributing to these objectives.

(Test) scenario generation refers to the process of creating synthetic (test) scenarios in a defined format based on existing scenarios, data, and/or knowledge. The literature already outlines various methods for generating (test) scenarios. However, specific challenges arise in the context of scenario databases. To address these, the following sections first establish requirements before defining inputs and outputs.

The (test) scenario generation methodology is based on transferring data and knowledge into (test) scenarios. In this process, already identified and enriched scenarios can be used, along with additional data sources, interviews, or expert knowledge. Meeting the diverse needs of different stakeholders is unlikely to be achieved with a single (test) scenario generation method or even within the same abstraction level. Therefore, multiple methods are combined within an overarching methodology to account for different use cases. A more specific overview of different approaches is given below, whereas specific tools will be described in Deliverable 5.2.

Requirements

Table 5 Requirements for scenario generation methodology

ID	Requirement Topic	Requirement
Rq1	Generation input format	A consistent interface (file format) is required to support the manifold sources of data. It enables the development of consistent scenario generation algorithms.
Rq2	Generation output format	The (test) scenario export format shall be standardized and transparent. It shall enable broad use across various companies and institutions. It shall enable the relevant use-cases of the project. It shall be established and tested. It shall have broad support in relevant simulation software. (Test) scenarios made available on the platform should use the ASAM OpenSCENARIO XML format. Other output formats may be provided for different simulation tools.
Rq3	Utilization of data	The generation method should be capable of efficiently making use of input data/scenarios.

Rq4	Common concept	The scenario description for scenario identification and (test) scenario generation should share the same concepts.
Rq5	Scenario perspective	It shall be possible to specify scenarios with different perspectives, e.g. they may or may not include an ego vehicle
Rq6	Coordinate system	The input data format should also specify the coordinate system. For example, ASAM OSI specifies the ground truth to be in a Cartesian coordinate system.
Rq7	Abstraction levels	The generated scenarios may be classified as functional, abstract, logical, and concrete.
Rq8	Human understandable	The generated scenarios should be human-understandable.
Rq9	Test scenarios	The methodology should be capable of creating test scenarios to be utilized for simulations/ testing systems. The purpose of testing and the system under test may be different depending on the needs of the stakeholder. This can have an influence on the information needed for the (test) scenarios.
Rq10	Reproducibility	The methodology should provide the possibility to generate (test) scenarios reproducibly.
Rq11	Redundancy	The methodology shall prevent the generation of redundant (test) scenarios.
Rq12	Traceability	The methodology should provide the ability to trace from which information (test) scenarios have been generated.
Rq13	Scenario relevance	The methodology should be able to generate relevant test scenarios (e.g. challenging situations based on accident data) for a specific system under test (e.g. AD functions such as collective perception testing).
Rq14	Quality	The methodology should outline the limitations that influence scenario quality. The scenarios should be clearly unambiguous.
Rq15	Searchability	Information should be provided to enable the retrieval of a pool of scenarios for a specific purpose (e.g. testing a given AD function or ODD).

Input/Output

In the context of a scenario generation tool with different data sources and requirements, there are many different inputs for creating scenarios. These can initially be divided into three elements: data, knowledge, and requirements based on a specific use case for further use. Data or knowledge may already be enriched.

Not every input is necessary for each individual scenario generation method, as the inputs used depend on the underlying processing chain. In its entirety, however, the higher-level generation method should be able to use the possible inputs to generate scenarios in a relevant and efficient manner. Differentiating between data and knowledge-driven generation methods, specific traffic data is required for data driven approaches, whereas additional knowledge is needed for knowledge driven approaches, but not necessarily for data-driven approaches.

The scenarios generated in this way represent the output of the method. It is initially not specified whether these should be scenarios in general or specifically test scenarios. As a rule, these are driving scenarios, but perception data is also a possible output. Metadata is also provided. The generated scenarios are enriched via metadata, which provides further information. The overview of the described components is given in Figure 19.

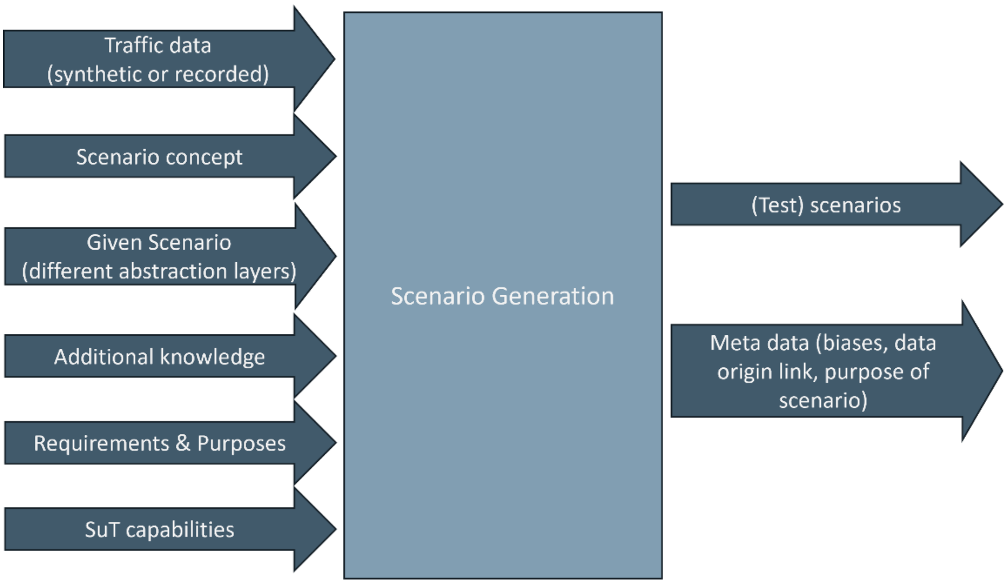


Figure 19: Inputs and outputs for scenario generation

3.7.1 Concrete scenario generation from an abstract scenario

In the SYNERGIES scenario methodology, concrete scenario generation from an abstract scenario provides a direct link between the scenario concept and the processes related to generation of Scenarios of Tomorrow and scenario integration. No logical scenarios are used as intermediate steps. The generated scenarios correspond to abstract base scenarios, or a combination thereof, from the SYNERGIES scenario concept. Concrete scenarios shall be stored in a data format that can be interpreted by simulators (preferably ASAM OpenSCENARIO and OpenDRIVE) or be ready to be converted into one.

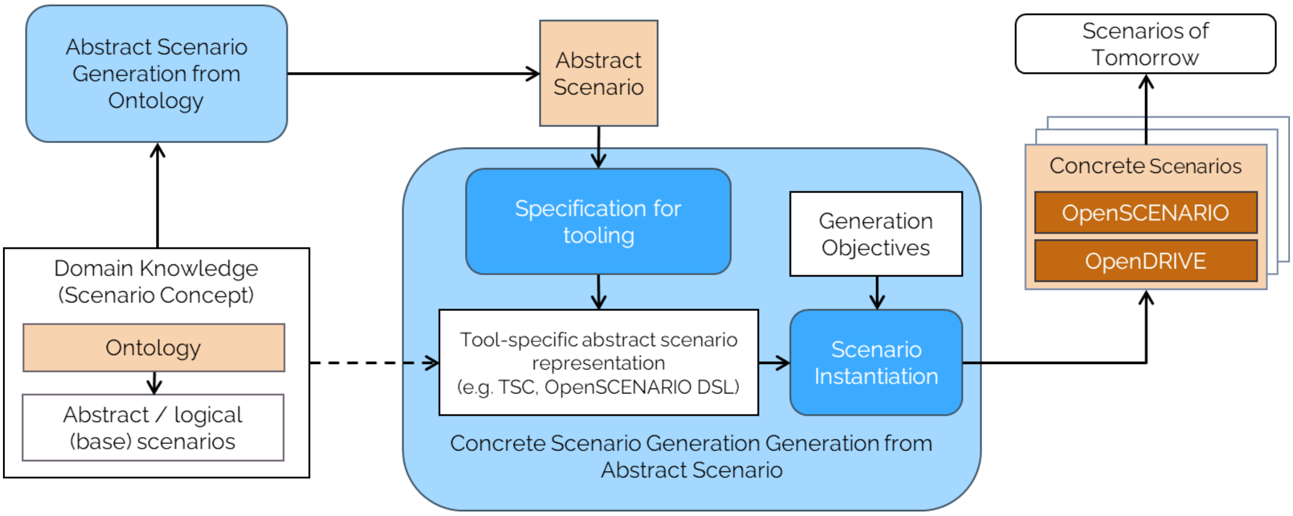


Figure 20: Concrete scenario generation from an abstract scenario method overview

Figure 20 shows an overview of the concrete scenario generation from an abstract scenario workflow and how it is embedded into the SYNERGIES scenario methodology. The method consists of three steps:

1. Specification of the abstract scenario in a formal language supported by the tooling. The abstract scenario may be a combination of base scenarios (see Section 3.2.2) from the scenario concept. As outlined in Section 3.2.2 and the introduction to 3.4, the abstract scenario can originate from different inputs, e.g., expert knowledge and/or recorded traffic data
2. Formulation of generation objectives, e.g., the number of concrete scenarios and criticality bounds
3. Automated generation of instances (concrete scenarios) according to the formalization and the generation objectives

The rest of this section describes the first two steps in more detail. The third step is covered by the TSC2GEMSTAR toolchain, which is described in Deliverable D5.2.

