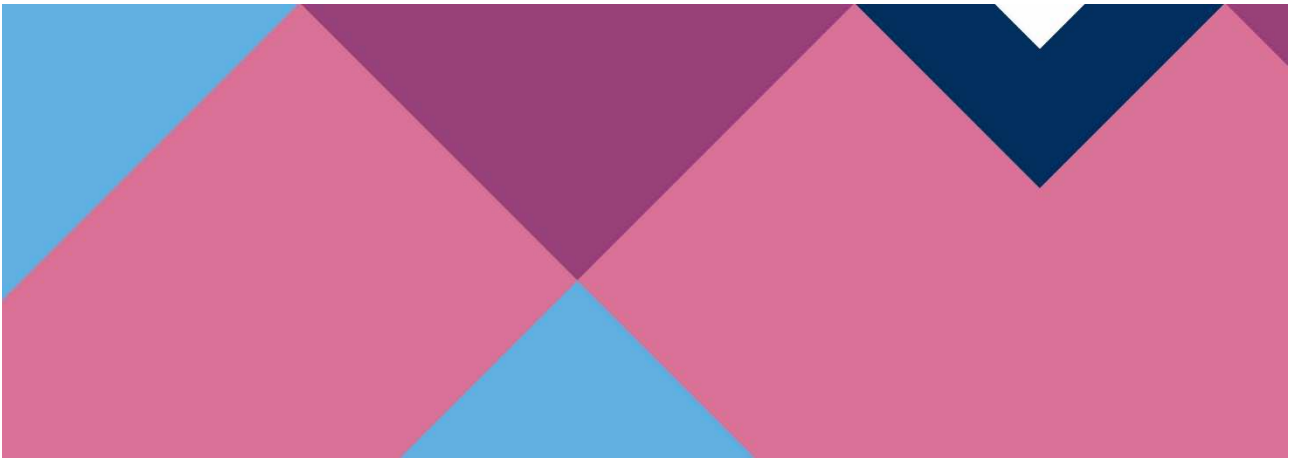


► **VVM Deliverable 03**

Assurance framework and new methodologies within the V & V structure



Version 1.0

Editor Jürgen Nuffer

Project coordination Robert Bosch GmbH and BMW AG

Due date 31/07/2023

Creation date 31/08/2023

Date of publication 05/09/2023



Gefördert durch:



aufgrund eines Beschlusses
des Deutschen Bundestages

Document information

Authors

Jürgen Nuffer, Matthias Rauschenbach, Simon Kupjetz (LBF)

Jan Reich, Tobias Braun (IESE)

Martin Mai (ZF)

Julian Pott (Ford)

Reviewer

Jürgen Nuffer

Contact

Jürgen Nuffer

Fraunhofer LBF

Bartningstr. 47

64289 Darmstadt

Phone: +49 (0)6151 705-281

Email: juergen.nuffer@lbf.fraunhofer.de

Revision log

| Version | Date | Comment | Author | Partner |
|---------|------------|------------------------------|-----------------|-----------|
| 0.1 | 20.04.2022 | Template prepared | Nuffer | LBF |
| 0.2 | 10.05.2022 | Contrib. System requirements | Mai | ZF |
| 0.3 | 30.06.2022 | Contrib. GQM | Pott | Ford |
| 0.4 | 11.08.2022 | Contrib. probFMEA/CFT | Kupjetz, Braun | LBF, IESE |
| 0.5 | 22.11.2022 | Contrib. Assurance Framework | Reich | IESE |
| 1.0 | 31.08.2023 | Final chapters and typo | Kupjetz, Nuffer | LBF |

Table of Content

| | |
|---|-----------|
| 1 Introduction | 7 |
| 2 Assurance Framework | 9 |
| 2.1 Stakeholders and their argumentation needs | 9 |
| 2.2 Absence of unreasonable risk and its aspects | 10 |
| 2.3 Generic argument decomposition strategy based on safety concern types | 11 |
| 2.4 ADS assurance decomposition strategy | 12 |
| 2.5 VVM Safety Assurance Framework | 13 |
| 2.6 Exemplary concerns for core artifacts | 16 |
| 2.7 Outlook | 17 |
| 3 V & V Structure | 18 |
| 3.1 General | 18 |
| 3.2 Branches of decomposition | 19 |
| 3.3 Decomposition for verification | 20 |
| 3.4 How the structuring of V & V tackles the challenges | 22 |
| 4 Goal – Question – Metric (GQM) | 23 |
| 4.1 Introduction and State of the art | 23 |
| 4.2 Development within VVM | 23 |
| 4.3 Creating and maintaining the Model | 24 |
| 4.4 Models created in the VVM Context | 24 |
| 4.5 Usage within VVM | 24 |
| 4.6 Outlook | 25 |
| 5 Probabilistic FMEA and Component Fault Trees | 26 |
| 5.1 State of the art and purpose | 26 |
| 5.2 Development within VVM | 27 |
| 5.3 Safety Analysis | 30 |
| 5.4 CFT Analysis | 30 |
| 5.5 ProbFMEA Analysis | 32 |
| 5.6 Prototype Safety Box Implementation | 32 |
| 5.7 Summary safety analysis | 34 |
| 6 System requirements | 35 |
| 6.1 General considerations | 35 |
| 6.2 Development within VVM | 35 |
| 6.3 System Requirements Definition | 36 |
| 7 Conclusion and Outlook | 38 |

Table of figures

| | |
|---|----|
| Figure 1: Stakeholders and their argumentation needs | 9 |
| Figure 2: Exemplary decomposition of a global safety goal into subgoals | 10 |
| Figure 3: Strategy for argument decomposition regarding fundamental safety concern types | 11 |
| Figure 4: Three-circle model with gaps to be argued (left) and their assignment to the levels of VVM safety argumentation (right) | 12 |
| Figure 5: The VVM safety assurance framework in context..... | 13 |
| Figure 6: Capability, engineering and real-world perspectives as structuring scheme for the three big ADS assurance challenges | 14 |
| Figure 7: Core artifacts allocated to assurance structuring scheme | 14 |
| Figure 8: Concerns of capability layer artifacts to be addressed in safety argument..... | 16 |
| Figure 9: Concerns of engineering layer artifacts to be addressed in safety argument..... | 16 |
| Figure 10: Concerns of real world layer artifacts to be addressed in safety argument..... | 17 |
| Figure 11: V & V branch within the assurance framework | 18 |
| Figure 12: Actual result of the structuring of V & V | 19 |
| Figure 13: Detail of the decomposition on the engineering layer | 20 |
| Figure 14: Flux of information along the engineering layer a) leading to test requirements and b) backwards. C) shows another detail of test orchestration and test execution | 21 |
| Figure 15: Introduction of a new traffic sign only affects relevant branches on the engineering layer, leaving the capability layer unaffected..... | 22 |
| Figure 16: Exemplary GQM model..... | 23 |
| Figure 17: Model excerpt for GQM..... | 24 |
| Figure 18: General input-output structure of the safety analysis procedure based on probFMEA and CFT..... | 28 |
| Figure 19: Analysis steps for the probFMEA / CFT methodology across the different development views..... | 28 |
| Figure 20: Exemplary probFMEA failure net..... | 29 |
| Figure 21: Exemplary modeling of a lidar sensor's weakness and triggering conditions..... | 33 |
| Figure 22: Filter mechanism "Include/Exclude typed events" | 33 |
| Figure 23: Use case-based filter "SOTIF Failure without Measure" | 34 |
| Figure 24: Exemplary Stakeholder Needs..... | 36 |
| Figure 25: Exemplary System Requirements | 37 |

List of tables

| | |
|---|----|
| Table 1: Requirements with relation to Model | 25 |
| Table 2: Description of different event types | 31 |

1 Introduction

This deliverable displays a first methodological approach on the structuring of the V & V space. In this course, it describes several aspects of conceptual descriptions. At first and most important part of this concept, the so-called assurance framework is presented. The fundamental purpose of the assurance case for the absence of unreasonable risk in an open context is to satisfy the needs of several different stakeholders. Thus, looking at the argumentation's addressees and their expectations is an essential task before creating the argumentation structure. The VVM Safety Assurance Framework provides methodological guidance to collect key safety concerns relevant to the safe operation in the open context, systematically structure these concerns with the expected argumentation lines of stakeholders in mind and to determine required core evidence, which can address the concerns appropriately on the one hand and that can realistically be provided by engineering and assurance processes.

Based on the assurance framework as core-part of this deliverable, the V & V space is structured. During the elaboration phase of the structuring of V & V it became evident that verification and validation both are part of the safety assurance but follow different paths in the framework and need different quality measures. Therefore, this deliverable mainly follows the verification path, while the validation path will be completed in a later deliverable together with results based on the argumentation, which will bring verification and validation in relation to each other.

In the verification, three branches of decomposition are identified, these are the decomposition of safety goals, of the system design and the scenarios. In order to derive system and in consequence test qualities as a basis for the verification, several methodologies were developed in the project. In this deliverable, it is described how these different methodologies are organized within that framework and the development status of these methodologies themselves is also displayed.

The first methodology highlighted here is the so-called "Goal Question Metric". It is a method that has been developed to measure goals of organizations and their projects. In the context of VVM we use the first three steps to derive quantitative answers to questions. These questions are derived from multiple inputs. Inputs that are part of the development process of a system, as well as social, regulatory and other inputs to the system as a product interacting with the world. At this time, we have looked at the following inputs, followed the GQM process and thus catalogued metrics for them: 1) requirements derived from the Capability-Based Architecture, 2) Known measures like time to collision, 3) Component capabilities, e.g. LIDAR talking to experts which relevant questions arise towards the operational usage of lidar when used in an automotive application. These quantifiable answers deliver metrics which supports and defines a baseline on how the system needs to be designed and tested.

Secondly, new methodologies were found to be necessary within the topic of safety analysis, since the amount of data to be processed within that framework will exceed the capabilities of actual tools like conventional FMEA. Thus, a novel methodology based on a so-called probabilistic FMEA (probFMEA) combined with Component Fault Trees (CFT) was developed within VVM. The safety analysis uses CFT and ProbFMEA modeling to prove that the safety objective is met, which arises in particular from SOTIF issues. Regardless of the modeling level, be it the capability level, the functional architecture or the system level, it is investigated whether and which safety mechanisms are active for certain cause combinations. Furthermore, probabilities are used to determine

whether a cause combination is sufficiently relevant for consideration. The methods of CFT analysis and ProbFMEA complement each other here. The CFT analysis examines the cut-sets and determines that safety target violations are sufficiently improbable. Possible relevant SOTIF faults are covered by safety measures in each case. The ProbFMEA method, on the other hand, shows that faults from SOTIF weaknesses and from safety measures do not cause a safety violation. A prototype implementation of the CFT analysis for the safeTbox development tool was also created to demonstrate its feasibility. In this deliverable, the principal procedure of safety analysis based on the novel tools is described by an example.

Finally, based on the developed methodologies, first system requirements were derived as a basis for functional, logical and technical architecture (which will be shown in a later deliverable). Up to this point the requirements specification consisted of functional requirements describing the intended behavior of the system. With help of the Goal-Question-Metric (GQM) approach first quality measures were defined. They were expressed as performance requirements refining the functional requirements. An excerpt of the actual list of system requirements is presented.

The deliverable ends with a conclusion and an outlook. As an outlook, it is described that this deliverable lays the ground for the next steps which will mainly consist in the derivation of the functional, logical and later the technical architecture (i.e., decomposition of system design), the derivation of test requirements based on the actual system requirements as well as the decomposition of scenarios. In later future, the validation path must be derived in a similar manner and brought together with the verification by means of the argumentation.

2 Assurance Framework

2.1 Stakeholders and their argumentation needs

The fundamental purpose of the assurance case for the absence of unreasonable risk in an open context is to satisfy the needs of several different stakeholders. Thus, looking at the argumentation's addressees and their expectations is an essential task before creating the argumentation structure.

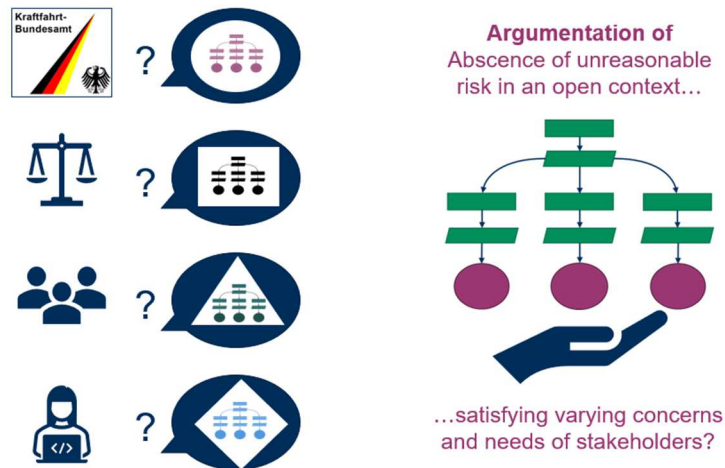


Figure 1: Stakeholders and their argumentation needs

The most important stakeholders are:

Certification Authorities are interested in demonstrated compliance with normative aspects of existing safety standards and regulations.

Legal Authorities are interested in structuring pieces of evidence along argumentation lines standard in legal environments to enable legal personnel to understand safety arguments from a non-technical perspective. Moreover, legal authorities are typically involved in *specific cases*, e.g. to legally examine the causes of and responsibility for crashes. These needs are very different from those of certification authorities, where an argument should focus on evidence about why *all relevant cases* have been considered.

General society is interested in a comprehensible argument for a positive risk balance, which demands evidence about the safety performance of ADS technology compared to the human driver reference system in addition to addressing functional safety, the safety of the intended functionality (SOTIF) and other essential aspects of unreasonable risk.

