VERIFICATION
VALIDATION
METHODS

VVM Deliverable 14

Status of instantiation - Consistent documentation of the overall method

Version 1.0

Editor Christian Bühler

Project coordination Robert Bosch GmbH and BMW AG

Due Date 19.12.2023

Creation date 12.12.2023

Publication 19.12.2023

Acknowledgement: The research leading to these results is funded by the German Federal Ministry for Economic Affairs and Climate Action" within the project "Verifikations- und Validierungsmethoden automatisierter Fahrzeuge im urbanen Umfeld". The authors would like to thank the consortium for the successful cooperation.

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

aufgrund eines Beschlusses des Deutschen Bundestages

PEGASUS FAMILY

Document information

Authors

Christian Bühler - PROSTEP AG

Reviewer

Martin Holland - PROSTEP AG

Contact us

Christian Bühler (Responsible for Sub-Project 9)
PROSTEP AG
Dolivostraße 11
64293 Darmstadt
Deutschland

Phone: +49 89 350 20 212
Email: christian.buehler@prostep.com

Revision log

Version	date	Comment	Author	Partner
0.1	12.12.2023	final for review	Buehler	PROSTEP AG
0.2	18.12.2023	review	Holland	PROSTEP AG
1.0	19.12.2023	final	Buehler	PROSTEP AG

Table of Content

List of figures

1 Introduction

A trend towards the digitalisation and virtualisation of product development processes can be observed in industry today. The development of technical systems affects many different domains and business areas. This applies in particular to the development of automated driving vehicles, but also to most other complex technical developments. Knowledge of all artefacts and their interrelationships and dependencies, especially across domain boundaries, is crucial for a consistent development process and supports the handling of complexity. To ensure consistency, one of the most important approaches in the VVMethod project is the use of assurance argumentation to prove conformity and at the same time as a structuring approach to managing complexity.

In recent years, highly automated driving has been part of a large number of research initiatives worldwide, including the VVMethods project. At a national level, these include the Pegasus project and the sister project SET Level, which is closely linked to VVMethods.

The expectations placed on highly automated driving range, for example, from an increase in traffic efficiency to an increase in road safety and a reduction in emissions [1].

As mentioned at the beginning, highly automated road vehicles are complex systems. In addition to the challenges described above, these vehicles also have to be operated in a highly complex environment, for example in urban traffic. There is a general trend towards simulations being used more and more in the development and validation of systems. In addition to system design, virtual prototypes and simulation models are increasingly being used to validate safety-relevant systems or system aspects [2].

In the case of highly automated driving, pure validation, e.g. through real test drives or tests, does not appear to be feasible. Virtual validation certainly offers the advantage that certain environmental effects, environmental conditions or malfunctions can be analysed more safely and efficiently than is the case with real test vehicles. In addition, virtual validation may also be absolutely necessary, for example, if the test kilometres required for validation cannot be recorded with real prototypes at a reasonable cost [3]. There may also be legal constraints that make it necessary to validate a system virtually before it can be used in a real application, as certain faults or constraints cannot be carried out safely or efficiently with real prototypes.

To create test and validation procedures, it is not only necessary to make them technically feasible. An elementary aspect within the verification and validation of highly automated vehicles is also to make the underlying procedure or method comprehensible and consistent and to ensure the trust of third parties in the method.

The VVMethods project has developed an overall method that starts early in the development process with the definition of requirements and safety concepts in order to be able to take key safety aspects of the highly automated vehicle into account at an early design stage. In addition, aspects such as risk assessment, scenario creation, customer functions and the manufacturing company are also included. This makes it possible to carry out a comprehensible and credible safety argumentation based on the individual method modules. Figure 1 shows the main elements of the initial presentation of the method consolidation of the VVM method.

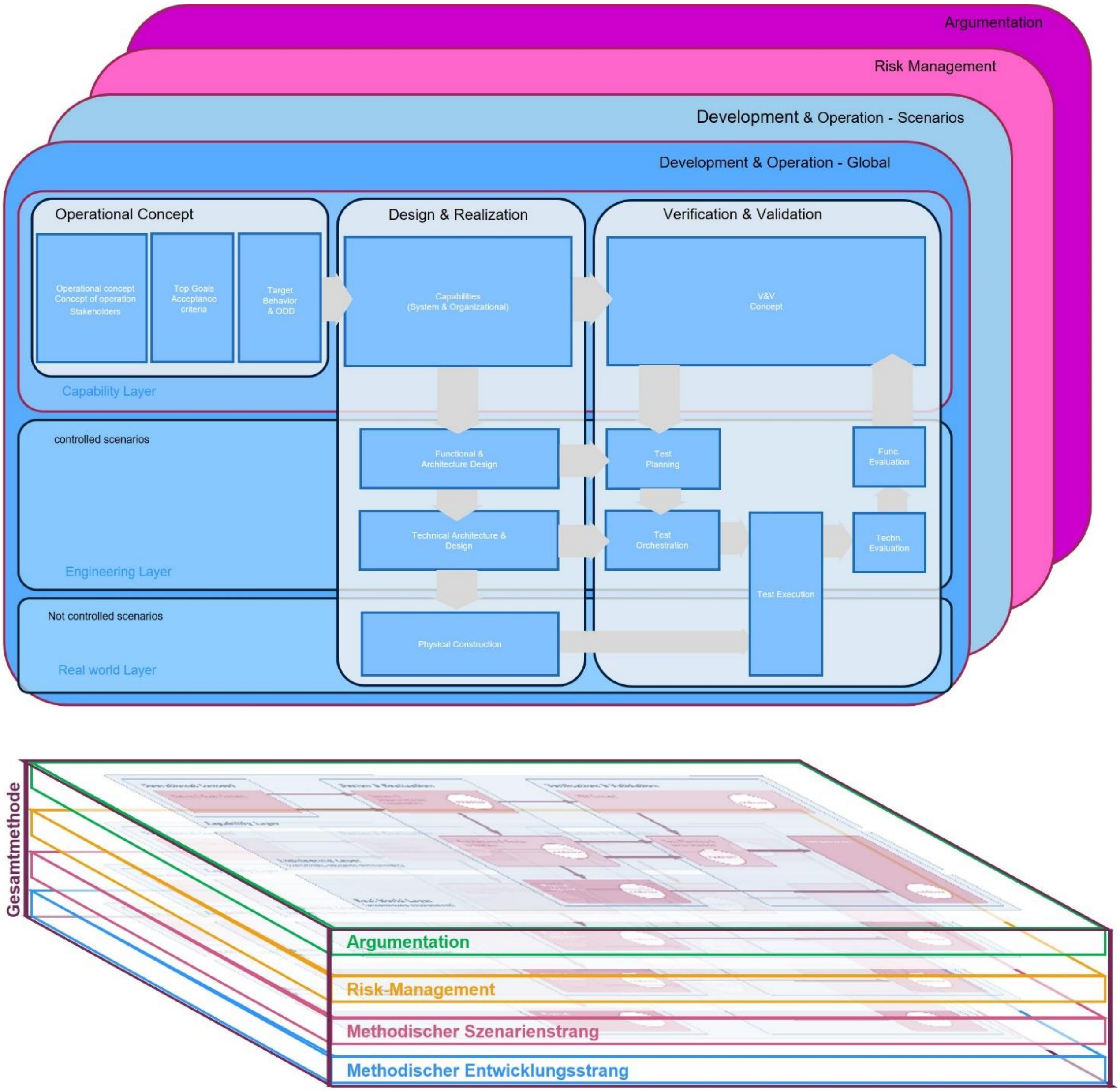


Figure 1: VVM overall method - method merging initial representation

As part of Deliverable 14, the method modules created in the VVM project were brought together as a whole in a SysML model. The overall method was also implemented in the PROSTEP traceability demonstrator developed in the project. One of the goals was to provide support for the subsequent introduction or adaptation of the processes and methods developed in the project in the companies.