The scenario concept provides an ontology of so-called base scenarios which serve as building blocks for more complex abstract scenarios. For the generation of concrete scenarios, the base scenarios need to be mapped to the native specification format used by the generation tool. Popular textual specification languages for abstract scenarios are OpenSCENARIO DSL and SCENIC [53]. Within SYNERGIES, we propose the use of Traffic Sequence Charts [54], which are the input language for the TSC2GEMSTAR tool developed in the project. TSCs provide rigorous mathematical semantics to a graphical representation of traffic situations. This allows verification that the TSCs for the basic scenarios match their intended meaning. Furthermore, TSCs support temporal operators such as sequence, parallel composition, and choice. This allows users to combine abstract scenarios (from the scenario concept) into a more specific abstract scenario showing a combination of different aspects. Such a combination may be derived from e.g. recorded traffic data.

An infinite number of different concrete scenarios can be generated for an abstract scenario. Therefore, objectives for the scenario generation need to be defined to select a finite set of relevant abstract scenarios. Borchers et al. (2025) propose to evaluate a set of concrete scenarios by their criticality and their diversity. Criticality metrics are commonly used to quantify the inherent risk of a traffic situation or scenario to the involved traffic participants [55]. Research from the VV-Methods project proposes them as a tool for test evaluation [56] and criticality analysis [57]. Criticality metrics are usually evaluated pairwise between actors by extrapolating their current state. For example, two popular criticality metrics are Time-to-collision (TTC) and Required Acceleration (aReq).

Time-to-collision (TTC) is the time span until two actors collide, hypothetically assuming they keep their current direction and speed. The TTC is naturally zero in a collision and infinite if the extrapolated trajectories do not intersect. Required acceleration (aReq) is the lateral acceleration required to avoid an approaching collision between two vehicles. The aReq is naturally infinite in a collision and zero if their paths do not intersect.

Criticality metrics may be aggregated over time and/or actors. Using adequate scenario enumeration techniques (as implemented in the TSC2GEMSTAR toolchain) ensures that the generated scenarios are unique. Besides scenarios being unique, one is usually interested in scenarios that differ by a significant extent. To address this, Borchers et al. (2025) evaluate different metrics for measuring the pairwise distance between concrete scenarios. The proposed enumeration technique (see Borchers et al., 2025) splits the scenario space into distinct subsets along the spatial relations in a TSC. In practice, this already guarantees a good diversity of the scenarios. To estimate the overall diversity, the authors propose to merge the pairwise distances into a single scalar value using a variant of differential entropy. Note that estimating the size of the complete concrete scenario space of a TSC remains an open research question. Therefore, it is not yet possible to estimate coverage of the concrete scenarios with respect to the abstract scenario.

Following the described ideas, we propose to use the following parameters to define generation objectives. These metrics are commonly found in literature and can be applied to many use cases:

- **Target values for one or more criticality metrics**. E.g., select only scenarios with a TTC < 1s and aReq > $5 m/s^2$. The selected metrics and the target values heavily depend on the scenario. The target values may also depend on the capabilities of the vehicles or the system in focus.
- **Number of scenarios**
- **Minimum scenario diversity**. Because scenario diversity tends to oscillate at the beginning of the scenario generation and stagnate with an increasing number of scenarios, Borchers et al. propose to formulate a threshold of the form $\frac{Q}{\ln(1+n\cdot d_{max})} > \alpha$ where Q is the current diversity, n the number of scenarios, and d_{max} the estimated maximum possible distance between two scenarios. They recommend $\alpha \approx 0.5$

The first metric is used for selecting candidates for the generated scenario set and the other two serve as exit conditions. Since it cannot be guaranteed that the required number of scenarios or diversity can be reached, it is reasonable to also define a timeout or similar termination criterion.

3.7.2 Concrete scenario generation from a logical scenario

Concrete scenario generation refers to the instantiation of logical scenarios by assigning exact values to parameters such as vehicle speed, road curvature, or weather conditions. An effective concrete scenario generation method must (1) ensure sufficient coverage of the parameter space, often defined by the Operational Design Domain (ODD), and (2) achieve this within feasible time and computational resources. The challenge, therefore, lies in identifying and prioritizing the most relevant combinations for a given use case. This process can be fully automated, for example [58]. In this approach, the main challenge of capturing the most relevant parameter combinations for a given use case is addressed using combinatorial testing for sampling. To further understand this challenge, the following section explains the importance of parameter distributions, sampling methods, and discretization methods, as these concepts directly influence the relevance, coverage, and efficiency of generated test cases.

Parameter distribution refers to how values of certain parameters, such as vehicle speed, road geometry, or weather conditions, are spread across the ODD. The two targets for these methods are maximizing **coverage** and/or achieving **statistical realism**. The relevance of both targets is well described by the SOTIF [59] standard. A test strategy must ensure sufficient coverage of the ODD. Once the test scenarios are executed, the safety case is based on the information about how often a certain situation happens in the real world and what the possible consequences are when encountering this situation (risk = exposure x severity).

Statistical distribution modelling aims to replicate real-world behaviour by deriving distributions from empirical data, such as traffic recordings or sensor logs. For example, vehicle speeds in cities may follow a normal distribution centred around 40 km/h. Probability density functions formalize these patterns, with common choices including normal, uniform, or beta distributions, or not assuming a predefined distribution at all. Some parameters may be independent, while others often correlate (such as, for example, road curvature and vehicle speed). For that purpose, multivariate distributions must be modelled to capture joint behaviour. Common modelling approaches are, e.g., copulas and kernel density estimations [52]. Surrogate model-based methods rely on the definition of probability distributions of the scenario parameters, which may be cumbersome to extract. Such distributions can be extracted from collected real data. However, often the existing datasets may be limited and may not be able to offer representative parameter distributions. In such cases, hybrid methods have been proposed, which combine both data-driven and knowledge-based information [Glasmacher et al. Generation of Concrete Parameters from Logical Urban Driving Scenarios Based on Hybrid Graphs, VEHITS 2023]. In practice, hybrid approaches often combine the methods to balance completeness, efficiency, and criticality.

Coverage-oriented distribution modelling focuses on maximizing the exploration of the parameter space. The parameter range is, in most approaches, discretised as a first step. Discretization involves dividing continuous parameter domains into finite intervals or categories to enable systematic mapping. One value from each interval is selected and considered as a representative of the interval (equivalence class). The choice of discretization strategy (sampling) affects both the granularity of the search space and the efficiency of scenario generation. Uniform discretization splits domains into equal-sized intervals or, more generally, bins. It is the simplest method, applicable even if no context about the domain is given, but it may overlook critical regions if the parameter distribution is non-uniform. Adaptive discretization adjusts those bins based on distribution density or criticality metrics, such as collision thresholds. Domain-specific discretization categorizes parameters based on semantic relevance. For example, road curvature might be discretized into “straight,” “moderate turn,” and “sharp turn” considering vehicle speed, reflecting its impact on vehicle dynamics.

Discretized domains can be explored using various sampling strategies, each with trade-offs in terms of coverage, efficiency, and relevance to the intended use case. The overview of the sampling strategies is covered in the following section and shown in Table .

Full Factorial Testing (FFT) exhaustively explores all parameter combinations, guaranteeing maximum coverage but quickly becoming infeasible due to combinatorial explosion. Random Testing (RT) reduces complexity and scales easily, but wastes test resources on irrelevant regions, with an overall low probability of hitting the small set of high-risk combinations. Combinatorial Testing (CT) offers a compromise by systematically covering subsets of parameters (e.g., pairwise or t-wise interactions), reducing the search space while still targeting meaningful interactions. Search-Based Testing (SBT), in contrast, uses optimization or evolutionary algorithms to steer scenario generation toward critical regions (e.g., near-collision cases). This makes it highly effective at discovering edge cases, but its success depends heavily on the design of fitness functions, and it does not guarantee full coverage or provide information about what percentage of edge cases was discovered. Finally, surrogate models may be used as a replacement for resource-consuming testing. Such models can be trained to predict scenario outcomes and, once trained, may be employed for exhaustive search of the parameter space in significantly reduced time. This approach can offer higher coverage of the parameter space depending on the model's predictive performance and confidence. On the downside, the effectiveness of such approaches depends on the model's design and may require a significant amount of data for training.