Original equipment manufacturers (OEMs) and their suppliers are interested in the effective generation and management of the assurance case, which enables the type of approval and trust in safe systems over the entire product lifecycle.

The stakeholders have different interests; therefore, no concrete argumentation structure exists that can satisfy all stakeholders simultaneously. Therefore, VVM did not aim to provide a concrete argument structure directly. Instead, we developed a framework to systematically collect relevant stakeholders and their concerns and structure them in a reference model, from which concrete arguments can be derived for particular ADS systems.

2.2 Absence of unreasonable risk and its aspects

The top goal is to demonstrate the absence of unreasonable risks generated by the ADS. Unreasonable risks arise from different aspects (Figure 2), their respective demonstrations of absence are referred to here as subgoals. The risks arise from:

- technical defects
- functional deficiencies
- insufficient compliance with traffic and liability laws
- failure to meet the ethical expectations of society and its requirements for accident reduction to be achieved by the ADS.

VVM focuses on these four categories in the project. From the abstract global safety goal of “absence of unreasonable risks,” plausible verbal subgoals are derived. These are then mapped to indispensable framework conditions for the establishment of methodical development processes, to necessary capabilities or relevant target behavior of the ADS, or to process requirements for validation and verification, as well as to metrics and benchmarks.

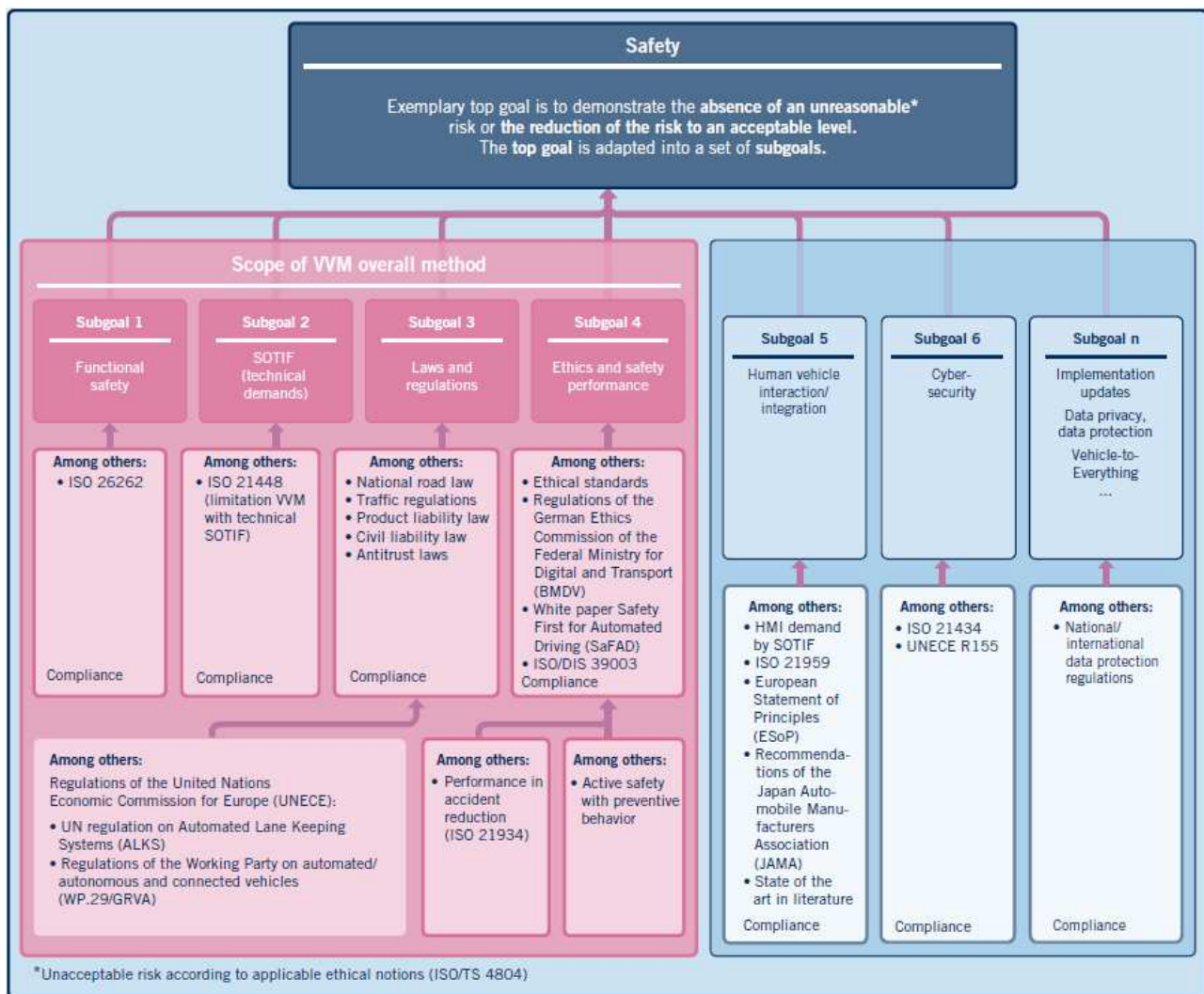


Figure 2: Exemplary decomposition of a global safety goal into subgoals

In order to systematically achieve a complete and actionable goal decomposition and to be able to derive and structure the evidence required for this purpose, the VVM assurance framework is used (see Section 2.5). The safety concerns relevant for the open-world context are collected, structured and classified with respect to the expected lines of argumentation of relevant addressees of the safety case. The evidence that can be realistically presented with existing engineering and assurance methods is determined.

2.3 Generic argument decomposition strategy based on safety concern types

In the VVM argumentation strategy, the top-level argument is decomposed into normative, objective, and subjective branches (Figure 3).

The normative branch provides a positive argument for meeting normative aspects from applicable laws and standards – both during development and for operation. However, for several reasons, an ADS system is not necessarily safe enough, even if all normative aspects have been addressed: First, not all aspects of unreasonable risks are covered by exhaustive standards today. Second, technological development happens usually faster than regulation and standardization. Thus, we conclude that normative compliance is necessary to achieve safety in real world, but not sufficient.

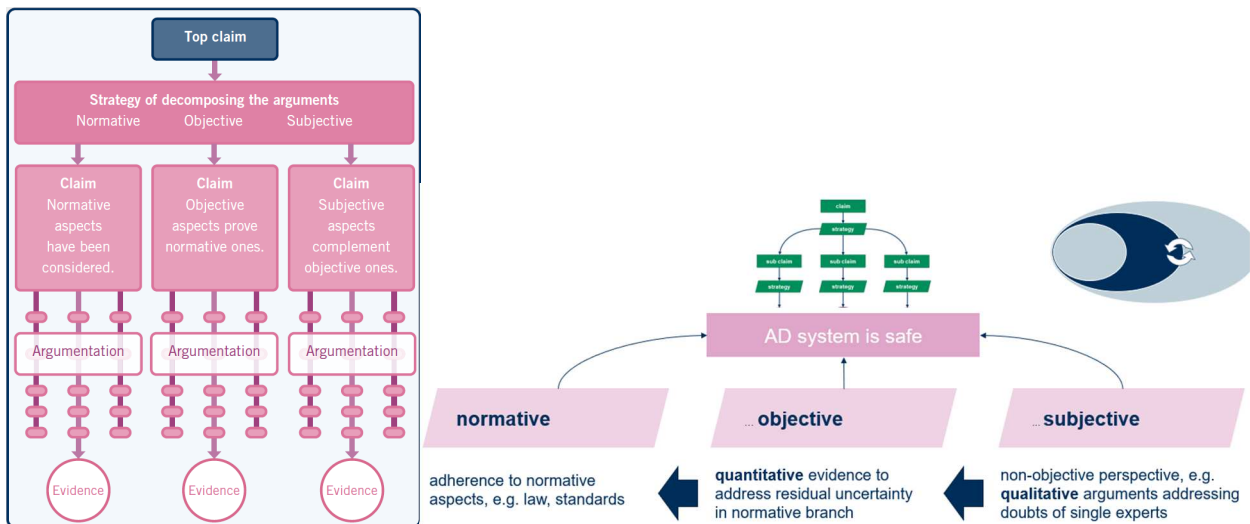


Figure 3: Strategy for argument decomposition regarding fundamental safety concern types

For this purpose, the objective branch takes a negative argumentation perspective and argues by means of quantitative evidence the appropriate consideration of safety concerns on why residual unreasonable risk does not remain in the product after following laws and standards. Since not all existent concerns can be addressed by objective quantitative evidence, the subjective argument branch takes a non-objective perspective that gives room for addressing non-quantifiable concerns of single experts by means of qualitative arguments.

2.4 ADS assurance decomposition strategy

In addition to the different generic sources of safety concerns, which are independent of the ADS system class, this section classifies the ADS-specific engineering and assurance challenges by means of the conceptual three-circle model according to Stellet et al.¹ (Figure 4, left). This enables a divide-and-conquer safeguarding approach. In it, the triangular relationship between required, specified, and actual behavior and the resulting gaps are presented, with a focus on the concerns imposed by those gaps that may arise in the development and verification/validation process:

- The specification gap must be argued through a systematic determination of the required behavior and its correct specification.
- The implementation gap closes through a correct behavior specification to be implemented and its systematic transformation into a product with actual and, thus, verifiable behavior.
- The validation gap assumes the existence of a real system and addresses the control of unexpected behavior.

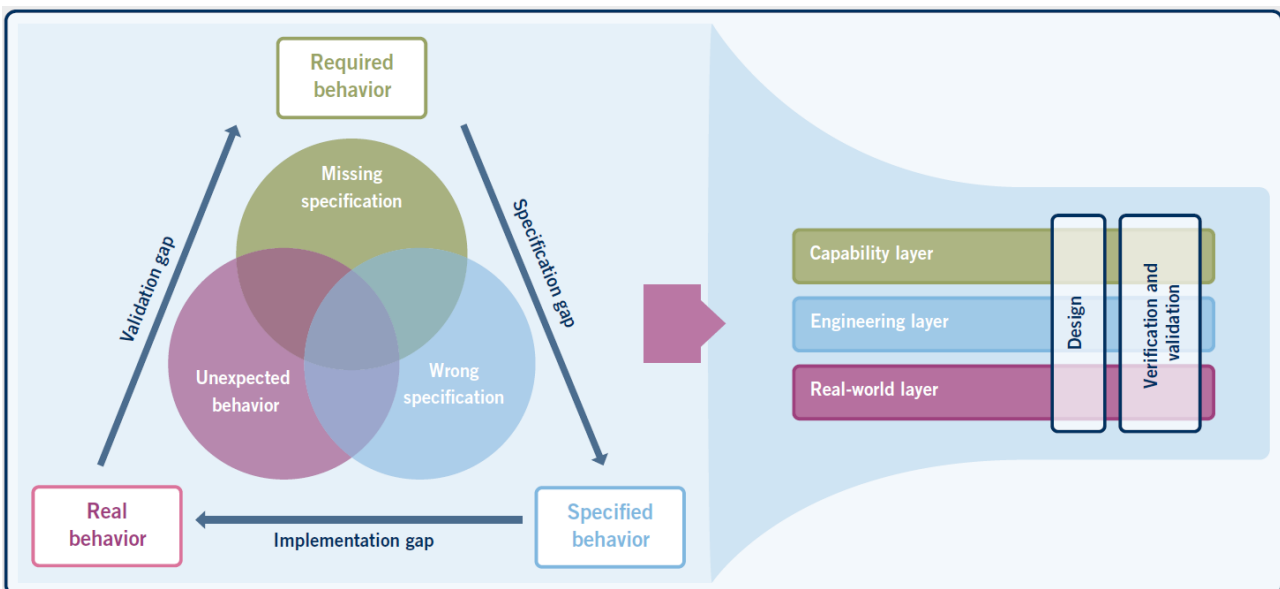


Figure 4: Three-circle model with gaps to be argued (left) and their assignment to the levels of VVM safety argumentation (right)

The classification of challenges forms the conceptual basis for the structure of the VVM safety assurance framework. The specification gap is addressed at the capability level, the implementation gap at the engineering level, and the validation gap at the real-world level (Figure 4, right).

¹ Stellet, J. E. et al.: Formalisation and algorithmic approach to the automated driving validation problem. 2019 IEEE Intelligent Vehicles Symposium (IV), Paris

2.5 VVM Safety Assurance Framework

Context

The VVM Safety Assurance Framework (Figure 5) provides methodological guidance to

1. Collect key safety concerns relevant to the safe operation in the open context.
2. Systematically structure these concerns with the expected argumentation lines of particular stakeholders in mind and
3. Determine required core evidence, which can address the concerns appropriately on the one hand and that can realistically be provided by engineering and assurance processes.

Note that these steps are not strictly executed in this order. Especially steps 2 and 3 influence each other: Possible argumentation lines are usually dependent on the presence of particular evidence and vice versa. Therefore, an iterative approach is necessary.

An argumentation framework, i.e. a set of consistent argumentation principles and strategies to be instantiated for different parts of the ADS engineering method, is not sufficient for providing a convincing safety case alone, as the adequacy of the safety argument depends on the concerns arising from particular artifacts produced for by methods and processes. Thus, an explicit interface between argumentation and detailed engineering/assurance methods has been created, which enables to express argumentation patterns aligned around core artifacts of the overall methodology and their relation – the “common ground”, which is more and more agreed upon in industry and academia. One such core methodological element is the scenario-based validation approach.