Therefore, a large number of workshops were moderated in order to ensure, together with the research partners and the industrial partners, the basis for the smoothest possible introduction of the VVM results in the companies and to ensure application into the industrial application. The documentation of the method is designed in such a way that the project partners can decide independently whether they want to utilise the complete processes and methods or, if necessary, only implement partial aspects.

The Deliverable D14 described in this document therefore consists of two main fields. Firstly, the description of the model of the overall VVM method created in SysML (Chapter 2) and on the other hand the description of the implementation of the contents of the overall method in TRACY, PROSTEP's traceability demonstrator developed in the project (Chapter 3). The two implementations and thus Deliverable 14 contribute to the industrial instantiation of the project results and subsequent dissemination beyond the project.

Below are some of the main potential uses and benefits of applying modelling and implementation:

- Global structure according to the four main **"building blocks"** of the **VVM-Method**
- Graphical representation supports easy access to the **VVM-Method**
- The SysML model supports the general understanding of the **interconnections** between process steps (activities) as well as the **interchanged artefacts** between them.
- The SysML model supports the **consistency** of the VVM method and the **interfaces** between the process steps.
- The formalised description of the VVM-Method enhances **discussions** and **coordination.**
- Easy to use **HTML export** of the SysML-Model supports navigating through the whole VVM-Method
- Current implementation of the **VVM-Method** in TRACY can be used as a **"blueprint**" for industrial application.

2 The overall method in the SysML model

2.1 Preliminary work, rationale, and general procedure

Deliverable 14 aims to support the industrial instantiation of project results. The stakeholder-orientated procedure for baselining in the engineering process developed in the VVM project was used for this purpose. Using the initial Confluence-based approach, in which the project results are collected based on the synchronisation points of the VVM method, numerous individual discussions and workshops were held with those responsible for the synchronisation points.

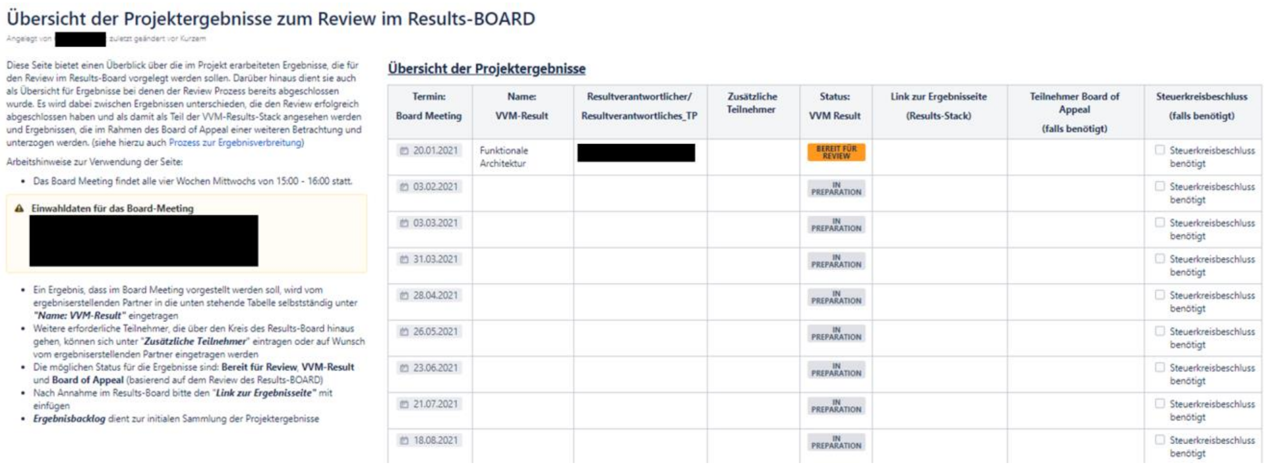
The interfaces identified between the synchronisation points and their comparison enable a consistent description of the method, which improves or enables industrial applicability and instantiation. The aim of the procedure agreed in the project is to ensure the consistency of the synchronisation points of the assurance framework and the linking of the scenarios with it. In addition, the connection to argumentation and risk management was also integrated during the process.

The creation of an integrated and consistent description of the VVM method also supports subsequent instantiation in the organisations. In this way, method modules can be clearly localised in the method and their interrelationships with other method modules become apparent. This makes it possible, for example, to apply individual contents of the method in the houses and to integrate them into an existing procedure, as the interfaces of the modules or synchronisation points are clearly described.

This procedure is additionally supported by the process for internal project baselining and internal project results approval and documentation, which was developed early in the project together with sub-project 1 and sub-project 9. The implementation of the process within the framework of BOARD meetings was carried out cyclically by the EICT and the project partners during the project period until the end of the project.

The process is divided into several stages. In the first step, project results are subjected to a review in an extended sub-project management meeting. The aim of the review is to critically reflect on the project results and to make them available to the overall project as a VVM project result once the review has been successfully completed. This also creates a basis for the initial modelling, as the results stack can be used here, as well as for the discussions and workshops with the respective stakeholders.

The procedure was agreed with all sub-project managers and a page was created to document and manage the process in Confluence for use in the overall project. A section of this page is shown in Figure 2 can be seen.



Übersicht der Projektergebnisse zum Review im Results-BOARD

Angelegt von [redacted] zuletzt geändert vor Kurzem

Diese Seite bietet einen Überblick über die im Projekt erarbeiteten Ergebnisse, die für den Review im Results-Board vorgelegt werden sollen. Darüber hinaus dient sie auch als Übersicht für Ergebnisse bei denen der Review Prozess bereits abgeschlossen wurde. Es wird dabei zwischen Ergebnissen unterschieden, die den Review erfolgreich abgeschlossen haben und als damit als Teil der VVM-Results-Stack angesehen werden und Ergebnissen, die im Rahmen des Board of Appeal einer weiteren Betrachtung und unterzogen werden. (siehe hierzu auch Prozess zur Ergebnisverbreitung)

Arbeitshinweise zur Verwendung der Seite:

- Das Board Meeting findet alle vier Wochen Mittwochs von 15:00 - 16:00 statt.

Einwahldaten für das Board-Meeting

[redacted]

- Ein Ergebnis, dass im Board Meeting vorgestellt werden soll, wird vom ergebniserstellenden Partner in die unten stehende Tabelle selbstständig unter ***"Name: VVM-Result"*** eingetragen
- Weitere erforderliche Teilnehmer, die über den Kreis des Results-Board hinaus gehen, können sich unter "***Zusätzliche Teilnehmer***" eintragen oder auf Wunsch vom ergebniserstellenden Partner eingetragen werden
- Die möglichen Status für die Ergebnisse sind: **Bereit für Review**, **VVM-Result** und **Board of Appeal** (basierend auf dem Review des Results-BOARD)
- Nach Annahme im Results-Board bitte den "***Link zur Ergebnisseite***" mit einfügen
- ***Ergebnisbacklog*** dient zur initialen Sammlung der Projektergebnisse

Übersicht der Projektergebnisse

Termin: Board Meeting	Name: VVM-Result	Resultverantwortlicher/ Resultverantwortliches_TP	Zusätzliche Teilnehmer	Status: VVM Result	Link zur Ergebnisseite (Results-Stack)	Teilnehmer Board of Appeal (falls benötigt)	Steuerkreisbeschluss (falls benötigt)
20.01.2021	Funktionale Architektur	[redacted]		BEREIT FÜR REVIEW			Steuerkreisbeschluss benötigt
03.02.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
03.03.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
31.03.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
28.04.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
26.05.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
23.06.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
21.07.2021				IN PREPARATION			Steuerkreisbeschluss benötigt
18.08.2021				IN PREPARATION			Steuerkreisbeschluss benötigt

Figure 2: General structure of the Confluence page for project-internal release of results and documentation (personal content blacked out)

As described above, the aim of the procedure described here is the consistent description of the VVM methodology so that it can also be used industrially. For the industrial applicability of the results, it is essential that the described findings are not only consistent, but that their representation can also be easily accessed. With this in mind, it was decided together with the project management and the sub-project leaders to carry out the documentation in a SysML model, among other things. In addition to its native format, the tool used for modelling also offers an HTML version that can be opened with a current Internet browser and navigated throughout the entire model. This also offers a low barrier to entry outside the project, which means that dissemination activities can also be supported. As the Cameo Systems Modeller is also already being used by some project partners, the native format is also made available in addition to the HTML export to enable the project partners to continue working on the content independently after the end of the project and to integrate it into any SysML models that may already exist in the organisations. This is also intended to support the industrial instantiation and dissemination of the VVM method results.