Table 6: An overview of concrete scenario sampling methods

Method	Strengths	Limitations
Full Factorial Testing (FFT)	+ Maximum coverage of the parameter space + Systematic and exhaustive	- Computationally infeasible for large parameter sets due to combinatorial explosion - Requires discretization
Random Testing (RT)	+ Simple, scalable, low setup cost + Good for broad exploration	- Uneven coverage - Low probability of capturing rare critical cases
Combinatorial Testing (CT)	+ Systematic coverage of parameter subsets	- May miss higher-order interactions - Requires discretization
Search-Based Testing (SBT)	+ Targets critical regions effectively + Uncovers rare and safety-relevant edge cases	- Dependent on well-designed fitness functions - Lacks full coverage
Surrogate models	+ Targets critical regions effectively + Systematic coverage of parameter space	- Dependent on model design - Dependent on potentially large amount of training data - Scalability may be an issue

3.7.3 Other scenario generation methods

This chapter provides an overview of some additional methodologies for scenario generation, that are both widely used and actively researched today. These methods aim to complement traditional approaches by enabling richer, more diverse, and logically consistent scenarios.

Large Language Models (LLMs) are proving to be powerful tools for extracting scenarios from unstructured sources such as accident reports. These reports often contain detailed descriptions of real-world events, including environmental conditions, vehicle behaviours, and causal factors. LLMs can identify critical elements and transform them into structured, simulation-ready formats. This capability significantly reduces manual effort and enables scalable scenario extraction from large text corpora.

Beyond extraction, knowledge-based synthesis plays a crucial role in generating scenarios that are not present in historical data. By using traffic rules, ontologies, and domain expertise, this approach allows the creation of logical and hypothetical scenarios, including rare or extreme cases. Such scenarios are essential for testing edge cases and validating system robustness under unusual conditions.

Another promising technique is scenario fusion, which combines scenarios derived from different sources, such as accident data and traffic observations, into unified representations. Bäumler and Prokop [60] introduced a fusion framework that merges heterogeneous scenario datasets to improve coverage and diversity. This method can be integrated with adaptive replay approaches, as discussed by Weber et al. [61], where real-world trajectories are adapted to simulation environments to maintain relevance while ensuring variability.

Finally, scenario-based operational design domain (ODD) modelling, proposed by Glasmacher et al. [62], links scenario databases to ODD definitions. This approach enables systematic coverage analysis and supports the generation of scenarios tailored to specific ODD boundaries, enhancing the completeness of testing campaigns.

These methodologies represent some of the most actively researched and practically relevant techniques today. While not exhaustive, they illustrate the trend toward combining data-driven and knowledge-based strategies to achieve comprehensive scenario sets for automated driving system validation.

3.7.4 Conversion to a simulation-ready file format

The final step of the scenario generation process is the conversion of concrete scenarios into simulation-ready file formats that can be executed and validated within the SYNERGIES platform and external simulation environments. This process ensures that the outputs of WP5 are fully interoperable with the broader SYNERGIES toolchain and compliant with international standards.

Standardized Output Formats

All scenarios generated within SYNERGIES are exported using the ASAM OpenX family of standards, ensuring consistency and reusability across the different toolchains:

- ASAM OpenDRIVE defines the static road infrastructure, lane geometry, and traffic rules
- ASAM OpenSCENARIO 1.x encodes the dynamic elements, including actors, trajectories, and manoeuvres
- ASAM OpenLABEL captures contextual metadata such as weather, traffic, ODD parameters, and provenance information
- ASAM OpenODD (where applicable) provides machine-readable descriptions of the Operational Design Domain

The integration of these standards allows generated scenarios to be natively executed in a wide range of simulation tools (e.g. CARLA, VTD, CarMaker, PreScan, or SCANeR) and shared seamlessly via the SYNERGIES Scenario Dataspace and Marketplace developed in WP7.

Data Model and Interoperability

The OMEGA-PRIME Common Data Framework (CDF) acts as the internal exchange layer between scenario generation modules and the simulation export tools. During conversion, concrete scenario data, originally stored in OMEGA-PRIME (.mcap) format, is mapped to the OpenX schema through standardized adapters developed jointly by ICCS and IRT SystemX ([63], [64]). This mapping ensures:

- Spatial consistency between OpenDRIVE maps and dynamic entities
- Temporal synchronization of trajectories and events defined in OpenSCENARIO
- Semantic integrity with the ontology defined in Section 3.5 and with logical scenario parameters

Validation and Quality Assurance

Before scenarios are released as simulation-ready packages, an automated validation process verifies:

- Geometric and coordinate coherence, ensuring that OpenDRIVE maps and trajectories share a common reference frame
- Kinematic plausibility, confirming that motion profiles respect physical and behavioural limits

- Semantic compliance, ensuring that actor roles, manoeuvres, and environmental conditions match the ontology and the source logical scenario
- Traceability, with all scenario files linked back to their abstract and logical origins through unique identifiers and OpenLABEL metadata

Integration with the SYNERGIES Platform

Once validated, each scenario is packaged as a simulation-ready bundle, consisting of OpenSCENARIO, OpenDRIVE, OpenLABEL, and metadata files. These bundles are:

- Indexed and made accessible through the SYNERGIES Scenario Dataspace using a federated approach, enabling discovery and retrieval by other work packages
- Executable through the WP7 Orchestration and Marketplace tools, which enable automated simulation runs and feedback collection
- Version-controlled to maintain reproducibility and enable future re-execution under updated simulation frameworks

This conversion pipeline closes the loop between abstract scenario definition and executable testing, guaranteeing that all scenarios produced in WP5 are traceable, physically valid, and interoperable within both SYNERGIES and external CCAM simulation ecosystems.

Manoeuvre-based vs. Trajectory-based Scenario Representations

Within SYNERGIES, simulation-ready scenarios may be exported in two complementary forms: manoeuvre-based and trajectory-based, each serving different purposes in verification and validation workflows.

Manoeuvre-based scenario files describe traffic situations using parameterized behavioural primitives such as lane changes, cut-ins, braking manoeuvres, or pedestrian crossings. These scenarios are typically encoded in OpenSCENARIO 1.x/2.0, where actors are defined through a sequence of actions, triggers, and conditions rather than explicit point-by-point paths. This representation is compact, human-interpretable, and well-suited for closed-loop simulation, allowing the System Under Test (SUT) or the simulation engine to compute exact trajectories dynamically based on vehicle models, controllers, and interaction logic. Manoeuvre-based scenarios thus support generative variability, coverage analysis, and sensitivity studies, making them particularly valuable within SYNERGIES for logical-to-concrete instantiation.

Trajectory-based scenario files, in contrast, define fully specified time-discrete paths for all actors. These are typically derived from OMEGA-PRIME recordings (OpenDRIVE + OSI trajectories) or extracted SSD. During conversion, the sampled positions, velocities, and headings are exported either as OpenSCENARIO trajectory actions or directly streamed via OSI interfaces. This representation enables open-loop or semi-open-loop replay, preserving the exact kinematics of observed situations, which is essential for reproducing real-world episodes, validating perception stacks, or assessing crash/near-miss replay fidelity. Trajectory-based scenarios therefore emphasize fidelity and repeatability over behavioural flexibility.

In the SYNERGIES toolchain, both representations are supported: manoeuvre-based scenarios provide abstraction and variability for safety-relevant exploration, while trajectory-based scenarios ensure traceable reproduction of recorded or extracted events. The conversion pipeline ensures that either form can be packaged consistently into OpenX-compliant scenario bundles executable across simulators.

3.7.5 Considerations

As outlined in this section, multiple ways exist to generate (test) scenarios and to use them in a context of a database. several considerations are therefore highlighted.

First, there is no such thing as the perfect scenario generation method, and multiple approaches must be combined to generate (test) scenarios that serve different stakeholders and needs.

Second, regardless of the chosen approach, generating a (test) scenario requires a method that comes with certain limitations and assumptions. These may arise from the design of the method but may also be introduced by the underlying data or user input.