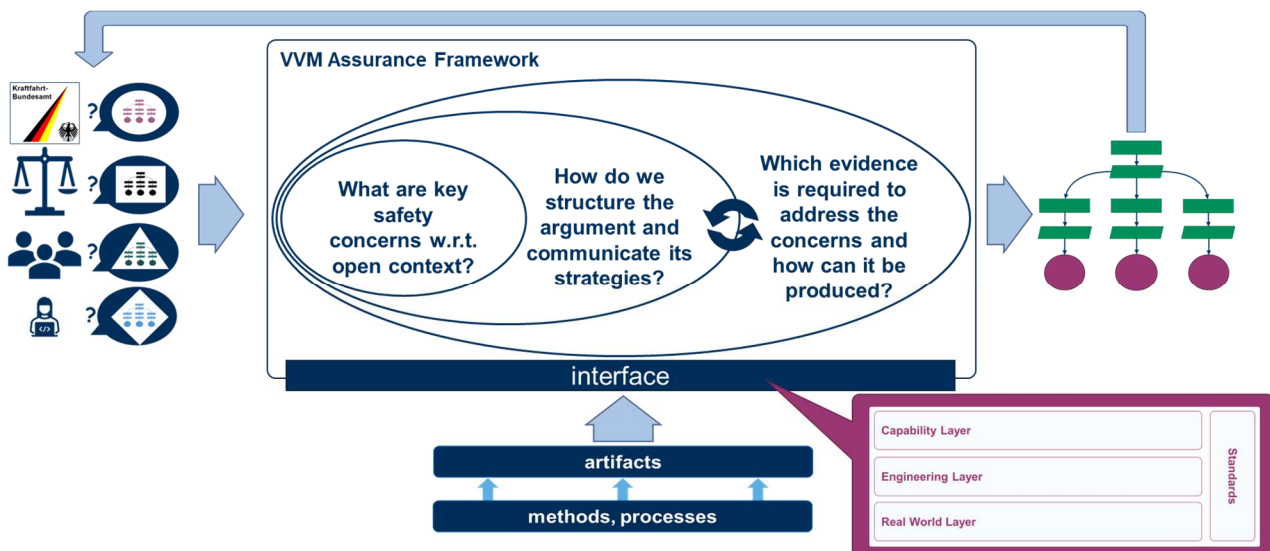


Figure 5: The VVM safety assurance framework in context

The VVM Safety Argumentation Framework contains a concrete proposal for how a reference model for such a common ground can look like.

For the derivation of this reference model, several core principles were used:

1. A suitable level of abstraction to argue the decomposition of the open context to achieve complete coverage of artefacts without losing comprehensiveness
2. the usage of an architectural approach as integral part of the safety argument to achieve inherent traceability between argumentation, methods and evidence
3. compatibility with existing relevant industry standards

We must argue that the system in its environment is...

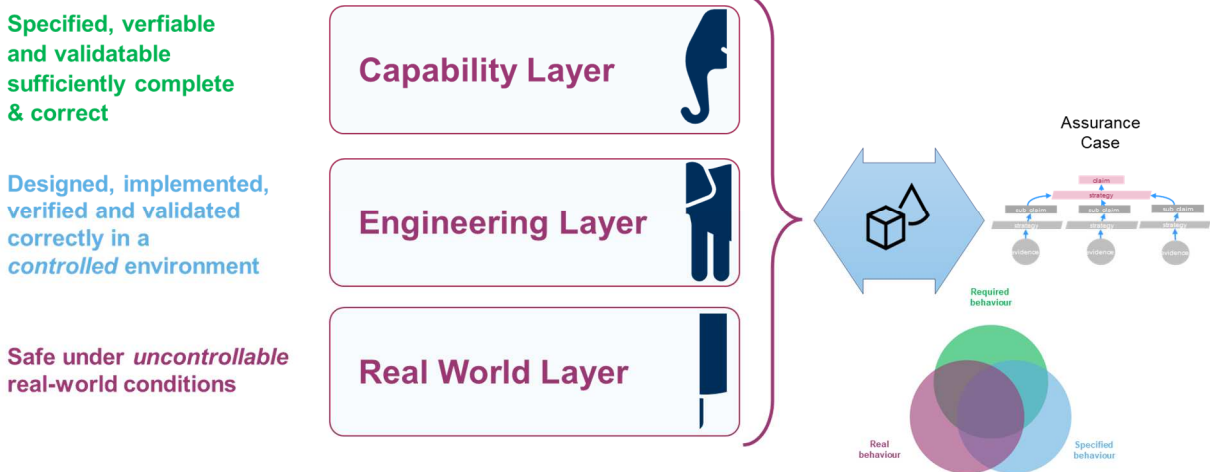


Figure 6: Capability, engineering and real-world perspectives as structuring scheme for the three big ADS assurance challenges

The reference model (Figure 7) organizes core artifacts and their relations in the context of the ADS engineering and V&V method along the three perspectives representing major assurance challenges (Figure 6).

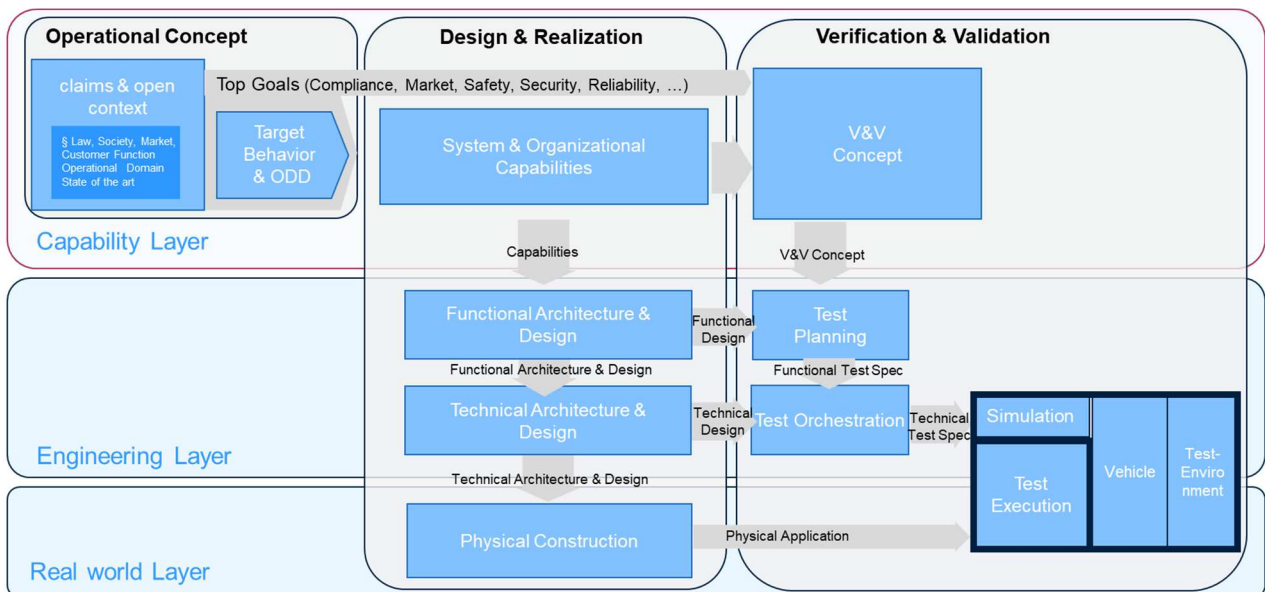


Figure 7: Core artifacts allocated to assurance structuring scheme

Capability Perspective

The Capability Layer provides a perspective for the derivation of capabilities of the organizations and systems. Capabilities enable a methodical approach to effective release argumentation at an abstract level.

The Capability Layer artifacts provide the basis for an abstract argument as to why the capabilities of the system to realize safe behavior in the open context are sufficiently specified and why the V&V concept can provide sufficient evidence to deliver those capabilities.

Core aspects

Delineation of engineering layer: capability layer is abstract layer describing problem domain, engineering layer is technical layer describing solution domain

Consideration of organization and ADS with same capability concept

Engineering Perspective

The engineering layer provides a perspective to represent the solution architecture of organization and system. This view-oriented architecture enables effective release argumentation at the technical level.

The artifacts of the Engineering Layer provide the basis for an argument why the generated solution leads to the delivery of the capabilities by means of systematic engineering methods in the modeled context and why the results of the implemented V&V concept ("Technical V&V Design") can be trusted with respect to the delivery of the capabilities in the modeled context.

Core aspects

- Focus on solution space
- Controlled environment through modeled context based on assumptions
- "classical" view-based automotive systems engineering
- technical design of capabilities for vehicle behavior and runtime monitoring/learning

Real-World Perspective

The Real World Layer provides a perspective to represent activities in order to provide empirical evidence with the physical system in the real environment, which continuously prove or disprove the validity of the assumptions from Capability and Engineering Layer ("Known Unknowns"). In addition, unpredictable developments/events of the open context occurring in the real environment are fed back into the development cycle in a controlled manner ("Unknown unknowns").

The Real World Layer artifacts provide the basis for an argument that addresses the interaction of the physical system with the real world and argues why an acceptable residual risk exists with respect to the validity of the assumptions from Capability & Engineering Layer.

Core aspects

- Distinction from Engineering Layer: Eng. → solution in controlled environment, Real-World → solution in real environment.
- Test vehicle in real-world (influence by manufacturer where to drive) vs. Deployed fleet (no influence where to drive).

2.6 Exemplary concerns for core artifacts

The VVM safety assurance framework and in particular its artifact reference model enables to systematically raise concerns to be addressed in the safety argumentation. Figure 8, Figure 9 and Figure 10 provide examples of concerns to be addressed for elements in all three perspectives.

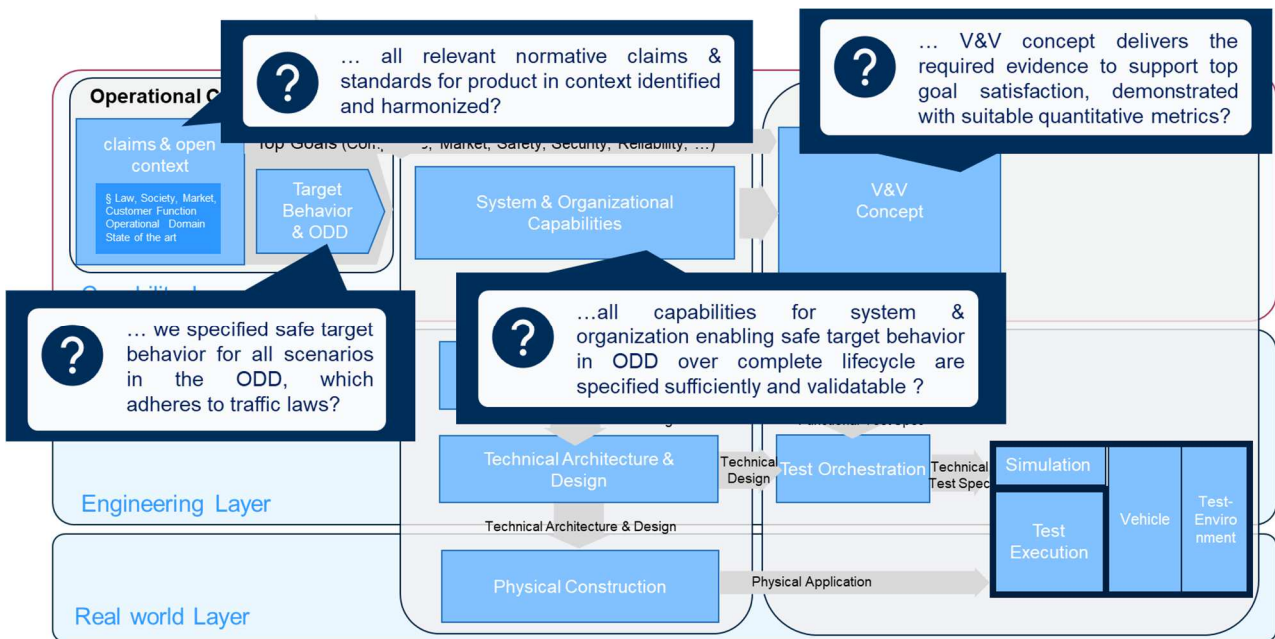


Figure 8: Concerns of capability layer artifacts to be addressed in safety argument

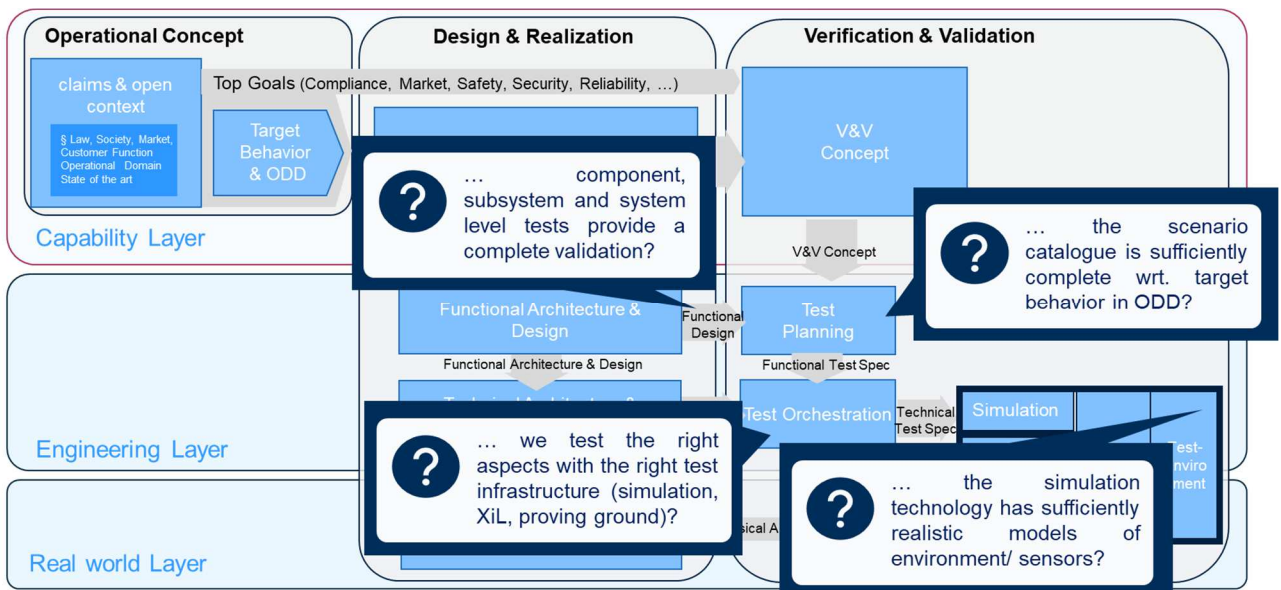


Figure 9: Concerns of engineering layer artifacts to be addressed in safety argument