2.2 Realisation in the SysML model

For the concrete, model- or SysML-based implementation of "Baselining in the Engineering Process", the synchronisation points of the VVM method were used as a guide in coordination with the project management at the beginning of the work. The aim is to identify and harmonise their interfaces and transfer artefacts and thus ensure consistency across the synchronisation points within the assurance framework (referred to as "Development & Operation - Global" in the further course of the project and finally in the project).

To this end, the persons responsible for the synchronisation points were identified and, together with a project partner, a document structure was set up in the project and afterwards revised by sub-project 9, which makes it possible to describe the activities of each synchronisation point and the resulting inputs and outputs in text form. Figure 3 shows the general structure of the document used for this procedure for the synchronisation points. The documents generated in this way support a uniform understanding of the granularity of the artefacts to be described. In addition to the results already obtained and made available from other Deliverables and from the results board, intensive discussions were held with those responsible for the synchronisation points as well as across synchronisation points based on this content. The aim was on the one hand to further detail the internal structure of the synchronisation points and on the other hand to harmonise the interfaces between the synchronisation points. This procedure was also applied in an adapted version to the other method modules, Development & Operation - Scenarios, Argumentation and Risk Management.

3 Synchronization Points of the VVM Method

3.1 Operational Concept

Main Author: NN

- **Goals** – a general description of what the constituent shall achieve
- **Inputs and Outputs**: Both a definition of the information going in and out, and the format in which this information is represented.
- **Activities, Methods, and Tools**
  - In- and Outputs of Subactivities of the Synchronization Points
  - Methods and Tools (Description of Methods and Tools; adequacy, consistency, and completeness of results)
  - Exemplification based on for FUC 2.3.

3.2 Criticality Analysis

Main Author: NN

- **Goals** – a general description of what the constituent shall achieve
- **Inputs and Outputs**: Both a definition of the information going in and out, and the format in which this information is represented.
- **Activities, Methods, and Tools**
  - In- and Outputs of Subactivities of the Synchronization Points
  - Methods and Tools (Description of Methods and Tools; adequacy, consistency, and completeness of results)
  - Exemplification based on for FUC 2.3.

Figure 3: General structure of the document on the synchronisation points of the assurance framework (called "Development & Operation - Global" in the further course of the project and finally in the project)

In addition to the synchronisation points of the assurance framework (referred to below only as "Development & Operation - Global"), the overall VVM method consists of other components. Figure 4 shows the final diagram presented as a landing page in the SysML model. In the "Development & Operation - Global" tab (top left), the former synchronisation points are shown as light pink boxes. The diagram also contains the other method modules, Development & Operation - Scenarios, Argumentation and Risk Management in schematic form.

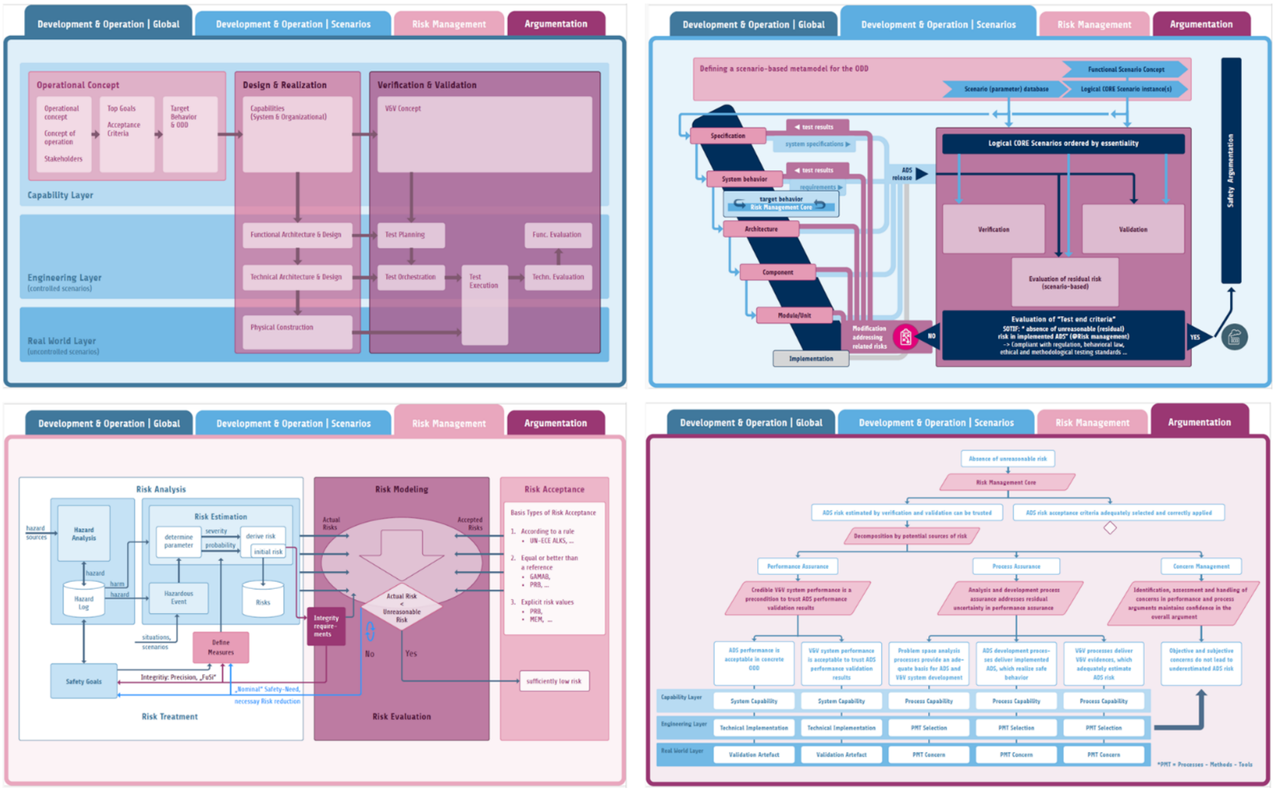


Figure 4: VVM overall method - method merging (landing page - SysML model)

The elements shown in Figure 4, which contribute to the overall structure of the method, were also adopted in this form in the SysML model as a landing page for the HTML documentation. The presentation of the main elements as an entry point in the form of a landing page was primarily chosen in this form for a later publication of the model. It enables a low-threshold introduction to the method via the model, as agreed with the management and the project, as the same elements were also used on the associated reference pages, posters and in the presentations at the final event, thus guaranteeing a clear recognition value.

For the integration of the deeper, more detailed levels of the overall method, the main elements of the overall VVM method were not only graphically represented, as shown in Figure 4 were included in the model. They were also integrated into the SysML model as a top-level layer. The formal representation of the main elements of the VVM overall method with their interrelationships is shown in Figure 5. On the one hand, this representation serves to make it easier to enter into discussions and coordination, as the global interfaces between the main elements of the method can be discussed directly on the model. This means that all newly gained insights are documented directly and centrally in one place and are also visible to the representatives of the other main elements. Secondly, the formal notation makes it easier to ensure the consistency of the interfaces, as the transfer and exchange objects, as well as the direction in which they are transferred, are directly visible in this representation.