Furthermore, scenarios within a database should be agnostic to a potential SUT. However, since there is a strong link between scenarios and SUTs, these influences should be reflected when converting scenarios into test scenarios.

Considering the limitations and assumptions of scenario generation methods and the resulting scenarios, as well as the potential differences in how test scenarios influence a SUT [65], it is important that these aspects are documented comprehensively.

Finally, scalability of methods is essential for scenario databases to serve not just a single specific scenario but to allow for a certain level of coverage. Therefore, these methods should run as automatically as possible.

3.8 Scenarios of Tomorrow

The interactions between human drivers, pedestrians, and automated vehicles in future mixed-traffic environments represent a decisive challenge for the safe and efficient integration of automated mobility. This chapter focuses on a structured methodology for the analysis of such interactions in a realistic yet risk-free manner. The focus lies on the identification and definition of scenarios that reflect the complexity of tomorrow's traffic, followed by a systematic analysis of behavioural patterns, safety aspects, and risk factors.

Human-in-the-loop simulators are used to incorporate real human behaviour into these investigations. They enable the collection of reliable data on decision-making processes, reaction times, and behavioural adaptations. These insights allow human-behaviour models to be refined and subsequently transferred into practical test standards for next-generation mobility systems. This approach not only delivers a methodological framework for scenario-based analysis but also ensures that results can be communicated, integrated into existing toolchains, and adapted for future use. The concept of a Scenario of Tomorrow (SoT) can be understood as follows:

"A Scenario of Tomorrow (SoT) is an extended scenario concept based on the scenario definition established in SYNERGIES. It optionally extends the underlying ontology to accommodate the representation of possible yet unknown circumstances and developments. A SoT may be instantiated as a Test, Functional, Abstract, Logical, or Concrete Scenario. It envisions the operation of connected, cooperative, and automated mobility (CCAM) systems in future contexts, enabling the integration of assumptions regarding technological progress, policy evolution, user behaviour, and infrastructure development. Through this, a SoT provides a structured framework to describe and analyse potential mobility environments, challenges, and interactions beyond current capabilities in both theory and practice."

New scenarios may emerge not only from empirical data but also from research questions and the development of new standards, which introduce requirements beyond current assumptions. For example, unresolved questions relating to ADS responses to unexpected pedestrian behaviour, the dynamic allocation of lanes via digital infrastructure, or the integration of new mobility forms highlight the imperative to move beyond retrospective or legacy test cases.

This chapter therefore establishes the foundation for defining and analysing emerging scenarios, addressing requirements such as standardisation, interoperability and communication of results, while at the same time recognising challenges related to scalability, compatibility and validation.

3.8.1 Identification of Scenarios of Tomorrow

The identification of Scenarios of Tomorrow (SoT) is a systematic process for selecting and categorizing driving scenarios that do not yet exist in current testing frameworks but are expected to become critical as technologies, behaviours, and infrastructure evolve. In contrast to traditional scenario identification, which relies primarily on existing data from recordings or databases of past events, this approach proactively anticipates future conditions and interactions that must be understood and tested before they occur at scale.

To ensure broad coverage of potential developments, the identification process draws input from diverse sources, organized into three categories:

1. **Empirical data sources**: Empirical sources include existing scenario databases that serve as baselines. Additionally, they include empirical data from new Automated Vehicle (AV) pilot deployments and observational studies, which reveal emerging patterns, such as changes in human spacing around AVs or increased pedestrian distraction due to mobile phone use.
2. **Prospective and expert inputs**: Prospective inputs involve expert knowledge from traffic psychology, engineering, and urban planning, together with scientific literature on trends such as micro-mobility growth, cybersecurity risks, and climate-related changes to infrastructure. Exploratory simulations also contribute by revealing interaction patterns that have not yet occurred in real-world data but are likely to arise under different levels of automation or connectivity. Such simulations vary parameters like AV penetration rates or traffic composition.
3. **Normative and regulatory inputs**: Normative inputs come from new regulations and roadmaps, such as the EU driver-monitoring mandates for driver monitoring (Regulation (EU) 2019/2144) that introduce new functional requirements, or future Euro NCAP [66] assessment tracks, which indicate operating conditions and behaviours that need to be reflected in future test scenarios.

This diverse input foundation ensures that scenario identification captures both evidence-based emerging phenomena and anticipated future developments. The identification of Scenarios of Tomorrow follows a structured process that builds on traditional scenario-identification methods but adapts them to accommodate emerging conditions. The process comprises two main stages, (1) coverage gap identification and (2) prioritization and selection.

1. Coverage Gap Identification

The first stage consists of identifying gaps in current scenario frameworks. This involves comparing existing catalogues with the collected input to determine which behaviours, actors, or conditions are missing. Empirical recordings from AV pilots are reviewed to detect previously unobserved interactions, for example, identifying cases of altered human spacing around AVs. These detected instances become candidate scenarios.

When direct data is not available, documented evidence, statistical findings, and expert insights are synthesized to construct candidate scenarios that address identified gaps. For example, using the distraction prevalence data to define distracted pedestrian crossing during AV approach.

Simulation results add further insight by uncovering future-critical patterns. The simulation parameters, such as AV penetration, connectivity levels, and traffic composition are systematically varied. The findings that are not represented in current catalogues become candidate scenarios.

Regulatory requirements (mandates and standards) are translated into scenarios as well, for example by defining conditions related to various driver states to comply with driver-monitoring mandates.

Finally, existing databases are reviewed to identify structural limitations and to determine which extensions to actor types, behavioural states, or interaction patterns must be added.
The result of this stage is a set of clearly justified candidate scenarios; each linked to the specific gap and input source that motivated its creation.

2. Prioritization and selection

Once candidate scenarios are identified, they are evaluated to determine which require immediate development. Temporal relevance plays an important role, and scenarios are categorized by when they become testable and necessary:

- Near-term (2025-2027): Scenarios based on currently observable or emerging factors. These involve behaviours where data is already available or regulatory mandates create immediate testing needs (e.g., driver monitoring data, pedestrian distraction, early AV interactions).
- Mid-term (2028-2030): Scenarios based on anticipated developments with moderate certainty. These involve gradual technological and infrastructure changes that are projected but not yet widespread (e.g., moderate AV penetration, expanding V2X infrastructure, climate-related infrastructure changes).
- Long-term (2031+): Scenarios based on transformative future conditions. These involve fundamental shifts in traffic composition and adapted human behaviours that require long-term projection (e.g., widespread Level 4/5 deployment, fully adapted human-AV interactions).

Safety criticality is another factor. Scenarios with a high potential for harm or those that expose vulnerabilities in ADS receive higher priority and are assessed using established risk assessment frameworks that consider both severity of potential harm and likelihood of occurrence. Scenarios may be ranked using approaches such as ISO 26262 [67] matrices or criteria adapted from established AV testing methodologies, ensuring transparent and reproducible prioritization that addresses safety-critical situations before lower-risk cases.

The significance of each identified gap also influences prioritization, based on both the nature of the coverage gap and the expected real-world exposure. Scenarios introducing fundamentally new elements into the testing ontologies (e.g., new actor types, behavioural states, or interaction patterns) are flagged for review, but their priority is weighted against anticipated real-world exposure, accounting for factors such as projected penetration rates and deployment timelines. Exposure may be estimated using established classification frameworks such as ISO26262 which categorizes operational situation frequency into defined probability classes. This ensured that prioritization reflects actual testing needs rather than novelty alone. In some cases, extending an existing high-exposure scenario type is more urgent than developing a novel but low-exposure scenario.

Finally, data availability plays a practical role. Scenarios for which validation data or behavioural models already exist (or are soon-to-be-available) are prioritized, as they can be developed and validated more reliably.

The final output of the identification process is a prioritized catalogue of candidate Scenarios of Tomorrow. Each entry describes the specific traffic situation, its underlying motivation, the temporal relevance, the addressed gap, and its priority level. This catalogue forms the basis for developing concrete, executable scenarios for future testing.

3.8.2 Derivation of Scenarios of Tomorrow

The scientific validity of this exploratory approach lies in its ability to anticipate future, potentially unforeseen traffic situations. Achieving this requires a structured and iterative workflow for scenario generation. The process begins with the formulation of hypotheses based on a diverse set of inputs, including empirical studies, literature reviews, accident and incident reports, normative requirements, expert assessments, emerging mobility trends, and existing scenario databases. Researchers then determine which elements of current scenarios should be modified or extended, such as actor behaviour, road infrastructure, or interaction rules. These adaptations are subsequently formalised into new scenario definitions that can be implemented and tested. Insights gained from this process feed back into hypothesis formation, creating a continuous refinement loop.