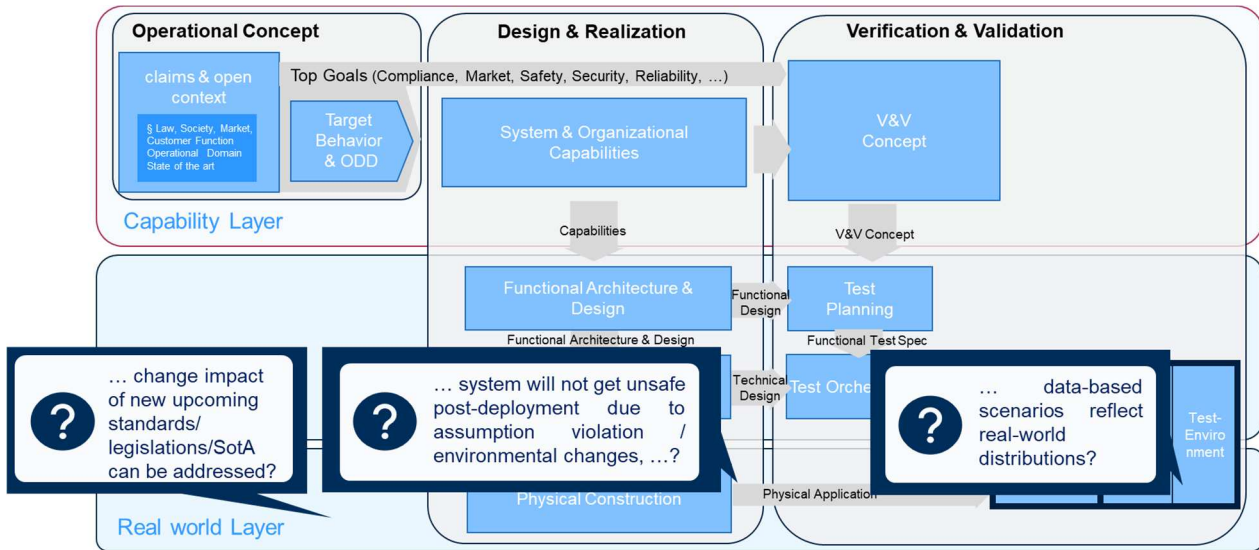


Figure 10: Concerns of real world layer artifacts to be addressed in safety argument

2.7 Outlook

In the remaining project period, a concrete top-level argumentation structure will be created based on the safety assurance framework elements outlined in this section. The argumentation structure will consist of a GSN module structure, where argument patterns will be proposed to capture concrete argumentation strategies and evidence types to address the sub goals of Figure 2, with a focus on sub goals 2 (SOTIF compliance) and 3 (Law and regulation adherence).

3 V & V Structure

3.1 General

Within the Assurance Framework, the V&V structure covers the part of the assurance framework highlighted in Figure 11.

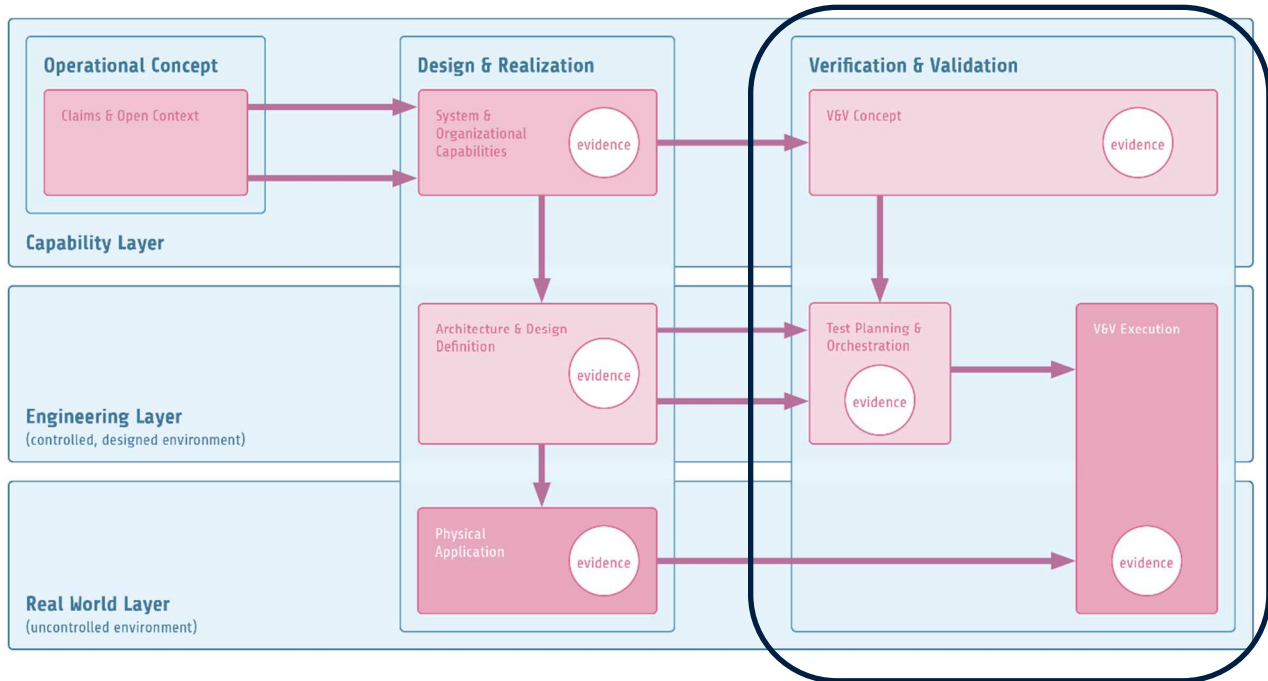


Figure 11: V & V branch within the assurance framework

The main aim of the V & V branch is to **deliver evidence for the safety argumentation**.

To understand the motivation for the approach followed in the VVM project, at first the main challenges of the V&V for an ADS are described as follows:

- **Feasibility** in Open context forces V&V to cover a huge number of variations of conditions (test coverage).
- **Changeability** forces seamless V&V for tailored systems (subsystems/components/variants) and forces seamless V&V for environmental changes (by open context) a also preservation of test-results of unmodified components.
- **Efficiency** forces the use of test instances according to their strength and seamless test-integration including of virtual and real artefacts.
- **Conformity** forces V&V to deliver evidence for argumentation of safety including safety goals.

The V & V structure explained here aims at

- Introducing several methodological approaches needed on the way to deliver system- and test requirements.
- Allocate these methods at the three layers introduced in the previous chapter.
- Therefore, serve as a link between the assurance framework and the methods described in the following chapters.

3.2 Branches of decomposition

Within this section, it is described how the highlighted part of the assurance framework is structured and how this structuring lays the ground for the V & V process to be derived. In so far this section has a twofold purpose: At first, it describes the information flow within the V & V branch, on the other hand it places several methodologies within that workflow, which will be described in the subsequent chapters.

In the actual approach, there are three branches of decomposition, as can be seen in Figure 12.

These branches are the decomposition along the

- System design
- Safety goals/Qualities
- Scenarios

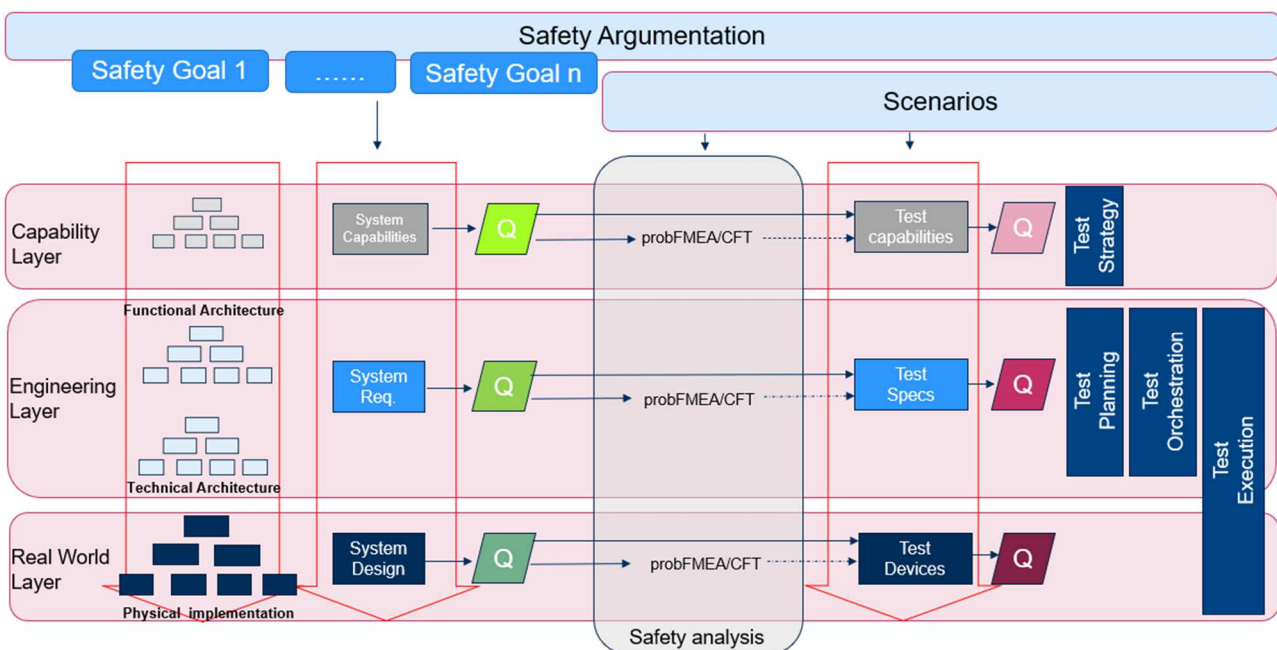


Figure 12: Actual result of the structuring of V & V

These branches are highlighted in vertical red arrows in this picture. In order to create system and test requirements along the three layers after the decomposition is done, we will define

- System qualities
- Test Qualities,

indicated as greenish and reddish “Q’s” respectively, within the above Figure 12. These qualities are the main basis for the system design and for the deriving of goals for testing. How these qualities are derived and processed will be described in chapter 5, where a methodology known as the “Goal – Question – metric” approach (GQM) will be discussed.

While deriving the test qualities on the different layers, an important additional aspect is defined by the safety analysis. The safety analysis itself will also contribute to some extent to the test specification. Since the safety analysis of autonomous systems is confronted with additional challenges compared to conventional systems, there are also some new developments on the methodological side. Therefore, in chapter 6 a novel methodological approach based on a so-called probabilistic FMEA combined with component fault trees (probFMEA/CFT) is presented.

3.3 Decomposition for verification

In the VVM project, both verification and validation must be addressed. As a first step, we will concentrate on the process of verification here to highlight the flow of information within the established framework. In Figure 13 a zoom on the engineering layer is shown, now showing the refined system structure on the left-hand side as well as the corresponding test qualities on the hierarchical levels of the system breakdown.