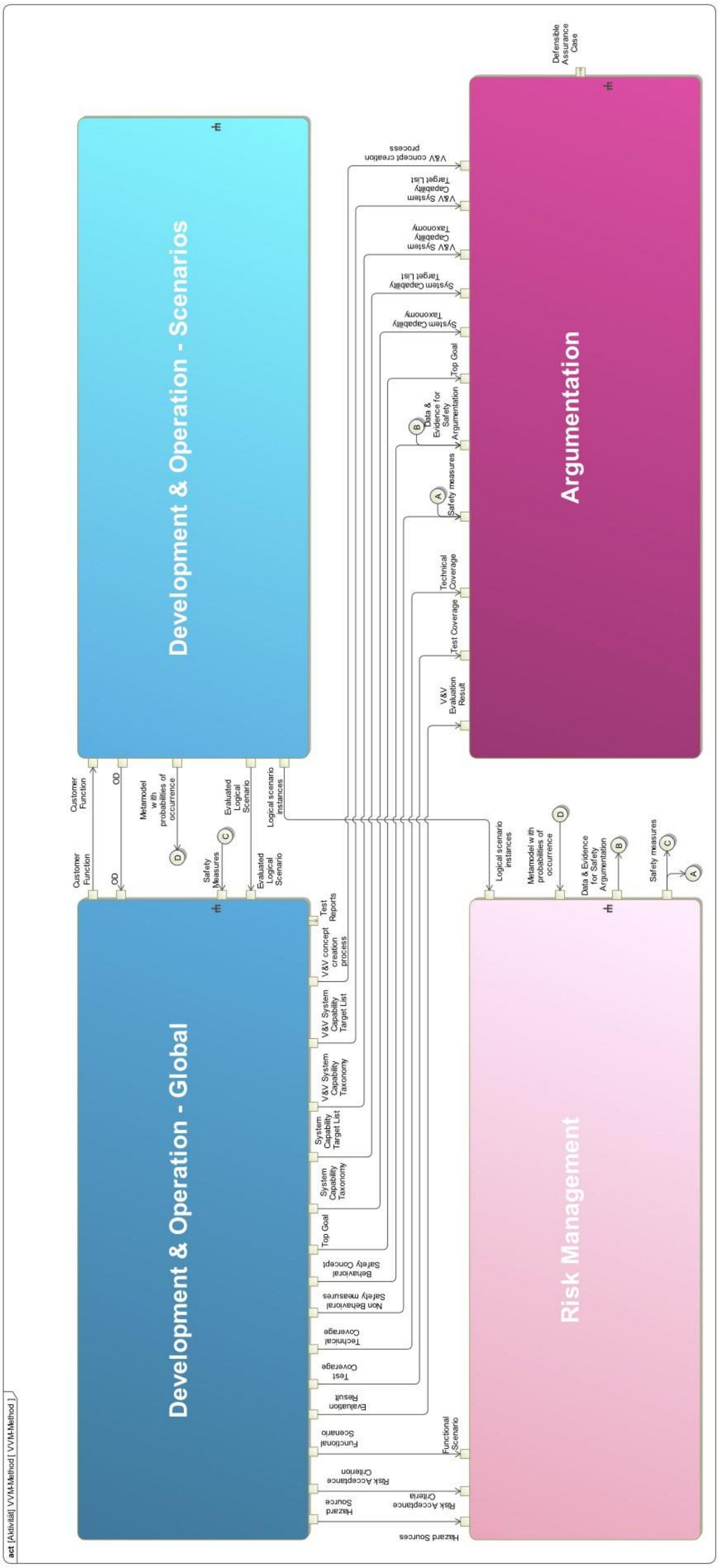


Figure 5: Modelling of the main elements of the overall VVM method with their interrelationships

A comprehensive SysML model of the overall method was created based on the available documents, project content and in consultation with those responsible for the respective method elements. More detailed descriptions were created in the model for the respective main elements of the VVM method. These are also described by activities with their inputs and outputs. This means that the interaction within the main elements of the VVM method can also be consistently described and documented.

This procedure offers the possibility of directly documenting the interfaces and transfer artefacts consistently and identifying any inconsistencies. Based on the ever-growing model created in this way, discussions regarding the consistency of the entire method were held cyclically with those responsible. Encapsulation within the method modules in the form of individual interlinked activities also made it possible to describe individual sub-areas at different levels of detail, as the global interfaces between the activities were always clearly described at the higher level and therefore the level of detail within them did not affect them.

This approach was used on several levels, especially but not exclusively in the main element "Development & Operations - Global". This also guarantees that the modelled content is easy to understand and clear, as only a manageable amount of information is described in a diagram and detailed information can be found one level lower. This top-down approach also makes it possible, if desired, to consider only individual method components in detail and still capture the overall context within the method. The more abstract descriptions are also clearly and consistently linked to the detailed activities due to the consistent interfaces.

Figure 6 shows an example of the relationship between different model- /method levels. The main element "Development & Operations - Global" was selected at the top. One level lower, you will find the "layers" structuring this main element, Capability, Engineering and Real World Layer (in shades of blue). Here, too, the relationships within the main element of the method have been described in a relatively abstract manner for the sake of clarity and understanding. However, it can also be seen here that the inputs and outputs of the higher level appear in full in this representation and are used by method elements. The concrete core activities within the main element are described one level lower. These were also described as synchronisation points within the project and in some cases are directly related to individual sub-projects. Here, too, the "layers" are completely shown for better categorisation. The last layer shown here describes the structure of the “Criticality Analysis”. The left-hand side of the diagram shows the transfer objects that are supplied at the interfaces of the criticality analysis by other method modules for their internal use in the “Criticality Analysis”. The objects that are provided after processing are shown on the right-hand side of the diagram and are made available to other method modules via this interface. The diagram itself shows the process within the "Criticality Analysis". The diagram therefore shows how the individual activities follow on from each other and how objects run through the activities and are processed.