In contrast to conventional validation methodologies, which typically replicate historical incidents or standardised cases, research-driven approach encourages a proactive stance. Rather than only reproducing known conditions, it systematically extends existing scenario structures to explore future mobility contexts and newly emerging risks. To translate these conceptual impulses into executable test cases, scenarios must first be defined clearly and consistently. In this phase, a scenario is understood as an abstract, structured description of a traffic situation, characterised by its actors, road infrastructure, environmental conditions, and interaction dynamics, that can be instantiated in simulation. Scenarios of Tomorrow are therefore not isolated recorded events but systematically derived configurations. They may arise from empirical observations, from established scenario libraries, or from newly developed hypotheses on future mobility behaviours and system interactions.

To ensure reproducibility and transferability, scenarios are formalised in standardised formats such as OpenSCENARIO, OpenDRIVE and OpenODD, as described in 3.7.4. Once encoded in these formats, the scenarios can be consistently studied across simulation environments and toolchains.

Human-in-the-Loop (HiL) simulations complements this workflow by providing insights into how human participants interact with these newly formulated scenarios. Observing driver and pedestrian behaviour under future-oriented conditions enables researchers to assess scenario plausibility, identify unrealistic assumptions and refine behavioural models accordingly. Through this combination of methodological exploration, normative alignment, and HiL-based validation, a structured, forward-oriented process is created. It strengthens the robustness and adaptability of ADS systems against technological, social and infrastructural change that extend beyond current real-world experience.

3.8.3 Integration and execution of Scenarios of Tomorrow (Human-in-the-Loop)

Driving simulators have become indispensable tools in the development and validation of automated driving systems, particularly when human interaction is a critical factor. They offer a controlled, flexible and safe environment for studying complex traffic scenarios that would be difficult to replicate on public roads or test tracks.

The advantages of driving simulators are key for their use. They offer a safe environment as potentially dangerous or rare scenarios can be reproduced without exposing participants or equipment to real world risk. This also leads to the advantage of reproducibility. Experimental conditions can be precisely controlled and repeated across multiple trials and subjects, ensuring consistency in the data collection process.

A significant application area for driving simulators is within human-in-the-loop studies. In these studies, human drivers interact directly with the simulated environment, which can also include the interaction with potential automated vehicles (AVs). Despite the benefits, simulators also present certain limitations that must be considered in the scientific evaluation. A common problem refers to simulator sickness. Some participants may experience motion sickness like symptoms due to discrepancies between visual cues and physical sensations. A different problem refers to the transferability of the results. Behavioural responses observed in simulator studies may not always be generalized fully to real-world driving due to differences in the precepted feedback, risk perception or immersion.

Recognizing these strengths and limitations is essential when designing human in the loop studies. By leveraging the capabilities of driving simulators while acknowledging their constraints, researchers can systematically investigate the performance and behaviour of humans within specific scenarios.

A well-designed **driving simulator study** follows a structured sequence of phases to ensure methodological completeness, participant comfort and reliable data collection.

Typical phases and their purpose are described in the following section. As the research question must be considered when designing a study, their concrete design might differ. Figure 21 visualizes a general process for driving simulator studies.

Figure 21: General study design based on [68]

In general, a study starts and ends with a questionnaire to collect subjective feedback and information. The goal is to collect data in a standardized format assessing the subjective experience such as workload, trust in automated vehicles or perceived safety. Post-experiment debriefings can also provide qualitative insights into participants perception and identify potential confounding factors.

In a first step the study begins with the introduction phase. Within the introduction phase the participants are briefed on the experimental procedure, vehicle controls and other study relevant information. This phase often includes an acclimatization period allowing participants to familiarize themselves with the simulator environment and vehicle dynamics, thereby reducing initial learning effects or anxiety that could bias subsequent results.

To minimize anticipation or priming effects regarding upcoming test scenarios, studies frequently incorporate distraction drives or filler task. During this distraction phase, participants engage in routing driving or non-critical scenarios designed to mask the timing or nature of critical events under investigation.

The core component of the study consists of one or more experimental trials in which predefined test scenarios are executed. These trials are carefully constructed to probe specific research questions under controlled conditions. Scenarios may range from everyday traffic situations to rare but safety critical events.

To mitigate other effects (e.g., learning, fatigue) that might influence participant responses across multiple scenarios or conditions, studies employ randomization or counterbalancing strategies. By systematically varying the sequence of scenario presentation among subjects, researchers enhance internal validity and ensure that observed effects can be attributed to experimental manipulations rather than procedural artifacts.

To allow for a flexible design within the study, a modular **driving simulator system architecture** is needed. The Toolchain of a simulator includes several components to faithfully replicate key aspects of real-world driving. Although specific implementation may differ, certain components and their purpose are common. An example for a toolchain overview is given in Figure 22.

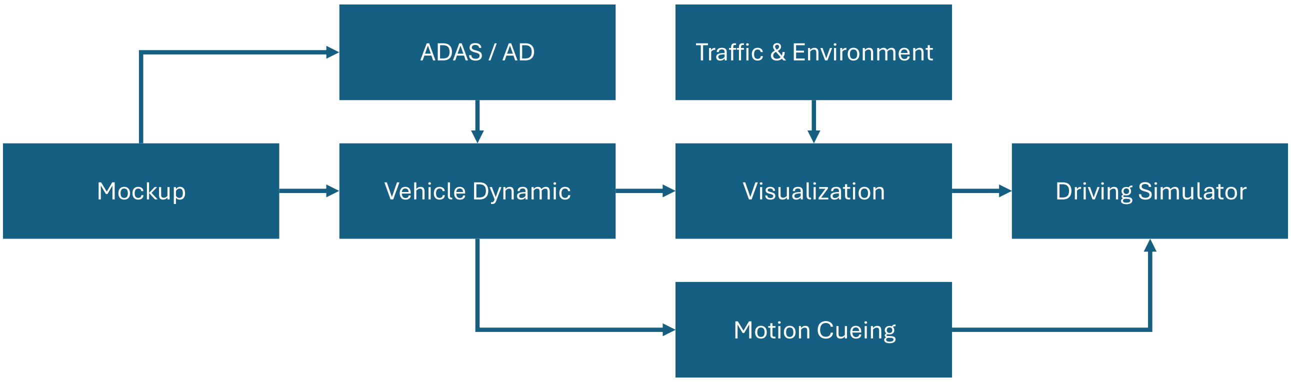


Figure 22: Simulator toolchain

The Mockup includes the physical cockpit or driving controls that the subject uses throughout the study. It provides tactile feedback and facilitates naturalistic human interactions with the simulated vehicle. A Mockup can range from a simple gaming steering wheel mounted to a desk up to a full-size vehicle with its original components.

Dependent on the Mockup is also the usage of a motion platform to simulate forces that act on the subject. Here is also a range from simulators using no motion platform up to simulators working on an 8 DOF Platform including Rails over 10m long to simulate acceleration over 10 m/s^2. The concrete selection of Mockup and motion platform is often based on the concrete research question and the needed realism.

Figure 23: ika's highly dynamic driving simulator [69]

One example for a full-size driving simulator is the ika highly dynamic driving simulator in Figure 23. It combines a Rail with a Hexapod to allow for a 7 DOF movement. With the 12m rail, it can have unscaled lane changes on highways. The use case of this simulator is focused on highway and rural road driving. [69]

The visualisation of the simulated environment can be realised with different option. Starting from a single monitor, usage of projectors realising a 360° surround view or the usage of VR headsets. Accurate visualization is critical for participants immersion and for realising realistic behaviour. Also, the used engine to visualize the souring must be considered.

The vehicle dynamic model simulates the behaviour of the simulated vehicle. It computes parameters such as acceleration, braking performance steering response, tire-road interaction, and more. Fidelity in modelling is essential to ensure that participants experience realistic handling characteristics. The range of possible used models are from simple single-track-models up to fully modelled dynamic models.

The scenario engine manages other traffic participants (e.g., vehicles, pedestrians) and orchestrates their behaviours according to predefined scenarios. It enables complex interactions between the ego vehicle and its environment either through scripted trajectories or AI-driven agents and triggers events at precisely defined moments during experiments. The concrete implementation is often closely tight to the used visualisation engine.

To allow the usage of the data, the data logging is essential for the system architecture. The data logging needs to record all relevant signals during the simulation, including driver input, vehicle states, scenario events and other data if available. Detailed logs are crucial for post experiment analysis and validation of research hypotheses. As the study takes place in a simulated environment, often more data is recordable compared to the real world. However, there is a need to identify the needed data in the study planning phase, to allow for all data to be captured in the defined format.