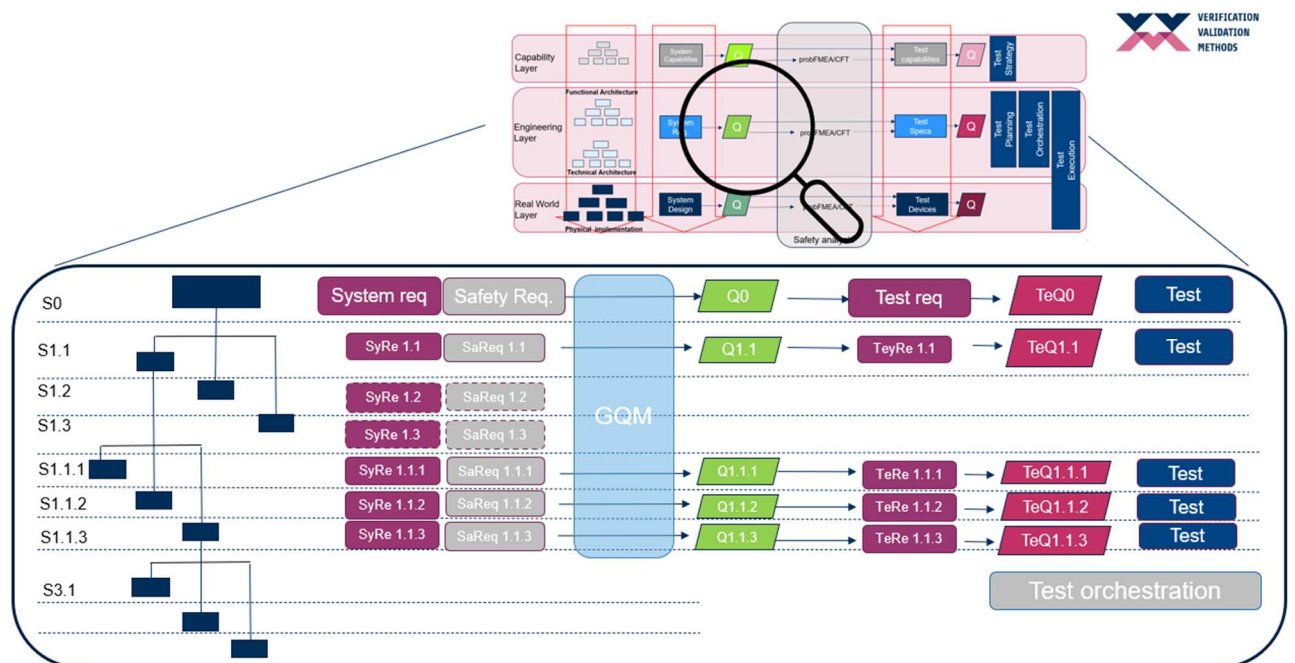
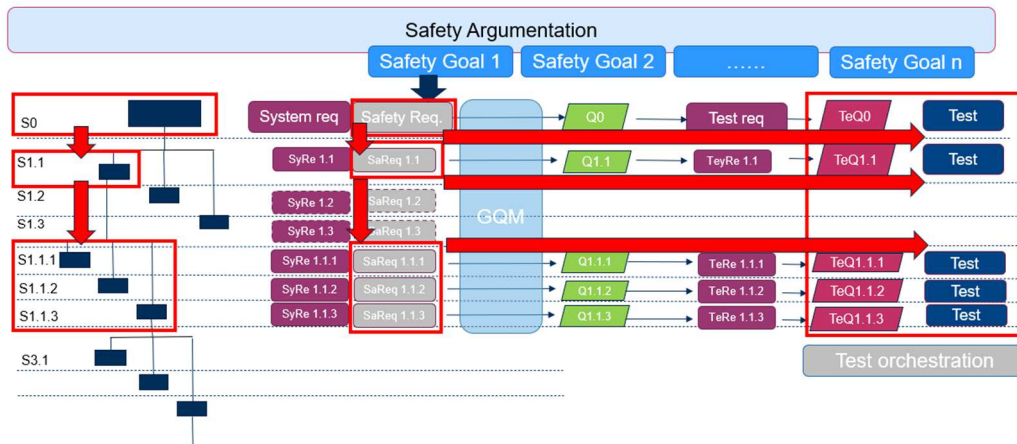
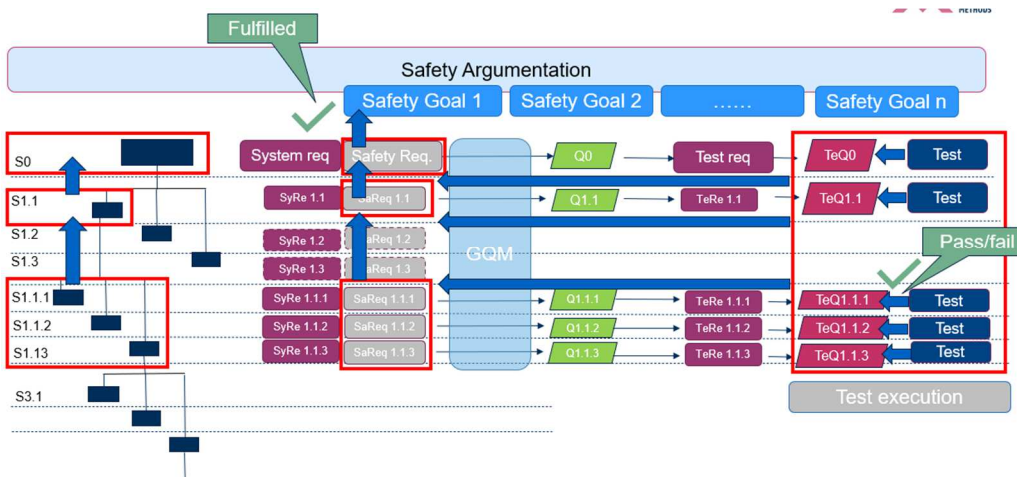


Figure 13: Detail of the decomposition on the engineering layer

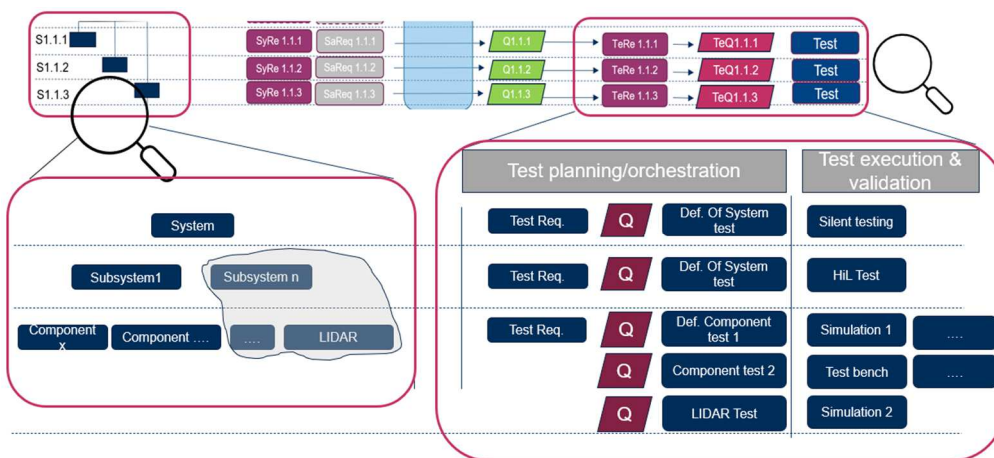
As can be seen in Figure 14, the decomposition of the safety goals into requirements is a together process along with the decomposition of the system architecture and the scenario breakdown (not explicitly sketched here).



a)



b)



c)

Figure 14: Flux of information along the engineering layer a) leading to test requirements and b) backwards. C) shows another detail of test orchestration and test execution

3.4 How the structuring of V & V tackles the challenges

The strategy followed in the projects contributes to the solution of the above-mentioned challenges in the following way:

- **Feasibility** in Open context forces V&V to cover a huge number of variations of conditions (test coverage).

This challenge is taken into consideration within the development of the probFMEA/CFT method, since these methods are, in contrast to classical methods, able to deal with the high amount of data. It also delivers a contribution to the question in how far the selected tests cover a satisfying range of the ODD. (see chapter 5)

- **Changeability** forces seamless V&V for tailored systems (subsystems/components/variants) and forces seamless V&V for environmental changes (by open context) and also preservation of test-results of unmodified components.

If we assume for instance that a new traffic sign is introduced as sketched in Figure 15, then it is of importance to ensure that the established homologation process must not be re-arranged completely from the very beginning. This is reached exactly by the concept of capability, engineering, and real-world layer. As can be seen in Figure 15, the introduction of the new traffic sign does not affect the settings already made on the capability layer but only the engineering layer, and here it will only affect those branches of the functional and technical architecture which are affected by this change. In other words, all the work that has already been done on the capability layer is unaffected by this procedure and can be kept identical no matter what changes are made in the environment of the ADS.

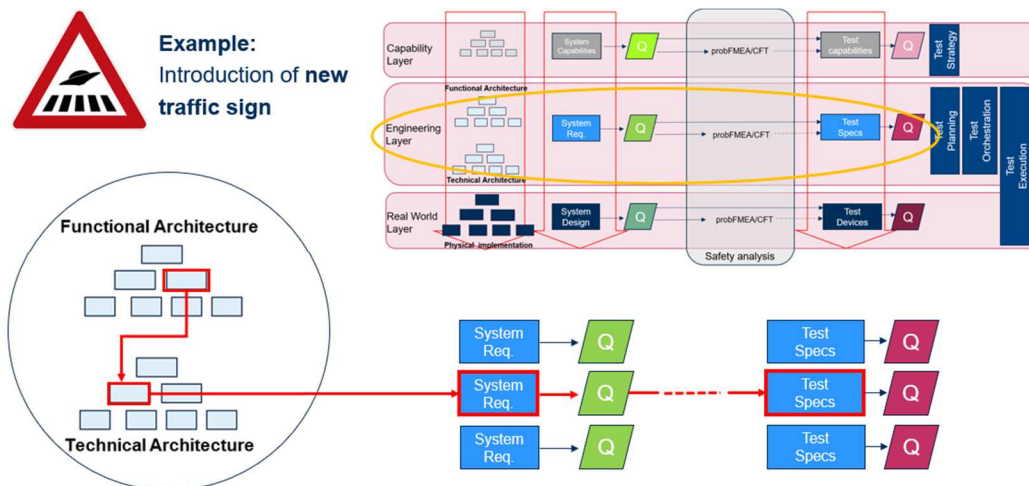


Figure 15: Introduction of a new traffic sign only affects relevant branches on the engineering layer, leaving the capability layer unaffected

The successful solution of the two remaining challenges, efficiency and conformity, will be a result of the further development of the project.

4 Goal – Question – Metric (GQM)

4.1 Introduction and State of the art

Goal Question Metric² is a method that has been developed to measure goals of organizations and their projects. An exemplary GQM model is shown in Figure 16. In the context of VVM we use the first three steps to derive quantitative answers to questions. These questions are derived from multiple inputs. Inputs that are part of the development process of a system, as well as social, regulatory and other inputs to the system as a product interacting with the world. These quantifiable answers deliver metrics which supports and defines a baseline on how the system needs to be designed and tested. An interaction between the GQM Model, which is a catalogue of these metrics, and requirements engineering as well as probFMEA & CFT has been presented at the halftime event of VVM.³

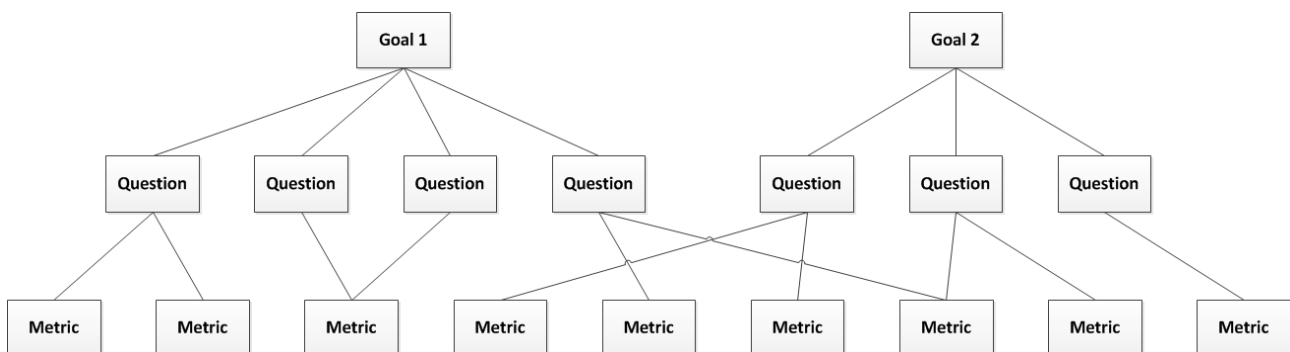


Figure 16: Exemplary GQM model

4.2 Development within VVM

At this time, we have looked at the following inputs, followed the GQM process and thus catalogued metrics for them:

- Requirements derived from the Capability-Based Architecture
- Known measures like *Time to Collision*
- Component capabilities,
 - o e.g. LIDAR: talking to experts which relevant questions arise towards the operational usage of lidar when used in an automotive application

These quantifiable answers deliver metrics which supports and defines a baseline on how the system needs to be designed and tested.

² Basili, Caldiera, Rombach (Encyclopedia of Software Engineering – 2 Volume Set, 1994) <http://www.cs.umd.edu/~basili/publications/technical/T89.pdf>

³ An Approach for Decomposition and Analysis”, J. Pott, M. Rauschenbach, et.al, VVM Midterm March 2022

4.3 Creating and maintaining the Model

A goal is defined by its purpose, an Issue, related object, or process, and finally a viewpoint. As an example, this would be “Improve – the understanding of – GQM – for the reader of this chapter”.

Towards the given goal multiple questions can be defined and once that has been done metrics can be developed to answer these questions. Either some or all of these metrics then can be used to answer questions regarding a goal, within any consuming process in the development project.

Maintenance of the model then follows the iterative needs of the overall project.

4.4 Models created in the VVM Context

At this time, there are 2 models created. One is an overall general model that has been created to deliver E4.1a, as well as an updated version going into E4.1d. The updated version also includes a secondary model which catalogues the GQM for a LIDAR. These models are currently written down in MS Excel tables.

4.5 Usage within VVM

Excerpts of both models have been modeled into the Prob FMEA & CFT tool SafeTBox to have one integrated model when it comes to FMEA, CFT and metrics from GQM.

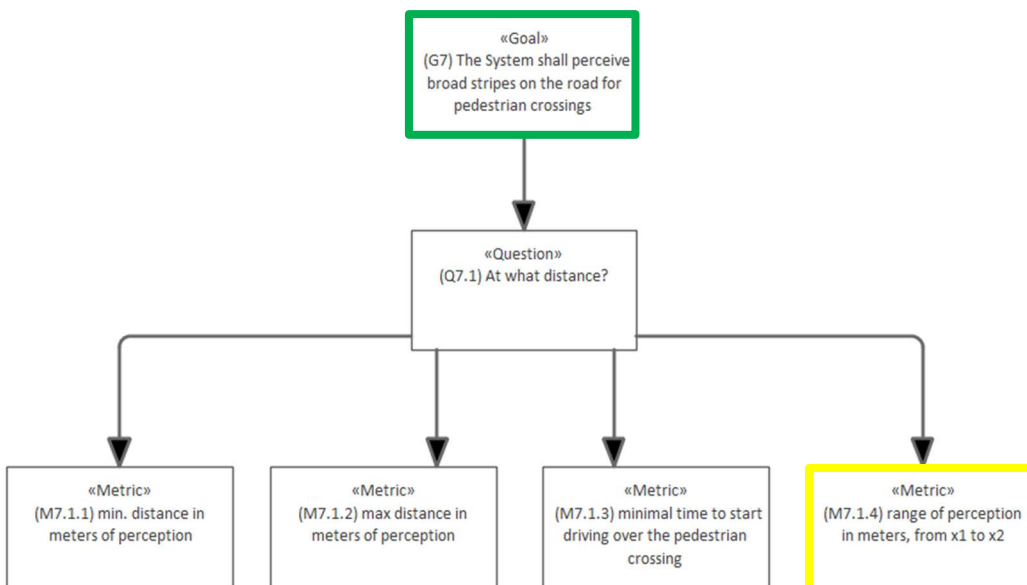


Figure 17: Model excerpt for GQM

With regards to requirements development, the model has also been used as an input to derive non-functional requirements for given functional requirements. If there was no matching metric for a given functional requirement, the GQM model (like it is shown in Figure 17) was extended to develop a metric which then can be used as an input for a new non-functional requirement. An excerpt of the requirements with a relation to the model can be seen in Table 1.