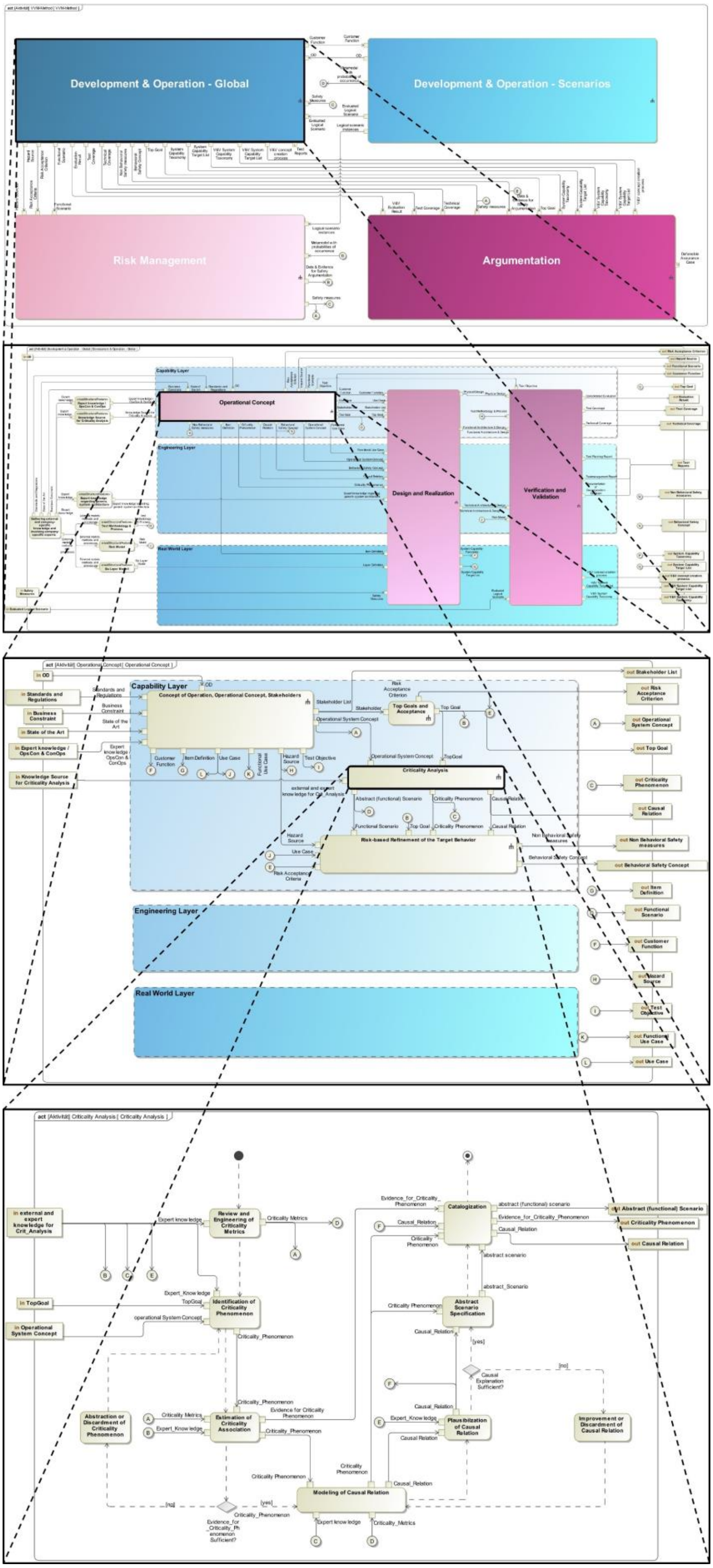


Figure 6: Illustration of the relationship between different model/method levels

The structure of the model shown in Figure 6 primarily shows the individual activities and processes and their sequence, the interrelationships and the exchange or transfer objects between the process steps.

During the modelling and creation of the model, it became apparent that this provides a very good overview of the individual components of the VVM method, but that certain dependencies and relationships are not fully captured in this way. For this reason, in addition to the pure activity diagrams, other diagram types were included in the model, with the help of which it is possible to depict the other interrelationships.

As a simple extension to the original approach, diagrams are included that contain requirements for certain process elements or transfer objects or are themselves transfer objects. In addition, diagrams have been introduced that can directly describe the direct relationships and the structure of certain transfer objects. This was mainly done for objects that themselves consist of various individual objects in order to be able to transfer them as a superordinate object from one activity to the next without having to transfer all individual parts. This procedure was also used when objects have more complex relationships with other objects. In both cases, this increases the clarity and comprehensibility of the models. Figure 7 (below) is an example of such a diagram.

The "Technical Test Specification" object is described centrally in this diagram. You can see that the "Technical Test Specification" consists of several objects. These include "Test Case", "Initial Assessment of Success of Test Sequence", "Test Infrastructure" and "Test Sequence Specification", as well as others as "References" and objects. This allows you to see directly which content is described in the "Technical Test Specification". Furthermore, if you follow the dependencies upwards, you can see, for example, that the "Test Sequence Specification" consists of a "Test Procedure", which in turn contains a "Criterion for Test End or Abortion", which is derived from the "Required Quality and Confidence" and the associated "Confidence Requirement".

With increasing model complexity, it seemed sensible not only to model such relationships, but also to document them in text form in the model in selected cases in order to keep the entry hurdle low, especially for central objects, or to have a reference work in addition to the modelling. One such example is shown in Figure 7 (above). In the textual description of the "Functional Test Specification", links (in blue) to individual other objects are also provided to enable simple navigation in the model.

As the modelling progressed, individual parts of the model were also established by some working groups as a concrete working medium for documentation and discussion. The encapsulation within the method modules meant that the results could be used directly, and the models developed could be integrated directly into the associated elements without having to adapt or revise the general procedure, even if the level of detail was sometimes significantly higher than was the case in other modules. The consistency of the model was also ensured by the global interfaces.

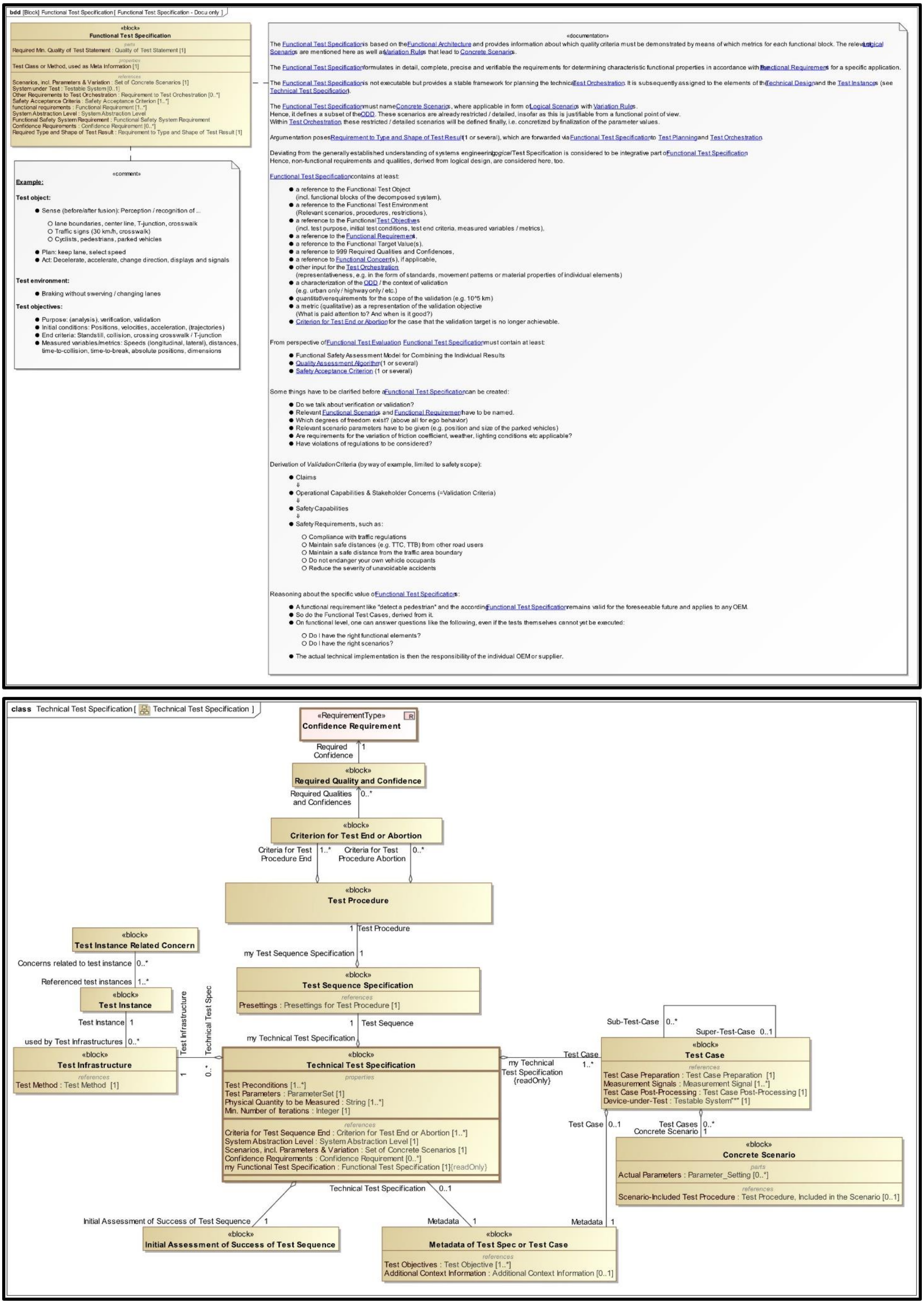


Figure 7: Example of further connections and documentation in the model

2.3 Documentation for dissemination and further use after the end of the project

For later dissemination, in preparation for use at the final event of VVMethods and for further possible use by project partners, it was decided together with the project management and the sub-project leaders, as described above, that the model would be made available as an HTML export.

In addition, this form of publication also offers a certain protection mechanism. As the user only has read-only rights, no changes can be made to the model if it is made available to people outside of VVMethods, for example at the final event or in exchange with external partners on the VVMethods homepage (https://cameo.vvm-projekt.de/). The proprietary format with all modification options is also made available within the project.