The **integration of scenarios into driving simulator studies** is a critical aspect that determines both the realism and the scientific validity of experimental results. A central challenge lies in embedding test scenarios seamlessly within a broader traffic context while maintaining control over experimental variables.

There are several approaches to managing the surrounding traffic context during scenario execution. In a minimalist approach, all background traffic is removed except for those vehicles directly relevant to the scenario under investigation. This allows precise manipulation and observation of specific interactions but may reduce validity due to the lack of contextual cues. Alternatively, some studies employ an isolated approach by leaving out all surrounding traffic entirely and focusing solely on one-to-one interactions between the ego vehicle and a single other road user or event trigger. While this maximizes experimental control, it further distances the simulation from real-world complexity. The embedded approach offers another pathway, where full background traffic is maintained throughout the simulation, but only certain actors are programmed to participate actively in the test scenario. This method preserves environmental richness while still enabling targeted evaluation of system responses to specific events.

Translating abstract, logical or concrete scenarios into executable formats for simulators presents additional methodological challenges. Scenarios are often defined using standardized languages such as OpenSCENARIO or described at various levels of abstraction, ranging from high-level event sequences to fully specified trajectories and behaviours. Converting these descriptions into simulator-specific implementations typically requires manual adaptation due to differences in engine capabilities, data structures, and supported features across platforms.

Human behaviour introduces further uncertainty into scenario execution. Unlike purely scripted simulations, human-in-the-loop studies must account for variability in driver actions, decision-making processes, and reaction times. As a result, robust implementation strategies are needed that can handle unexpected participant responses while preserving repeatability for scientific evaluation. One key consideration is balancing replayability, where scenarios unfold exactly as scripted regardless of participants input with adaptivity, which allows for dynamic adjustment based on real-time human behaviour.

Commonly two primary methodologies exist for scenario execution: manoeuvre-based approaches and trajectory-based approaches (see Conversion to a simulation-ready file format). Manoeuvre-based methods specify intended behaviours or manoeuvres (such as lane changes or braking) for each actor but leave their exact execution flexible based on ongoing interactions with the ego vehicle or other agents. Trajectory-based methods define fully scripted paths with predetermined timing and positions for all entities involved in the scenario. This ensures that actors behave the same each time but may limit naturalistic variation and the scenario seen by the ego vehicle. In the concrete scenario implementation, a combination of both is possible to allow for certain action to be manoeuvre based and others to be trajectory based. The concrete implementation depends on the study design, its goal and the engine capabilities.

3.8.4 Analysis of Scenarios of Tomorrow

The analysis of Scenarios of Tomorrow aims to derive empirically grounded insights into how emerging traffic situations may unfold and to translate these insights into actionable parameters for, for example, behavioural modelling, homologation, NCAP testing, and scenario generation. Within this chapter, a deeper look is taken into the data preprocessing and feature extraction steps. In a next step, the methodology for the behavioural modelling and scenario identification and analysis is presented. Based on the results achieved, predictions for Scenarios of Tomorrow are made and recommendations are derived.

All data from the Human-in-the-Loop experiments (e.g. from sensors, simulator logs, questionnaires etc.) are first synchronized and cleaned. Relevant variables, such as position, velocity, accelerations, time-to-collision and communications cues, are extracted for each actor in the scenario. Questionnaire data on subjective factors such as trust, perceived safety, workload are processed and aligned with the corresponding behavioural events. The data is then segmented into key events. This allows the behavioural metrics to be identified within each event and compared across participants and conditions and identify behavioural differences between today's familiar situations and emerging scenarios in future mixed traffic.

The recorded data can be used for various use cases, mostly dependent on the initial research question. In this section, a deeper look is taken at the **behaviour modelling and the analysis of logical scenario parameter distribution**.

A logical scenario describes a scenario with their parameters range. Within the recording of a subject, one or multiple concrete scenarios can be detected. To detect a concrete scenario and align it within the scenario concept is described within chapter Scenario identification and extraction from data. In general, those methods are also capable for detecting scenarios recorded within driving simulator studies. If scenarios shall be detected, which need an extension of the ontology, those aspects need to be implemented within the scenario identification methodology. However, the general methodology on how scenarios are detected is not influenced on the extension.

Based on the results from the scenario identification, further steps can be taken now. Based on the identified logical scenarios, parameter distribution can be compared with data from different sources. A possible use case can be the comparison of comparing the cut-in behaviour based on the rate of automated Vehicles surrounding the subject. The distribution of different rates can be compared. Depending on the results, different interpretations can be done which can be further used to do predictions for future scenarios.

Not only can the scenario be modelled, also the behavioural parameters can be identified to allow the parameterisation of driving models. Depending on the research question, different analyses can be done.

Descriptive analyses quantify the observed distributions of behavioural parameters (e.g., gap acceptance thresholds, braking onset times, lateral position). Measures of central tendency and dispersion are reported, skewness and tails are inspected, and distributions are visualized using histograms, kernel densities, and empirical cumulative functions to characterize typical and extreme behaviours.

Inferential analyses employ mixed-effects (hierarchical) regression to estimate how contextual variables shape behaviour while accounting for repeated measures and nested structure (trials within participants, participants within sites). Models include random intercepts and, where appropriate, random slopes, with fixed effects for automation level, communication modality (human vs. eHMI), and infrastructure configuration, including their interactions. Effect sizes are summarized with confidence or credible intervals, and model diagnostics assess fit and assumptions.

Predictive modelling integrates probabilistic, machine-learning, and agent-based approaches to extrapolate observed trends to untested or hypothetical scenarios. Models are calibrated to empirical data and validated via out-of-sample prediction or cross-validation, then used to generate predictive distributions under novel conditions. Agent-based simulations embed learned decision rules to explore system-level dynamics, with uncertainty quantification and sensitivity analysis supporting scenario planning.

Derived behavioural parameters and patterns are used to define representative and critical conditions for new scenarios. These synthesized scenarios reflect not only typical but also edge-case interactions that may challenge automated systems.

Further, insights from the SoT analysis and **scenario prediction** are combined to create recommendations for the industry, regulators and research community. These may include adjustments to safety validation frameworks, the refinement of testing procedures, or the formulation of new guidelines for the operation of CCAM systems.

3.8.5 Considerations

When it comes to generating and using Scenarios of Tomorrow, it is important to understand that this is a forward-looking task. It is not simply about reproducing past accidents or established test cases. Research questions and regulatory developments challenge existing assumptions and demand the creation of new scenarios that capture emerging risks, novel traffic interactions, and future infrastructures. Treating these impulses as starting points for scenario design is an effective method of avoiding a purely retrospective focus. This approach fosters a proactive methodology capable of anticipating conditions not yet visible in current datasets. The formalisation of such scenarios in standardised formats is essential, as it guarantees comparability, transferability and transparency. This, in turn, enables researchers, industry and regulators to build upon a shared scenario basis.

Another key element is the role of Human-in-the-Loop (HiL) simulations. By presenting human participants with newly designed scenarios, we can gather empirical evidence on how people perceive, interpret, and act in unfamiliar traffic situations. This evidence is vital for validating the plausibility and relevance of future scenarios, ensuring they are not only theoretically sound but also behaviourally realistic. However, HiL methods also entail certain limitations: participants may experience simulator sickness or reduced immersion, which can constrain the validity of their observed behaviours. In addition, the logistical effort required to recruit diverse participant groups limits the scalability of HiL studies. Beyond direct validation, HiL outcomes can be systematically translated into agent-based simulations: observed behaviours are abstracted into decision models, parameter ranges, or probabilistic rules, which can then enrich large-scale, automated simulations. In this way, human insights extend computational models, while simulations provide scalability and coverage.

In summary, the treatment of future scenarios necessitates an exploratory and iterative methodology. Rather than aiming for static definitions, the process emphasises the continuous generation of hypotheses, their testing in both human-centred and computational environments, and the refinement of assumptions based on observed outcomes. Human-in-the-Loop simulations are particularly useful in this regard, as they reveal behavioural dynamics that may otherwise remain hidden, thereby ensuring that scenario design remains realistic and adaptive. By systematically feeding these insights back into scenario development, the research community strengthens the robustness of autonomous systems and fosters a methodological framework capable of addressing the uncertainties of future mobility.

4 CONCLUSIONS

This deliverable has defined a comprehensive and harmonised methodology for scenario identification, extraction, enrichment, and generation within the SYNERGIES project. By structuring the scenario lifecycle across multiple abstraction levels and grounding it in an ontology-based scenario concept, D5.1 establishes a coherent and interoperable framework that enables systematic, reproducible, and scalable scenario-based evaluation of CCAM systems.