Table 1: Requirements with relation to Model

| ID | Title | System Requirement | Req. Type | Refines |
|------------------|---|--|-------------|-----------------|
| SR-3.1.2 | crosswalk marking perception | The system shall perceive broad stripes on the road for <i>crosswalk markings</i> . | Functional | |
| SR-3.1.2a | crosswalk marking perception range | The system shall perceive crosswalk markings on the vehicle's traffic lane in a distance at least between x and y meters in the direction of driving. | Performance | SR-3.1.2 |

4.6 Outlook

The work will continue, extending the existing GQM model to serve the delivery of E4.1f. With regards to inputs and outputs to the GQM model the consumer scope will be extended to the colleagues working on testing. New inputs will also arise and thus extending the model, continuing the iterative nature of the model. In this regard there might be a focus on additional components but more social / regulatory sources. Finally, the description on how GQM fits into the overarching assurance framework and what it delivers to it.

5 Probabilistic FMEA and Component Fault Trees

5.1 State of the art and purpose

The following excerpt describing the methodology of the state of the art of functional safety in combination with HADS and the application of the probabilistic FMEA in combination with Component Fault Trees is part of an accepted paper at the TRA Lisbon 2022 “*Verification and validation of automated driving systems utilizing probabilistic FMEA and simulation approaches*”⁴.

The development of fully automated vehicles raises new challenges in terms of suitable procedures and techniques for analyzing the conceptual and functional design and their implementation. Among other things, there is a need for suitable methodological approaches for the analytical review and evaluation of the safe execution of appropriate behavior in all road traffic situations⁵.

For road vehicles, the standard for functional safety⁶ requires methods such as a hazard analysis and risk assessment (HARA), Fault Tree Analysis (FTA)^{7,8} and a Failure Mode and Effects Analysis (FMEA)^{9,10,11} for the verification of electrically and electronically implemented product functions with safety relevance. For automated vehicles, the perspectives and the current methodological systematics of functional safety are not sufficient. They cover only one substantial part of the vehicle’s behavioral safety in the open context of public traffic since these depend on an explicitly definable functional behavior under exhaustively defined boundary conditions and sequences of events. This includes all reasonably assumable component failures as well.

The SOTIF standard¹² defines a complementary approach to functional safety. Driving functions with lower degrees of automation are evaluated and verified regarding the safety of the intended functionality. This assessment of safety includes situation-dependent and principle-related weaknesses, e. g. in environmental monitoring or scenario assessment.

⁴ Bein, Thilo; Atzrodt, Heiko; Bartolozzi, Riccardo; Kupjetz, Simon; Millitzer, Jonathan; Nuffer, Jürgen et al.: Verification and validation of automated driving systems utilizing probabilistic FMEA and simulation approaches. In: Transportation Research Procedia 2022.

⁵ Rauschenbach, M., Kupjetz, S., Wolschke, Ch. and Braun, T., 2021. Approach towards the methodical analysis of the safety of the functional concept of fully automated vehicles. 30. VDI-Fachtagung Technische Zuverlässigkeit, April 27.-28, Nürtingen, Germany

⁶ ISO 26262 1-12, 2018. DIN ISO 26262-1:2018-12 Road Vehicle – Functional Safety, part 1-12. Beuth-Verlag, Berlin.

⁷ IEC, 2007. DIN EN 61025:2007-08, Fault tree analysis (FTA) (IEC 61025:2006); German version EN 61025:2007. Beuth-Verlag, Berlin

⁸ Watson, H. A., 1961. Launch control Safety Study. Bell Labs, Murray Hill, NJ, USA, 1961.

⁹ MIL-P-1629, 1963. Procedures for Performing a Failure Mode, Effects and Criticality Analysis. US Department of Defense.

¹⁰ IEC, 1991. DIN EN 60812 Analysis Techniques for System Reliability - Procedure for Failure Mode and Effect Analysis (FMEA). Beuth-Verlag, Berlin

¹¹ AIAG/VDA, 2019. Failure Mode and Effects Analysis (FMEA) Handbook, AIAG and VDA QMC.

¹² ISO/PAS 21448, 2019. ISO/PAS 21448:2019-01 Road vehicles - Safety of the intended functionality. Beuth-Verlag, Berlin.

In the BMWi project PEGASUS¹³, an approach for the "identification and quantification of automation risks for highly automated driving functions"¹⁴ was defined, which also enables the consideration of higher degrees of automation. Still, the SOTIF approach does not cover all viewpoints relevant for the verification of behavioral safety of fully automated driving, since its scope and concepts address the automation of specific driving functions with comparably limited degrees of freedom of behavior.

The systematics of the safety assessment for functional safety according to ISO 26262 and for the safety of the intended function (SOTIF) according to ISO 21448 cannot or only indirectly capture an essential functional component of highly automated vehicles directly. In the previous project PEGASUS, a system for hazard and error analysis regarding automation risks of driving functions with higher degrees of automation had already been developed¹⁴. This also includes the superordinate functional architectural view of the components in the analytical verification of the driving function.

For highly and fully automated vehicles, comprehensive capabilities must be defined in the system concept, which enable the vehicle to independently generate a behavior that is adapted to the situation, classify it as sufficiently safe and then implement it in an appropriate manner. In some scientific contributions, the systematic development based on capability models for automated vehicles is proposed^{15,16}. While previous approaches methodically determine possible errors with a focus on component defects as well as functional and specification errors and technological weaknesses, automated driving capabilities as carriers of the situation-dependent behavior of the automated vehicle are not dealt with. Possible deviations from the desired behavior and their causes are not considered. To the knowledge of the authors, an analytical approach to this does not yet exist.

5.2 Development within VVM

Possible error phenomena are described and evaluated analogously to the effect chain analysis in the PEGASUS approach¹⁴ with the methodology of component fault trees (CFT) and core elements of the formalism of probabilistic FMEA (probFMEA). Based on the hazard scenarios determined in a HARA, possible causes that can lead to these scenarios are determined in the system descriptions of the capabilities, functional structures, and component design. Figure 18 shows the general in-/output structure of the proposed procedure.

¹³ PEGASUS, 2019. PEGASUS Method – An overview. <https://www.pegasusprojekt.de/files/tmpl/Pegasus-Abschlussveranstaltung/PEGASUSGesamtmethode.pdf>, last visited April 28, 2022.

¹⁴ Böde, E., Büker, M., Damm, W., Fränze, M., Kramer, B., Neurohr, Ch., Vander Maelen, S., 2019. Identifikation und Quantifizierung von Automationsrisiken für hochautomatisierte Fahrfunktionen. PEGASUS Technical Report, OFFIS Institut für Informatik, 2019. (Online). Available: https://www.pegasusprojekt.de/files/tmpl/pdf/PEGASUS_TechnicalReport_Automationsrisiken_17.07.2019.pdf. (last access 16 02 2021).

¹⁵ Nolte, M., Bagschik, G., Jatzkowski, I., Stolte, T., Reschka, A. and Maurer, M., 2017. Towards a Skill- And Ability-Based Development Process for Self-Aware Automated Road Vehicles. IEEE 20th Int. Conf. on Intelligent Transportation Systems, October 16 – 19, Yokohama, Japan.

¹⁶ Reschka, A., Bagschik, G., Ulbrich, S., Nolte, M. and Maurer, M., 2015. Ability and skill graphs for system modeling, online monitoring, and decision support for vehicle guidance systems. IEEE Intelligent Vehicles Symposium, June 28 - July 1, Seoul, South Korea.

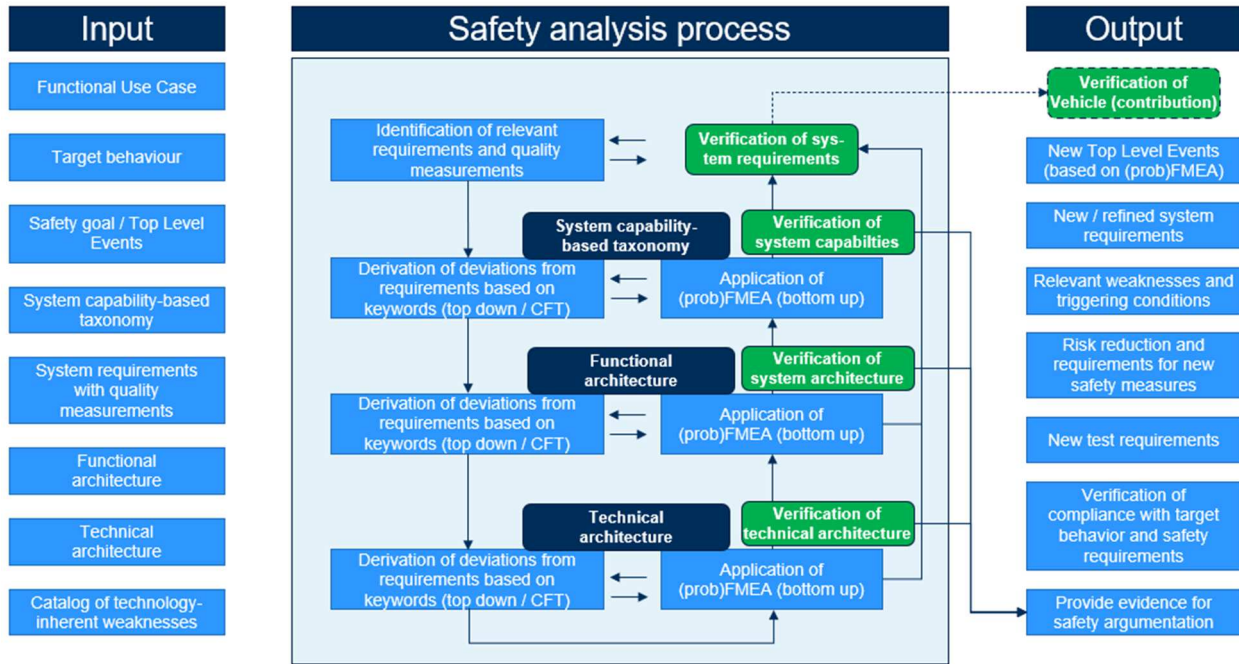


Figure 18: General input-output structure of the safety analysis procedure based on probFMEA and CFT

The basic principles of this approach were described in Rauschenbach (2021). In this publication, a focus is given to the probFMEA branch elucidating its inner structure and how it works in the context of this methodological framework. Figure 18 shows the typical bottom-up (probFMEA) and CFT (top-down) approach. The algebraic basis defined for the probFMEA allows the relationships to be determined in this way to integrate them with CFT, since their formalism is based on the same algebraic principles.

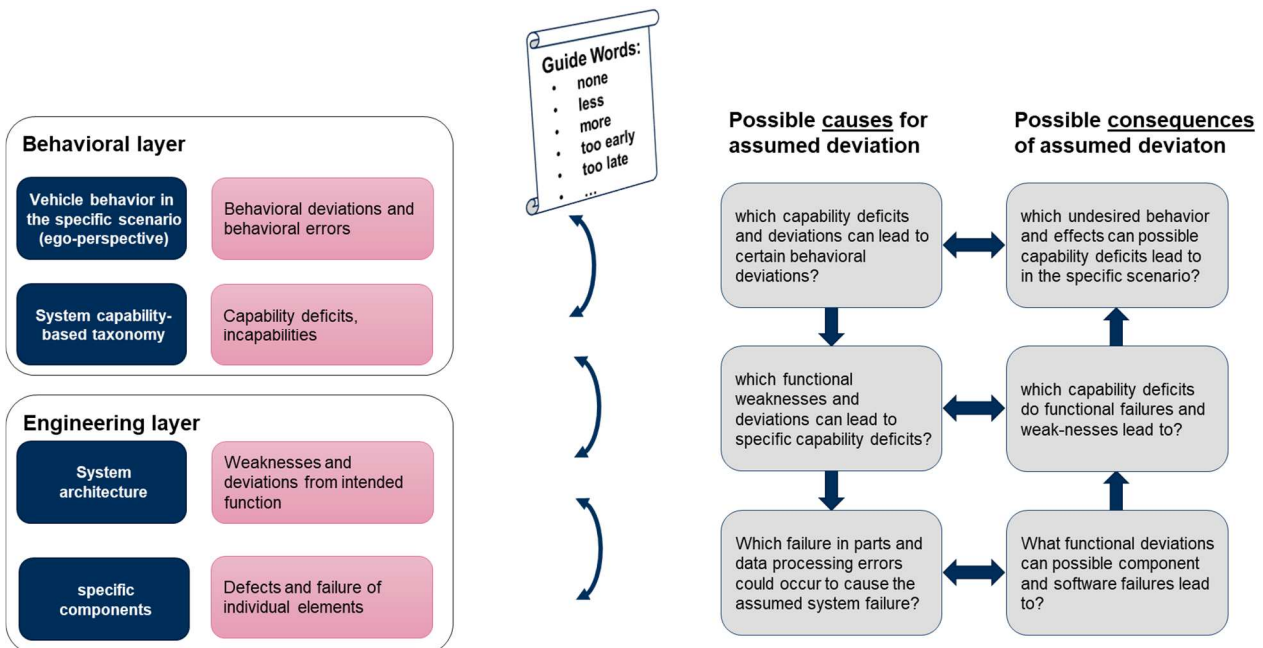


Figure 19: Analysis steps for the probFMEA / CFT methodology across the different development views

To perform this bottom-up-directed analysis step, possible error modes shall be determined in all system components that contribute to the functionalities required for the scenario under consideration. Their effects on the system behavior can be examined starting from these component errors (bottom-up). Figure 19 shows the steps that are taken to analyze the system across the different development views. If necessary, previously unconsidered basic events are subsequently included into existing CFTs or new, previously unidentified critical top-level events TLEs are defined for subsequent CFT analyses if required. This addition can increase the completeness of the analysis about a safety argument and verification based on it.