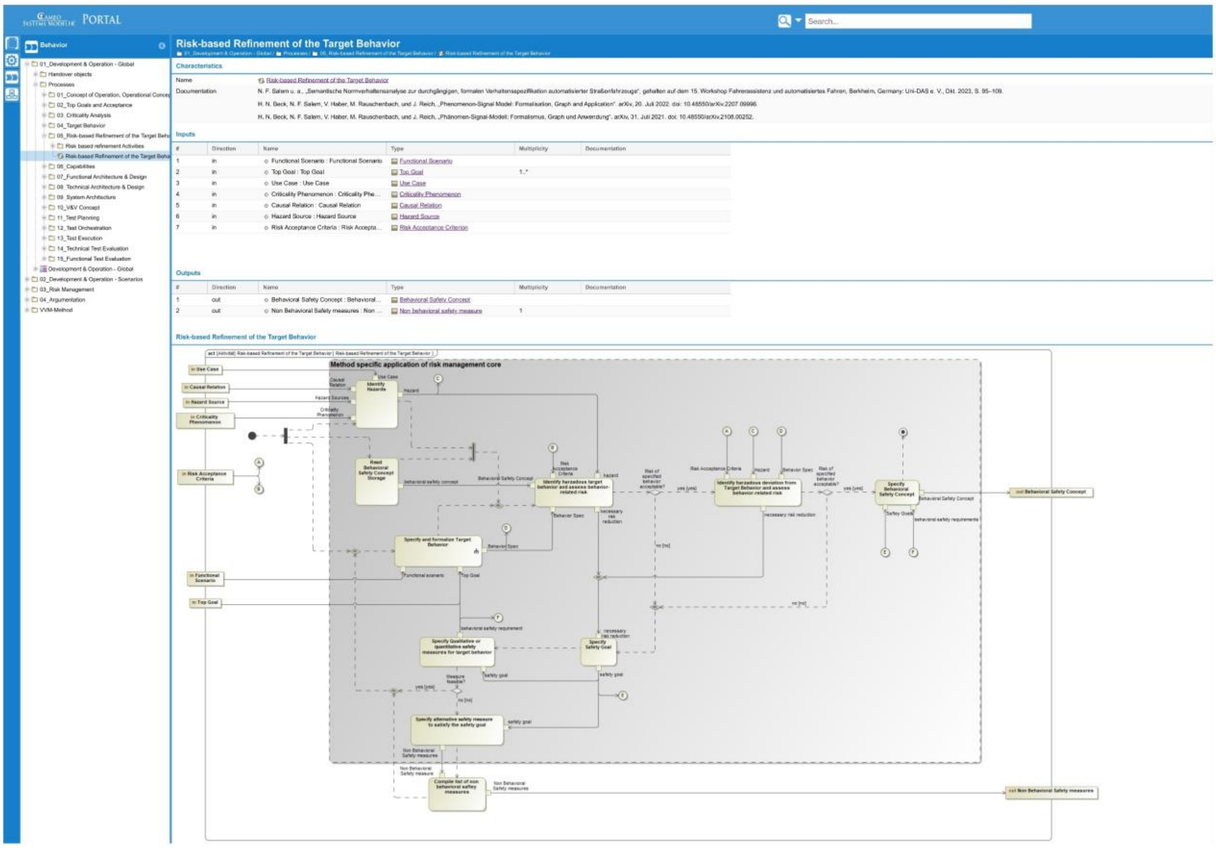
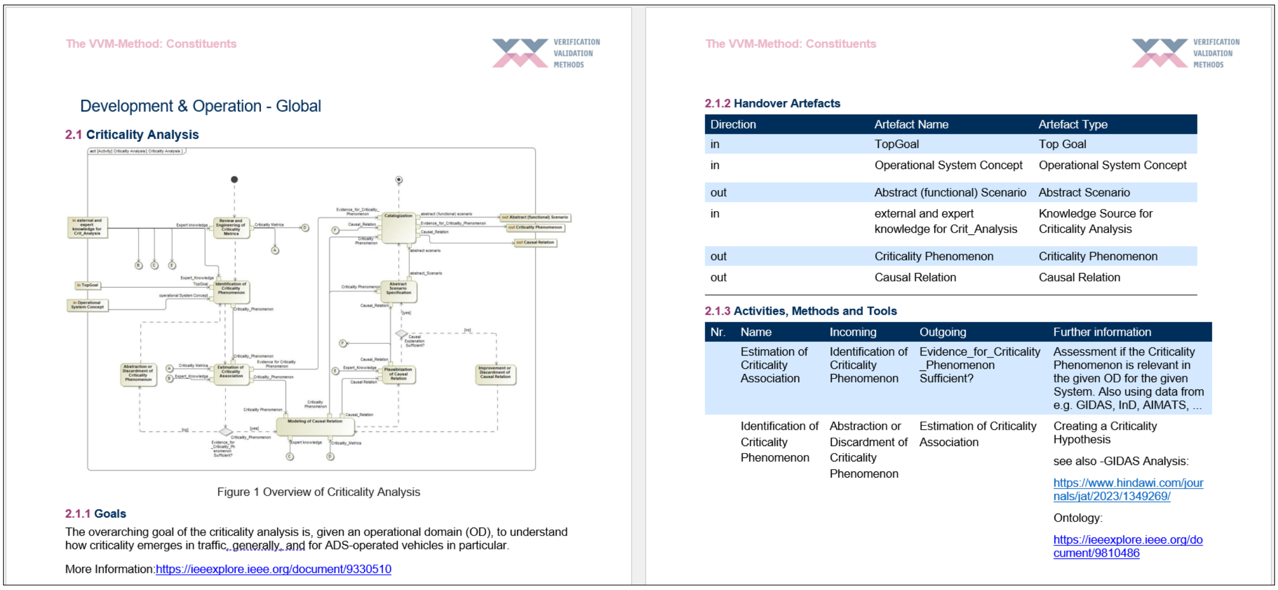


Figure 8: HTML documentation of the SysML model, view in the browser ("Risk-based Refinement of the Target Behaviour")

Figure 8 shows the documentation in HTML view. The graphical model of the "Risk-based Refinement of the Target Behaviour" can be seen in the lower part. The navigation area, in which the entire VVM method can be navigated, is located on the left-hand side. The inputs and outputs of the diagram are shown in the centre of the illustration. Further content can be displayed in the upper section, e.g. in this case further information in the form of links to publications. The individual elements of the diagram as well as the inputs and outputs are "clickable" so that they can be used to navigate through the entire method.

The project also discussed the form in which the content could be documented in writing, for example for Deliverables or reports. Figure 9 shows a possible documentation of such documentation directly from the SysML model, which can be stored as an unalterable PDF.

The diagram created in the model is also included in this documentation (see Figure 9 left). In addition, the individual activities of the method elements are again documented here, and their inputs and outputs are listed in tabular form. In addition, the documentation contains the explanatory content and further references, where included in the model. The documentation thus contains all the essential contents of the model in written form and can be regarded as a kind of "instruction leaflet" for the model.


The VVM-Method: Constituents

Development & Operation - Global

2.1 Criticality Analysis

Figure 1 Overview of Criticality Analysis

2.1.1 Goals

The overarching goal of the criticality analysis is, given an operational domain (OD), to understand how criticality emerges in traffic, generally, and for ADS-operated vehicles in particular.