The proposed methodology integrates data-driven and knowledge-based approaches, covering the full transformation chain from Scenario Source Data to executable test scenarios. Through clearly defined processes such as semantic compression, non-ordinary situation identification, accident-based scenario derivation, and parameterised scenario generation, the framework ensures traceability between observed real-world situations, abstract and logical representations, and concrete simulation-ready artefacts. This structured approach supports both coverage-driven testing and targeted safety assessment for specific Operational Design Domains and systems under test.

A key strength of the methodology lies in its alignment with existing standards and interoperability requirements. By relying on standardised scenario representations and metadata formats and by defining consistent interfaces between processing steps, the methodology enables seamless integration with other SYNERGIES work packages. In particular, it provides the foundation for scenario analysis and KPI-based evaluation activities in WP6 and ensures direct executability of generated scenarios within the SYNERGIES platform and marketplace developed in WP7.

Beyond current validation needs, the inclusion of Scenarios of Tomorrow introduces a forward-looking dimension to the methodology. By explicitly addressing anticipated changes in traffic systems, mobility concepts, infrastructure, and behaviour, the framework supports future-proof scenario development and contributes to the long-term relevance of scenario-based safety assurance approaches.

he concepts, workflows, and requirements defined in this deliverable form the methodological backbone of WP5. They will be further operationalised, implemented, and validated through the tools and demonstrations described in subsequent WP5 deliverables. Together, these results establish a robust, extensible, and interoperable basis for scenario-based safety assessment, supporting the overall SYNERGIES objective of advancing scalable and trustworthy validation methodologies for connected, cooperative, and automated mobility systems.

Future work will focus on optimizing generation pipelines for large-scale deployment. Continued alignment with the latest standards and expansion of human factors integration will further enhance the realism and applicability of the methodology.

5 REFERENCES

[1] ASAM e.V., *ASAM OpenSCENARIO,* 2025.

[2] HEADSTART, “HEADSTART,” [Online]. Available: https://www.headstart-project.eu/.

[3] SUNRISE, “SUNRISE,” [Online]. Available: https://ccam-sunrise-project.eu/.

[4] L3Pilot, “L3Pilot,” [Online]. Available: https://l3pilot.eu/index.html.

[5] Hi-Drive, “Hi-Drive,” [Online]. Available: https://www.hi-drive.eu/.

[6] PEGASUS, “PEGASUS,” [Online]. Available: https://www.pegasusprojekt.de/en/home.

[7] ADScene, “ADScene,” [Online]. Available: https://www.irt-systemx.fr/en/a-la-une-en/autonomous-vehicle-completion-of-the-experimental-programme-sam-and-launch-of-first-pilot-services/.

[8] ISO, “International Organization for Standardization,” [Online]. Available: https://www.iso.org/home.html.

[9] ASAM, “ASAM,” 2026. [Online]. Available: https://www.asam.net/.

[10] ASAM e.V., *ASAM OpenDRIVE,* 2025.

[11] ASAM e.V., *ASAM Open Simulation Interface (OSI),* 2025.

[12] ASAM e.V., *ASAM OpenLABEL,* 2025.

[13] ASAM e.V., *ASAM OpenODD,* 2025.

[14] R. a. B. J. a. K. L. a. E. L. Krajewski, “The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems,” in *2018 21st international conference on intelligent transportation systems (ITSC)*, IEEE, Ed., 2018, pp. 2118--2125.

[15] J. a. K. R. a. M. T. a. R. S. a. V. L. a. E. L. Bock, “The ind dataset: A drone dataset of naturalistic road user trajectories at german intersections,” in *2020 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2020, pp. 1929--1934.

[16] W. a. S. L. a. W. D. a. S. H. a. C. A. a. N. M. a. K. J. a. K. H. a. S. C. a. d. L. F. A. a. T. M. Zhan, “INTERACTION Dataset: An INTERnantional Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps,” *arXiv:1910.03088 [cs, eess],* sep 2019.

[17] F. a. P. J.-H. a. J. J. a. O. S. a. N. M. a. K. F. a. M. M. Poggenhans, “Lanelet2: A high-definition map framework for the future of automated driving,” in *2018 21st international conference on intelligent transportation systems (ITSC)*, IEEE, 2018, pp. 1672--1679.

[18] 2022. [Online]. Available: https://www.iso.org/standard/78951.html.

[19] 2023. [Online]. Available: https://www.iso.org/standard/78952.html.

[20] “ASAM OpenSCENARIO DSL,” 2024. [Online]. Available: https://www.asam.net/standards/detail/openscenario-dsl/.

[21] ASAM, “OpenXOntology,” [Online]. Available: https://www.asam.net/standards/asam-openxontology/.

[22] P. a. F. Y. a. L. Z. a. Z. X. a. L. J. a. S. J. a. Z. Z. Ji, “Txt2Sce: Scenario Generation for Autonomous Driving System Testing Based on Textual Reports,” *arXiv preprint arXiv:2509.02150,* 2025.

[23] Z. a. N. L. a. F. W. a. Z. T. Tu, “Multi-modal Traffic Scenario Generation for Autonomous Driving System Testing,” *Proceedings of the ACM on Software Engineering,* vol. 2, pp. 1733--1756, 2025.

[24] Z. a. N. L. a. F. W. a. Z. T. Tu, “Multi-modal Traffic Scenario Generation for Autonomous Driving System Testing,” *Proceedings of the ACM on Software Engineering,* vol. 2, pp. 1733--1756, 2025.

[25] H. Weber, C. Glasmacher, M. Schuldes, N. Wagener and L. Eckstein, “Holistic Driving Scenario Concept for Urban Traffic,” in *2023 IEEE Intelligent Vehicles Symposium (IV)*, 2023.

[26] H. Weber, C. Glasmacher, M. Schuldes, N. Wagener and L. Eckstein, Holistic Driving Scenario Concept for Urban Traffic, 2023.

[27] A. Thorsén, O. Bartels, C. Berger, T. Bouraffa, C. Kaya, M. Muro, A. Bolovinou, I. Panangiotopoulos, D. Becker, A. Farooqui, M. Skoglund, P. Stålberg, F. Warg, E. Arnoux and O. d. Cam, “D3.1 Report on baseline analysis of existing Methodology. Deliverable of the Horizon Europe project SUNRISE.,” 2024.

[28] J. Beckmann, X. Zhang, E. de Gelder, E. van Hassel, M. Nieto, G. Ben Nejma and M. Grabowski, “D5.2 Harmonised descriptions for content of CCAM safety assessment data framework. Deliverable of the Horizon Europe project SUNRISE.,” 2025.

[29] M. Brini, E. Arnoux, B. Foyer, G. Bresson, L. Durville, F. Gaillard and M. Pajon, “MOOVE & MOSAR Projects: a scenario library for designing & validating ADS,” in *Driving Simulation Conference (DSC)*, Antibes, France, 2020.

[30] Y. T. Azar, J. D. Ortega, M. Nieto, J. Beckmann, J. D. Mendoza, E. Mungan, D. Khtatba and C. Tlili, “D6.3 First European Test Case Library. Deliverable of the Horizon Europe project SUNRISE.,” 2025.

[31] E. Revert, L. Dulewicz, M. El Zeenny, K. Reisinger, M. Schuldes, E. Daskalaki, S. Tarlowski, A. Bolovinou, O. Yildirim, M. R. Martínez, J. Vanschoren, M. Alirezaei, M. Pechenizkiy, E. van Hassel, B. Zonooz, S. Garcia, T. Griffon, B. Kolbay, S. Di Piazza, E. de Gelder, P. Irvine, F. Warg, J. Jeyachandran, L. Friedmann and X. Zhang, "D2.2 Storylines definition and technical and interoperability requirements. Deliverable of the Horizon Europe project SYNERGIES.," 2025.

[32] C. Glasmacher, M. Schuldes, P. Topalakatti, P. Hristov, H. Weber and L. Eckstein, "Scenario-based Model of the ODD through Scenario Databases. Deliverable 13 of the VVM project.," 2023. [Online]. Available: https://www.vvm-projekt.de/veroeffentlichungen.html. [Accessed 12 September 2025].

[33] R. Villalba, J. Rodríguez, X. Li, L. Friedmann, J. Hidalgo, A. Kloeker, F. Alakkad, C. Glasmacher, J. Martiniak, L. Vater and E. Revert, "D2.1 Stakeholder high-level requirements," SYNERGIES, 2024.