FMEA facilitates the coverage of all faults and consequences of a system in one database but does not allow for a quantitative evaluation of a system due to the lack of an algebraic foundation. Thus, the FMEA is merely qualitative. In contrast to this, each FTA failure model is focused on a single top-event. It defines one consequence for all identified and associated possible causes or combinations of causes. Due to the reason that FMEA and FTA offer different perspectives on a system's properties by examining basically the same information, their combined application is recommended or even mandatory in specifically relevant cases, such as safety-critical systems.

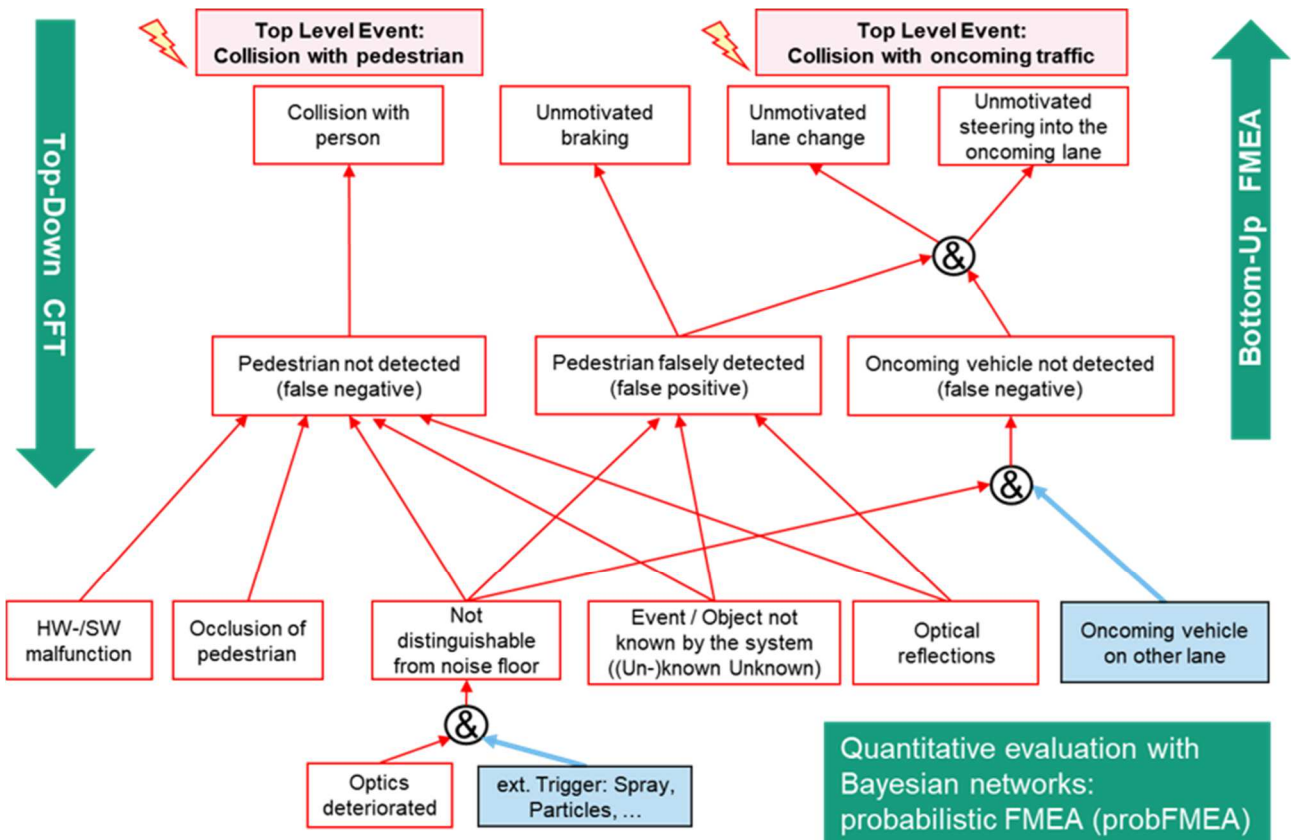


Figure 20: Exemplary probFMEA failure net

Thus, a failure analysis model concept featuring a holistic evaluation within consistent failure models for entire systems, appears advantageous. Rauschenbach elaborated the probFMEA approach in further detail along with its implementation and computation using Bayesian networks¹⁷. Further the usability and validity to represent consistent quantitative models of entire systems with holistic sets

¹⁷ Rauschenbach, M., Nuffer, J., 2019. Quantitative FMEA and Functional Safety Metrics Evaluation in Bayesian Networks. European Safety and Reliability Conference. September 24, Hannover, Germany.

of failure cause and consequence networks. An intermediate result obtained within the VV Methoden project is shown in Figure 20. It illustrates exemplarily a part of a failure network in the context of a probFMEA / CFT analysis for ADS. It contains multiple top-level events (TLE) logically connected with possible failure causes and their consequences. Following a practical example chosen in the VV Methoden project, the probFMEA is applied to an ADS approaching a crosswalk while a pedestrian is entering.

This advance represents a methodology to enable an analytical verification of HADS. The integral procedure combines fault trees and FMEA and means a reduction of effort and inconsistencies. The scenario-based behavior can be verified against the situationally required capabilities and enables an analysis of component weaknesses (SOTIF) and failures (functional safety). The methodology enables a coherent evaluation in a consistent database. It allows a qualitative determination of relevant triggering conditions and weaknesses through minimal cut-sets.

In the next step of such a methodological process, quantitative probabilities of occurrence of the different failure modes will be introduced. Their evaluation can be computed based on Bayesian networks, yielding the total probability of the TLE leading to a violation of each safety goal in one coherent model. Based on the safety requirements associated with each safety target, it is possible to determine if whether it is met accordingly, or else, whether refinements of the system concept and design are to be specified. This may be addressed on the behavioral, architectural, functional, or component level, as is appropriate for each case.

5.3 Safety Analysis

The safety analysis uses CFT and ProbFMEA modeling to prove that the safety objective is met, which arises in particular from SOTIF issues. Regardless of the modeling level, be it the capability level, the functional architecture or the system level, it is investigated whether and which safety mechanisms are active for certain cause combinations. Furthermore, probabilities are used to determine whether a cause combination is sufficiently relevant for consideration. The methods of CFT analysis and ProbFMEA complement each other here. The CFT analysis examines the cut-sets and determines that safety target violations are sufficiently improbable. Possible relevant SOTIF faults are covered by safety measures in each case. The probFMEA method, on the other hand, shows that faults from SOTIF weaknesses and from safety measures do not cause a safety violation. A prototype implementation of the CFT analysis for the safeTbox development tool was also created to demonstrate its feasibility.

The CFT analysis, ProbFMEA method, implementation, and a summary of the safety analysis are presented in the following.

5.4 CFT Analysis

The advantage of CFT analysis is the determination of the causes of safety objective violations. By assigning components to fault trees, the triggering components can be determined and traceability to the architecture can be achieved. Top-down creation of CFTs models complex fault trees that become manageable through component mappings. For the safety analysis, the Minimal Cut Set (MCS) analysis known from ordinary fault trees can be applied. This determines the smallest possible sets of events that can lead to the triggering of the top event, i.e. to the violation of the safety objective. As an extension of the usual MCS analysis, CFT's fault tree and thus also a cross-component view of the respective possible causes is generated. By using typed Basic Events it is possible to distinguish the following classes for SOTIF analyses, which is shown in Table 2.

Table 2: Description of different event types

| Basic Event Type | Description | Comment |
|--------------------|---|--|
| E/E-Failure | The classical electric or electronical fault, as it is known from ISO26262. | This type enables to random-failure models with SOTIF-related models. |
| Triggering Event | Triggering Events may trigger weaknesses of a system. | Different Triggering Events might be responsible to trigger a weakness of the system |
| Weakness | A weakness is an inherent reason, that a function of the system may not work as intended. | The weakness may arise from sensor or processing insufficiencies. |
| Triggered Weakness | A triggered weakness indicates the event, that a system may suffer the corresponding insufficiency. | The triggered weakness is usually sufficient to cause a safety goal violation. Hence, safety mechanisms are required to handle weaknesses. |

The goal of the analysis is to provide evidence that a safety objective violation is sufficiently absent. From classical safety analysis, it is recognized that single faults must be safeguarded with a safety mechanism. Multiple faults are to be safeguarded only if there is a common cause. The minimal cut set analysis provides the combinations of basic events that lead to a top event, i.e. to a violation of the safety requirement or the safety objective. We analyze the possible combinations to determine whether additional or different safety measures are needed. Among the possible combinations that may arise as a result of the MCS, there are the following possibilities:

- **There is exactly one E/E error without failure of an associated measure:** A safeguarding of the E/E error is necessary to sufficiently exclude a safety objective violation. Such a safeguarding corresponds to the state-of-the-practice for ISO26262 relevant systems.
- **Multiple E/E errors (independent of possible errors in safety measures) without SOTIF-related events:** Since E/E errors occur randomly in the system, separate protection is not necessary. Common-cause errors, which require a safety measure, are an exception here.
- **Triggered Weakness without erroneously inactive safety measure:** A distinction must be made here as to whether the case is practically relevant or not. If there is relevance, there should be a corresponding safety measure. For each relevant combination of Triggered Weaknesses, there should be a safety measure to ensure that no safety objective violation occurs. If a combination of triggered weaknesses is not safeguarded, this SOTIF weakness could lead to a safety violation in the relevant context. To ensure that a combination is sufficiently unlikely, it must be analyzed whether the triggering events are independent of each other or not. This results in an estimate of the joint probability of occurrence.
- **Triggered Weakness with erroneously inactive safety measure:** At least one safety measure must be assigned to each triggered weakness to ensure that triggered weaknesses are addressed. In practical implementation, this should be straightforward due to the naming of events. The assignment of safety measures indicates that all triggered weaknesses are safeguarded.

Not included in the modeling are accidentally active safety measures because they should not lead to a safety objective violation. If such an event is modeled in the CFT, the design shall be modified

accordingly. However, the ProbFMEA analysis shall investigate whether an inadvertently active safety measure could result in a safety violation.

The CFT analysis shows that system weaknesses such as those arising from sensor or from sensor processing can be addressed by appropriate measures. If cases with several occurring weaknesses are sufficiently improbable, the proof of the low probability of occurrence is sufficient and a safeguarding is not necessary. The disadvantage of the CFT method is that only known safety objective violations are analyzed. Whether further unintended or even safety-critical constellations arise in certain cases cannot be determined with the CFT methodology. To consider such possible effects, the ProbFMEA methodology is applied.

5.5 ProbFMEA Analysis

ProbFMEA is based on Failure Mode Effect Analysis (FMEA) and determines which failure states are possible under certain circumstances. By integrating conditional probabilities in the form of Bayesian networks, it is possible to calculate probabilities of occurrence for different events.

While in classical fault tree analysis the probabilities for events are assumed to be independent, this is insufficient for SOTIF considerations. Triggering events, such as reflections in the camera image, radar attenuation due to rain, and partial occlusions of objects can occur simultaneously. In the case of rain and reflections from light favored by puddling on the roadway, for example, there are conditional dependencies, so that a consistent quantitative model for the entire system with holistic sets of occurrence probabilities must exist for probability calculation.

The consideration of several simultaneously occurring events and their effects also leads to the analysis of which safety mechanisms are active at the same time. If several safety mechanisms are active, it must be shown that a mutual influence of the safety mechanisms does not lead to a safety target violation.

5.6 Prototype Safety Box Implementation

safeTbox allows integrated model-based Safety Engineering by providing modeling techniques for component-based architecture design, component-based fault tree creation (CFT), hazard and risk analysis (HARA) and for the specification of integrated safety concept and safety case using the Goal Structuring Notation (GSN).

Within safeTbox, CFT modeling has been adapted to support SOTIF specific basic event types. Beside supporting the SOTIF failure causes it is also possible to model their measures and again their failures. This allows to enrich the CFT with further information that supports the safety engineer with the failure analysis. Furthermore the user could also create own types for Basic events, if the given set of types were not sufficient.

safeTbox allows the user to perform a qualitative or quantitative fault tree analysis of CFTs like it's shown in Figure 21. Due to the different event types and the requirements to find for instance missing events of certain types during modeling, another functionality was implemented in safeTbox.