More Information: https://ieeexplore.ieee.org/document/9330510

The VVM-Method: Constituents

2.1.2 Handover Artefacts

Direction	Artefact Name	Artefact Type
in	TopGoal	Top Goal
in	Operational System Concept	Operational System Concept
out	Abstract (functional) Scenario	Abstract Scenario
in	external and expert knowledge for Crit_Analysis	Knowledge Source for Criticality Analysis
out	Criticality Phenomenon	Criticality Phenomenon
out	Causal Relation	Causal Relation

2.1.3 Activities, Methods and Tools

Nr.	Name	Incoming	Outgoing	Further information
	Estimation of Criticality Association	Identification of Criticality Phenomenon	Evidence_for_Criticality_Phenomenon Sufficient?	Assessment if the Criticality Phenomenon is relevant in the given OD for the given System. Also using data from e.g. GIDAS, InD, AIMATS, ...
	Identification of Criticality Phenomenon	Abstraction or Discardment of Criticality Phenomenon	Estimation of Criticality Association	Creating a Criticality Hypothesis see also -GIDAS Analysis: https://www.hindawi.com/journals/jat/2023/1349269/ Ontology: https://ieeexplore.ieee.org/document/9810486

Figure 9: Representation of the contents of the SysML model in written form, e.g. as a Word document or PDF (shown here: “Criticality Analysis”)

3 Representation of the overall method in the Traceability Demonstrator

In addition to documenting the overall method in the SysML model, the method was documented in the PROSTEP's traceability demonstrator TRACY developed in the project. The result was also used as a showcase for the final event, similar to the VVM half-time event.

TRACY will be used to make the overall VVM method accessible to a wide audience at the final event in an alternative presentation. Thanks to its graphical user interface, it offers an easy introduction to the content, which is less formalised than the SysML model for the overall method.

In addition, both representations of the overall VVM method were developed in such a way that the results of the procedure can be made available to the project and in particular to the VVM industrial partners after the end of the project. For the industrial applicability of the results, it is essential that the findings described are not only consistent, but that their representation can also be easily accessed. With this in mind, the implementation in the Traceability Demonstrator offers the option of exporting to XML, as well as a time-limited use of the demonstrator by the partners after the end of the project.

In addition, for later concrete application in the companies the Traceability Demonstrator offers the possibility of mapping different revision statuses in order to make the progress of the development traceable, in addition to the pure representation of the process of the overall VVM method and its development artefacts. This corresponds to the idea of configuration management in accordance with ISO 10007 [4]. It is also possible to link specific documents and development artefacts in TRACY from various external systems in order to support the concrete development along the method and to illustrate the existing dependencies in a graphically comprehensible manner.

In the following, we will briefly discuss the topic of configuration management and traceability and present the core functionalities of the traceability demonstrators.

3.1 Traceability in the context of configuration management

Traceability is the context-specific linking of artefacts (documents, structures, substructures, etc. and associated metadata) via trace links. All trace links and the associated artefacts form edges and nodes of a graph that can be used for efficient analyses, for example.

Traceability is particularly relevant in the development of safety-critical systems and for mandatory audits, e.g. in accordance with ISO 26262, Automotive SPICE and EN 50128. A common requirement of these guidelines is that critical requirements must be verified, and that this verification must be made transparent through traceability. Today, the corresponding processes are document-driven - which is time-consuming, error-prone, etc.

The advantages of implementing traceability are, for example, (system-, location-, organisation-) independent links between artefacts, persistent relationships, change impact analyses, coverage analyses, project status analyses, reuse of product components, test optimisation. Tasks with traceability support are 24% faster and 50% more correct [5].

In the context of highly automated driving, the focus is on both safety-critical and complex systems that are subject to high demands in terms of quality and reliability. Traceability therefore also plays an important role here. The basic idea of traceability relates to configuration management. Figure 10 shows the individual areas of configuration management in accordance with ISO 10007 [4].

ISO 10007 "Quality management - Guidelines for configuration management" [4]:

> *Configuration management is a* ***management activity*** *that applies technical and administrative direction over the life cycle of a product and service, its configuration* ***identification*** *and* ***status****, and related product and service configuration* ***information****.*
>
> *Configuration management documents the product or service configuration. It provides identification and* ***traceability****, the* ***status of achievement*** *of its physical and functional requirements, and* ***access*** *to accurate* ***information*** *in* ***all phases of the life cycle****.*

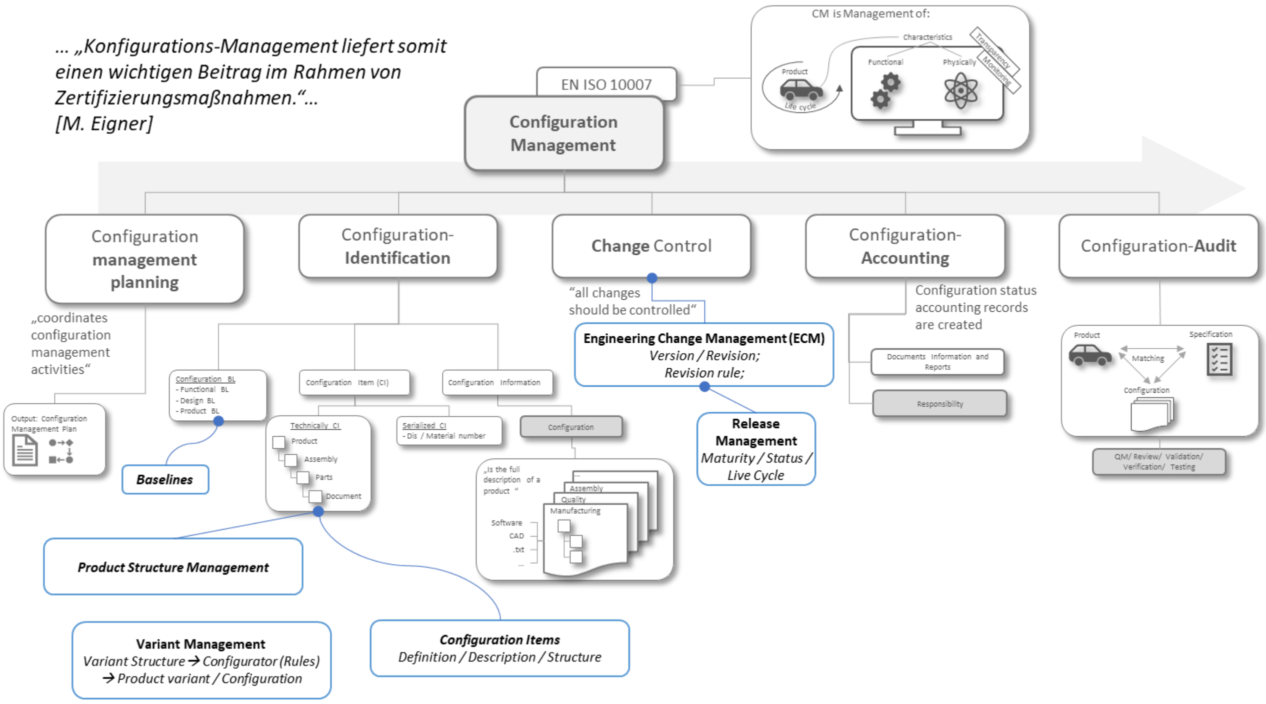


Figure 10: Configuration management areas according to ISO 10007 [4]

To prove the above-mentioned safety and quality of a highly automated vehicle, it is essential to document its exact configuration, to manage the compilation and configuration of the associated development artefacts and to be able to prove at any time how each individual artefact was created. This traceability is also an elementary basis for the credibility of the development results.

3.2 Traceability Demonstrator "TRACY"

Without the support of powerful software, it is almost impossible to keep the multitude of processes and artefacts and their interdependencies under control. For this reason, such software was developed as a demonstrator for traceability as part of VVMethods. TRACY is a neutral software demonstrator of traceability software that was developed in the publicly funded SETLevel and VVMethods projects and made available for them.

For the required documentation of the VVM method, TRACY was used as an innovative tool tailored to the specific complexity. It can enable traceability via baselines so that it can be reliably proven at any time which assumptions, models, data and parameters were used with the aid of which software. To this end, the tool links the relevant data and also enables the organised storage of information for which no system currently exists. It thus forms a technological basis for creating a baseline of all relevant artefacts. TRACY thus implements the basic principles of configuration management in accordance with ISO 10007 [4] (in particular trace links).