[34] M. Sonntag, H. Weber, S. Rahmani, E. d. Gelder and L. Eckstein, "Hi-Drive Driving Scenario Concept. Publication of the Horizon Europe project Hi-Drive," 2023.

[35] M. E. Scholtes, L. Westhofen, L. R. Turner, K. Lotto, M. Schuldes, H. O. Weber, N. Wagener, C. Neurohr, M. H. Bollmann, F. Körtke, J. M. Hiller, M. Hoß, J. Bock and L. Eckstein, "6-Layer Model for a Structured Description and Categorization of Urban Traffic and Environment," *IEEE Access,* vol. 9, pp. 59131-59147, 2021.

[36] M. Schuldes, C. Val and C. van Driesten, "Scenario Source Data – Specification of Data Model and Format (OMEGA-PRIME)".

[37] C. a. S. M. a. E. L. Glasmacher, "Comparison of Parametrization Approaches for Scenario-Based Testing," 2023.

[38] B. S. Institution, "Behaviour taxonomy for Automated Driving System (ADS) applications," 2024.

[39] A. A. B. a. I. P. a. Z. X. a. K. S. a. J. P. da Costa, "Ontology-based Scenario Generation for Automated Driving Systems Verification and Validation using Rules of the Road," 2024.

[40] C. W. L. K. T. M. E. B. E. &. H. A. Neurohr, "On Scenario Formalisms for Automated Driving," 2025.

[41] C. W. H. S. M. W. N. &. E. L. Glasmacher, "Generation of concrete parameters from logical urban driving scenarios based on hybrid graphs," 2023.

[42] C. S. M. T. P. H. P. W. H. &. E. L. Glasmacher, "Scenario-based model of the odd through scenario databases," 2023.

[43] C. Glasmacher, H. Weber, M. Schuldes and L. Eckstein, "Towards a Completeness Argumentation for Scenario Concepts," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, 2024, pp. 1106-1111.

[44] F. H. Y. &. B. X. Wang, "Infrastructure-enabled GPS spoofing detection and correction," in *IEEE Transactions on Intelligent Transportation Systems, 24(12), 13878-13892*, 2023.

[45] L. Klitzke, C. Schicktanz, G. Kay and M. Henning, "TASI: A Python Library for Traffic Data Analysis and Situation Interpretation".

[46] U. E. P. S. D. S. H. B. S. B. J. M. T. G. C. U. M. E. C. C. H. D. R. N. N. M. S. G. L. M. J. O. d. C. O. C. E. K. G. Sander, "Data Sources for Baseline Generation – Overview, Grading, and Recommendations. Deliverable D4.1 of the Horizon Europe project V4SAFETY.," 2024.

[47] S. K. A. W. a. P. J. E. Esenturk, "Analyzing Real-world Accidents for Test Scenario Generation for Automated Vehicles," in *IEEE Intelligent Vehicles Symposium (IV)*, Nagoya, Japan, 2021.

[48] "Stats19," [Online]. Available: https://www.gov.uk/government/publications/road-safety-statistics-stats19-review-update-and-future-development-roadmap/road-safety-data-and-statistics-stats19-review-update-and-future-plans.

[49] "GIDAS," [Online]. Available: https://www.gidas.org/start-en.html.

[50] E. a. S. T. a. H.-S. F. a. V. de Gelder, "Scenario Metrics for the Safety Assurance Framework of Automated Vehicles: A Review of Its Application," *Vehicles,* 2025.

[51] E. de Gelder and O. Op den Camp, "Scenario-Based Assessment of Automated Driving Systems: How (not) to Parametrize Scenarios?," in *IEEE International Automated Vehicle Validation Conference*, 2024.

[52] C. Glasmacher, J. Beckmann, D. Sander, M. Wisch and J. e. a. Bärgman, "Generating Baseline Scenarios for Predictive Safety Assessment. D4.2 of the Horizon Europe project V4SAFETY," 2025.

[53] D. J. D. T. G. S. Y. X. S.-V. A. L. &. S. S. A. Fremont, "Scenic: a language for scenario specification and scene generation," in *In Proceedings of the 40th ACM SIGPLAN conference on programming language design and implementation (pp. 63-78)*, 2019.

[54] W. M. E. &. R. A. Damm, "Traffic Sequence Charts for the ENABLE-S3 Test Architecture. Results of the ENABLE-S3 Project.," 2019.

[55] L. N. C. K. T. B. M. S. B. U. F. .. .. .. B. E. Westhofen, "Criticality Metrics for Automated Driving: A Review and Suitability," in *Archives of Computational Methods in Engineering*, 2023.

[56] C. W. L. H. T. d. G. T. M. E. &. B. E. Neurohr, "Fundamental Considerations around Scenario-Based Testing," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020.

[57] C. W. L. B. M. B. M. H. E. U. &. G. R. Neurohr, "Criticality Analysis for the Verification and Validation of Automated Vehicles," in *IEEE Access*, 2021.

[58] F. Klück, Y. Li, M. Nica, J. Tao and F. Wotawa, “Using ontologies for test suites generation for automated and autonomous driving functions,” in *2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, Memphis, TN, USA, 2018.

[59] I. ISO 21448, “Road vehicles — Safety of the intended functionality,” ISO, Geneva, 2022.

[60] M. a. G. P. Bäumler, “Test scenario fusion: How to fuse scenarios from accident and traffic observation data,” 2023.

[61] N. T. C. &. K. U. Weber, “A Needle in a Haystack--How to Derive Relevant Scenarios for Testing Automated Driving Systems in Urban Areas,” 2021.

[62] C. S. M. T. P. H. P. W. H. &. E. L. Glasmacher, “Scenario-based model of the ODD through scenario databases,” 2023.

[63] SYNERGIES, “D3.2 Assessment of available data from external sources,” 2025.

[64] SUNRISE, “D4.4 Report on the Harmonised V&V simulation framework,” 2024.

[65] C. Glasmacher, M. Sonntag and L. Eckstein, “Comparison of Parametrization Approaches for Scenario-Based Testing,” in *Proceedings of the 11th International Conference on Vehicle Technology and Intelligent Transport Systems - VEHITS*, 2025.

[66] E. N. Euro NCAP, “Vision 2030: A safer future for mobility,” [Online]. Available: https://cdn.euroncap.com/media/74468/euro-ncap-roadmap-vision-2030.pdf. [Accessed 2022].

[67] I. O. f. Standardization, “Road vehicles—Functional safety—Part 1: Vocabulary (ISO 26262-1:2018),” [Online]. Available: https://www.iso.org/standard/68383.html.

[68] P. a. K. J. a. B. F. a. W. M. a. R. E. a. B. M. Hock, “How to Design Valid Simulator Studies for Investigating User Experience in Automated Driving,” in *Proceedings of the 10th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, New York, NY, USA: ACM, 2018, pp. 105--117.

[69] N. Wagener, F. Russ, J. Beckmann and L. Eckstein, “ika’s Highly Dynamic Driving Simulator,” in *Driving Simulation Conference (DSC) Europe VR 2023*, 2023.

[70] SYNERGIES, “D2.2 Storylines definition and technical and interoperability requirements,” 2025.

A. ABBREVIATIONS AND DEFINITIONS

The terms and definitions can be found within [70].

Term	Definition
AD	Autonomous Driving
ADAS	Advanced Driver Assistance Systems
ADS	Automated Driving System
AI	Artificial Intelligence
ASAM	Association for Standardization of Automation and Measuring Systems
CCAM	Connected, Cooperative, and Automated Mobility
CDF	Common Data Format
EC	European Commission
eHMI	External Human-Machine Interface
HiL	Hardware-in-the-Loop
HMI	Human-Machine Interface
IR	Intermediate Representation
ISO	International Organization for Standardization
LLM	Large Language Model
MCAP	Message Capture format used for storing time-synchronised scenario data streams
ODD	Operational Design Domain
OMEGA-PRIME	Common Data Format used in SYNERGIES for storing and exchanging Scenario Source Data
OpenDRIVE	ASAM standard for the description of static road infrastructure and road networks
OpenLABEL	ASAM standard for the representation of semantic labels and metadata associated with scenarios
OpenODD	ASAM standard for the formal, machine-readable description of an Operational Design Domain
OpenSCENARIO	ASAM standard for the machine-readable description of dynamic driving scenarios
OSI	Open Simulation Interface (ASAM standard for simulator interoperability)

SoT	Scenario of Tomorrow
SSD	Scenario Source Data
SUT	System Under Test
TSC	Traffic Sequence Chart
TTC	Time-to-Collision
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Infrastructure
WP	Work Package