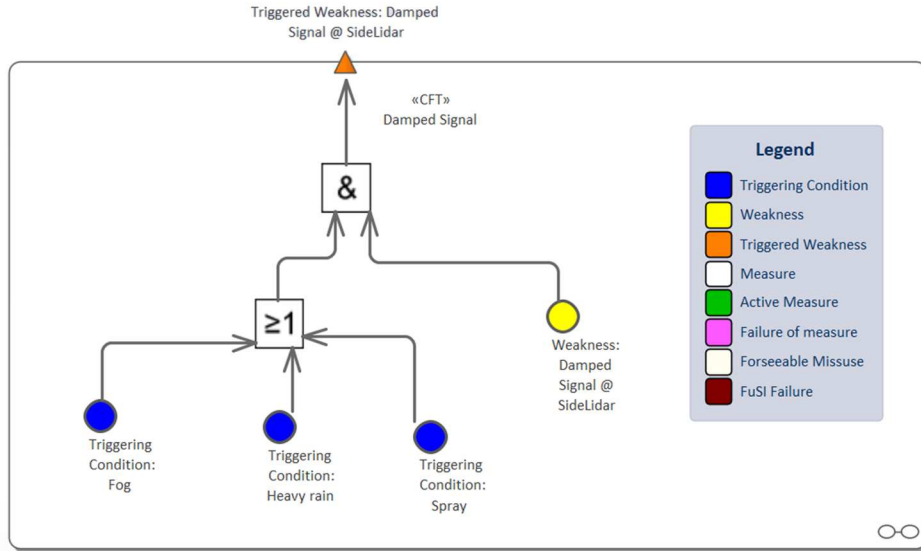


Figure 21: Exemplary modeling of a lidar sensor’s weakness and triggering conditions

The CFT analysis result can now be filtered in different ways. On the one hand this analysis post-processing allows the user to manipulate the minimal cutsets, in that the event can be included or excluded based on their type (see Figure 22). This might help for better identifying a certain type within the minimal cutsets (MCS).

Prime implicants

| # | Order | Probability | Importance | Prime implicant |
|---|-------|-------------|------------|--|
| 1 | 2 | 0.0 | 0.0 | Basic Event_32, Basic Event_44 |
| 2 | 4 | 0.0 | 0.0 | Measure_33, Measure_34, Triggering Condition_30, Weakness_31 |
| 3 | 4 | 0.0 | 0.0 | Failure of Measure_35, Failure of Measure_36, Triggering Condition_30, Weakness_31 |

Individual implicant events

| # | ID | Positive | Name | Event Type | Parent path | Failure Distribution |
|---|----|----------|----------------|------------|-----------------|----------------------|
| 1 | | True | Basic Event_32 | NONE | ComplexFilter_2 | |
| 2 | | True | Basic Event_44 | NONE | ComplexFilter_2 | |

Figure 22: Filter mechanism "Include/Exclude typed events"

On the other hand, there is another filter mechanism allowing to determine MCSs that match certain conditions defined for specific SOTIF related safety engineering use cases. For instance, MCSs

consisting of a single combination of Triggering Condition and Weakness corresponding to a classic Single-Point Fault and require special handling. Figure 23 shows how applying a filter for showing only MCSs that contain SOTIF Failures, where no measure has been considered yet, affects the list of MCSs within the analysis result.

The combination of SOTIF events and classical basic events allows the derivation of safety mechanisms that address both classical aspects of functional safety such as random hardware faults and the safety function as such.

In addition to the adjustments already described, the ODE metamodel and the DDI export functionality in safeTbox have also been adapted so that the newly introduced SOTIF failures are also supported here. A new subtype of Failure "SotifFailure" was introduced in the ODE metamodel, which can furthermore be typed according to the different SOTIF failure types. This allows to export the newly introduced SOTIF typed events within a CFT into a DDI model from within safeTbox.

Together with the developed ability to translate CFTs into Bayesian networks in an automated way, this allows for further quantitative analysis as well as the consideration of dependencies between different triggering conditions such as the mapping of the relationship of rain and a wet road (reduction of the friction coefficient) in this analysis.

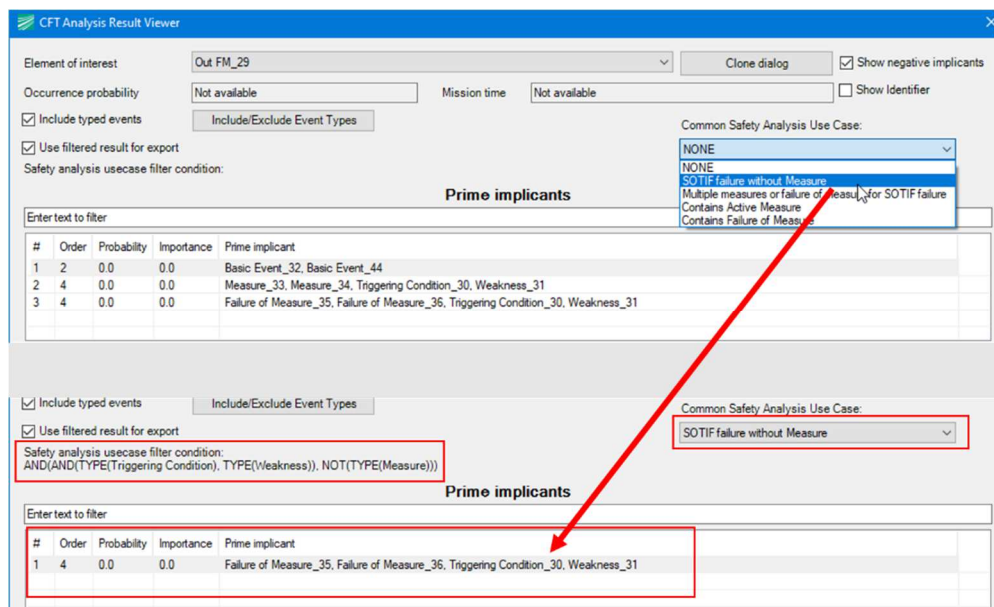


Figure 23: Use case-based filter "SOTIF Failure without Measure"

5.7 Summary safety analysis

The safety analysis identifies both the respective causes of safety objective violations and the possible effects of safety-critical events. An essential difference of the SOTIF considerations compared to conventional analysis methods is the consideration of simultaneously occurring events. Within the framework of CFT considerations, it is possible to analyze which event combinations can occur and which safety mechanisms should achieve compliance with the safety objective. Considering the respective driving situation, the probability of occurrence and thus the quantitative evaluation is possible with the help of ProbFMEA. The implementation in SafeTBox demonstrates that a tool-supported implementation is possible.

6 System requirements

6.1 General considerations

In this section, first system requirements are presented.

Requirements should be the foundation of each upcoming development project. The VVM project is about autonomous vehicles, so machines sooner or later moving safely on their own on public roads all over the world. This means that nearly each human being becomes a potential stakeholder and huge amounts of national traffic regulations have to be fulfilled by future solutions.

The VVM project focuses on verification and validation (V&V) methods for autonomous vehicles. In order to be able to verify and validate a product or service, we always have to define its desired capabilities, constraints, and behaviour first. Expressing all aspects of autonomous driving as requirements would be a tremendous challenge on its own.

But to proof V&V methods in VVM, a subset of consistent requirements should be sufficient. So, requirements are auxiliary means here and not expected to be complete by far. In the course of the project especially requirements regarding a vehicle's environment perception became of interest.

The *Handbook for the CPRE Foundation Level according to the IREB Standard*¹⁸ provides a neat introduction on system requirements:

“Whenever humans decide to build a system to support or automate human tasks, they have to figure out what to build. This means that they have to learn about the desires and needs of the persons or organizations who will use the system, benefit from it, or be impacted by it. In other words, they need to know about the requirements for that system. Requirements form the basis for any development or evolution of systems or parts thereof. Requirements always exist, even when they are not explicitly captured and documented.”

6.2 Development within VVM

Stakeholder Needs and Requirements Definition

According to ISO15288¹⁹ the “purpose of the Stakeholder Needs and Requirements Definition process is to define the stakeholder requirements for a system that can provide the capabilities needed by users and other stakeholders in a defined environment.”

In VVM we gathered needs and formal requirements from different project sources, partly in German language:

- Functional Use Case 2-3 “Occlusion of Bicyclist through Parking Cars”
- Functional Use Case 2-4 “Pedestrian Detection under the Influence of Sensor Blockage”
- Customer Function “Urbanes Fahren in V&V” (dated 2021-11-30)

¹⁸ Martin Glinz, Hans van Loenhoud, Stefan Staal, Stan Bühne *Handbook for the CPRE Foundation Level according to the IREB Standard* (IREB e.V., 2020)

¹⁹ International Standard ISO/IEC/IEEE 15288 *Systems and software engineering – System life cycle processes* (ISO/IEC/IEEE, 2015)

- Item Definition “Beschreibung der VVMethoden Fahrfunktion” (dated 2021-03-22)
- Requirements from SysML Model „VVM Gesamtarchitektur”

All Stakeholder Needs got a unique ID, a title, and a textual description, see Figure 24 for an example.

| # | Title | Requirement | Sources | Refined By |
|------|--------------------|--|------------------|--|
| SN-1 | roads | The system shall be capable of driving on roads including curves, crossings, junctions, road markings, crosswalks, parking areas, traffic signs, traffic lights, tunnels, nearby bicycle lanes. | | |
| SN-2 | t-junctions | <ul style="list-style-type: none"> • The system shall be capable of handling <i>t-junctions</i>. | FUC2-3 FUC2-4 | SR-3.1.1 t-junction perception SR-4.1.1 t-junction classification |
| SN-3 | crosswalk markings | <ul style="list-style-type: none"> • The system shall be capable of handling broad stripes on the road for <i>crosswalk markings</i>. | FUC2-3 FUC2-4 | SR-3.1.2 crosswalk marking perception SR-4.1.2 crosswalk marking classification |
| SN-4 | crosswalk signs | <ul style="list-style-type: none"> • The system shall be capable of handling traffic signs for <i>crosswalks</i>. | FUC2-3 FUC2-4 | SR-3.2.1 crosswalk sign perception SR-4.1.3 crosswalk sign classification |

Figure 24: Exemplary Stakeholder Needs

6.3 System Requirements Definition

According to ISO15288²⁰ the “purpose of the System Requirements Definition process is to transform the stakeholder, user-oriented view of desired capabilities into a technical view of a solution that meets the operational needs of the user.”

In VVM we rephrased and grouped the needs to derive a small, consistent requirements specification for an autonomous vehicle including a backend infrastructure. As stated before, the requirements specification in VVM does not claim to be complete. E.g., a driver inside the vehicle and a potential handover of the driving task between the vehicle and a driver were not addressed. Also, requirements regarding lighting, occupant protection, anti-theft protection, climatization, or entertainment were not considered at all.

Besides the stakeholder needs gathered previously we took inspirations from other requirements:

- White Paper “Safety First for Automated Driving”²¹, chapter “2.1.6.2 Overview of the Capabilities”
- VVM Subsystem Requirements Specification “Stakeholder Needs / Requirements an das Silent Testing System” (dated 2021-10-08)
- VVM Table „Definition erster grober Gütekriterien“ (dated 2020-06-29)

²⁰ International Standard ISO/IEC/IEEE 15288 *Systems and software engineering – System life cycle processes* (ISO/IEC/IEEE, 2015)

²¹ Aptiv, Audi, Baidu, BMW, Continental, Daimler, FCA US LLC, HERE, Infineon, Intel und Volkswagen *Safety First for Automated Driving* (2019)

The resulting requirements specification had the following chapters:

- System of Interest
- Order Perception
- Self-Perception
- Environment Perception
- Situation Assessment
- Option Adaptation
- Route Calculation
- Trajectory Calculation
- Trajectory Execution
- Monitoring
- Maintenance
- Central Support

| # | Title | Requirement | Req. Type | Derived From (Sources) | Refines |
|-----------|------------------------------------|--|-------------|---|---------------------------------------|
| SR-3 | Environment Perception | The system shall perceive its environment in the surroundings. | functional | SN-55 environment perception G301 | |
| SR-3.1 | road guidance perception | The system shall perceive the <i>road guidance</i> (e.g. lane boundaries, road markings, traffic lanes, crossings, parking areas and bus stops). | functional | SN-56 road guidance perception VVM_GLM-1 | |
| SR-3.1.1 | t-junction perception | <ul style="list-style-type: none"> • The system shall perceive the road guidance for <i>t-junctions</i>. | functional | SN-2 t-junctions | |
| SR-3.1.2 | crosswalk marking perception | <ul style="list-style-type: none"> • The system shall perceive broad stripes on the road for <i>crosswalks markings</i>. | functional | SN-3 crosswalk markings | |
| SR-3.1.2a | crosswalk marking perception range | <ul style="list-style-type: none"> • The system shall perceive crosswalk markings on the vehicle's traffic lane in a distance at least between ? and ? meters in the direction of driving. | performance | | SR-3.1.2 crosswalk marking perception |
| SR-3.2 | permanent static object perception | The system shall perceive permanent <i>static objects</i> at the roadside (e.g. traffic signs, traffic lights, constructional segregations, street lamps, billboards, buildings, or vegetation). | functional | SN-56 road guidance perception SN-57 structural object | |

Figure 25: Exemplary System Requirements

Up to this point the requirements specification consisted of functional requirements describing the intended behavior of the system. With help of the Goal-Question-Metric (GQM) approach we started defining quality measures. They were expressed as performance requirements refining the functional requirements.

7 Conclusion and Outlook

This deliverable shows the assurance framework as well as the detailed structuring of the V&V process and several methodologies like GQM and probFMEA/CFT operating within that framework.

It lays the ground for the next steps which will mainly consist in the derivation of the functional, logical and later the technical architecture (i.e. decomposition of system design), the derivation of test requirements based on the actual system requirements as well as the decomposition of scenarios. In later future, the validation path must be derived in a similar manner and brought together with the verification by means of the argumentation. In detail, these are the next steps based on this deliverable:

Nearer future:

- Structuring the block “Test planning”
- Further elaboration of the interception points
- Functional, logical and technical architecture
- Allocation of non-functional requirements
- Thus, finalization of the verification branch
- Derivation of (functional) test requirements

Later future:

- Implementation of the decomposition of scenarios into the test planning
- Elaboration of validation branch
- Elucidate the interaction of verification and validation within the assurance framework by establishing a V & V concept in combination with an argumentation structure,
- Elaborate an overall example highlighting the overall process,
- Establishing an overall GSN graph with the basic findings implemented,
- Implementation of all results into an interactive framework (e.g. cameo) along with the final description of the homologation process, both following the goal to make the procedure available and applicable to industry.