The traceability tool is used for interdisciplinary collaboration and can compile information from a wide variety of data sources without interfering with these data sources. The authoring systems can thus be easily adapted to necessary changes and innovations and it is possible to react flexibly to changes. Figure 11 shows the final expansion stage of the TRACY traceability demonstrator in VVMethods.

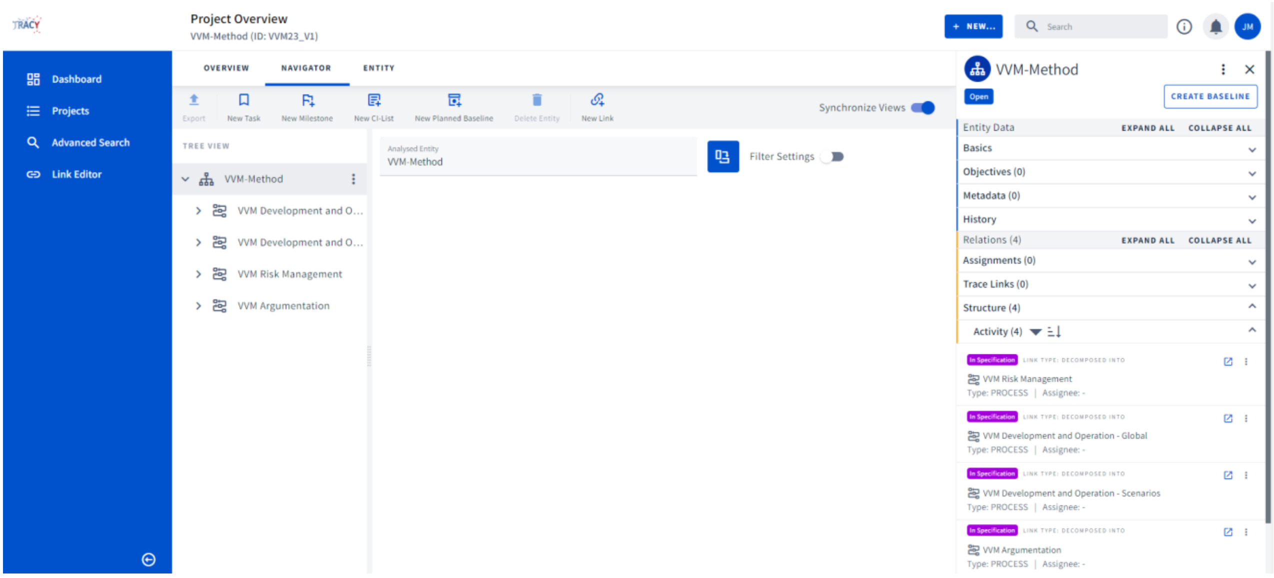


Figure 11: Final expansion stage of the traceability demonstrator in VV methods

3.2.1 TRACY - Use cases and functionalities

A traceability tool must map the central relationships and thus make the traceability between argumentation and development processes clear. The central use cases required in the project for such software are

Assure Compliance

- distil & store information
- support processes
- revision safety, auditing

Handle Complexity

- impact analysis
- document maturity
- tracking & steering
- monitor changes

Driving Efficiency

- search & reuse
- front-loading
- automation
- collaboration

Safeguard Quality

- Right information, in time etc.
- consistency
- completeness

In order to address and cover the use cases described above, the following functionalities, among others, were realised in TRACY:

- User management
- Role and rights management
- User-specific cockpits and views
- Workflow for quality assurance
- Realisation of notifications (information, e.g. when a new task has been assigned to a user)
- Create and search for artefacts (incl. metadata)
- Creation of trace links between artefacts (relations)
- Realisation of a baselining functionality
- Options for carrying out dynamic impact analyses
- Graphical user interface (GUI)
- Graphical representation of managed artefacts and their relations
- Display and filter mechanisms in the graph view
- Filling TRACY with project data
- Integration of further systems according to the OSLC standard (e.g. DOORS)
- Cloud capability of the traceability demonstrator

3.2.2 TRACY - Artefact types

Various classifications of artefact types have been implemented in TRACY to better structure the documentation of the method and the transfer artefacts. TRACY supports the following *classifications of artefacts, among others*:

- Activity
- Requirement
- Design Specification
- Test Case
- Scenario
- Model
- Simulation result
- Various importable SysML classes
- and much more.

Each of these artefacts can also be enriched with different *metadata. Furthermore,* different dependencies, the so-called trace links, can be set between the artefacts. This creates navigable triple graphs from the artefacts and the associated relationships.

3.3 Implementation in the traceability demonstrator

As with the modelling in SysML described above, the Traceability Demonstrator was also used in part as a working medium for documenting project or workshop results in order to demonstrate and evaluate its applicability for supporting "baselining in the engineering process". One of the aims was to define project-specific requirements for the Traceability Demonstrator through its specific use. This was done in order to create early acceptance of the demonstrator for use in documenting the overall method and presenting content at the mid-term event. The requirements arose both from the application and from specific consultations with project partners. Figure 12 shows an example of the implementation of the results of a project meeting in the Traceability Demonstrator.

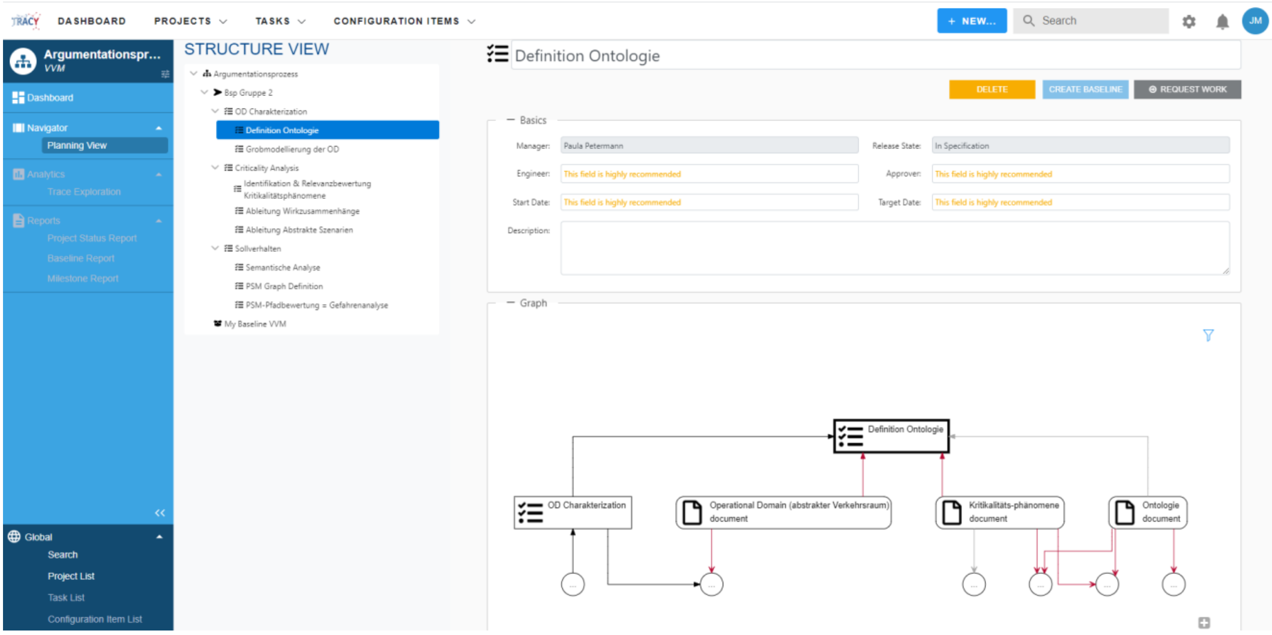


Figure 12: Exemplary documentation of the results of a project meeting in TRACY

After the initial work, the processes and artefacts within sub-project 3 were selected as the starting point for implementation in the Traceability Demonstrator. The knowledge gained here was then applied to the other elements of the method. Figure 13 shows an example of a section of the continuous argumentation chain in sub-project 3 and the connection with other sub-projects.

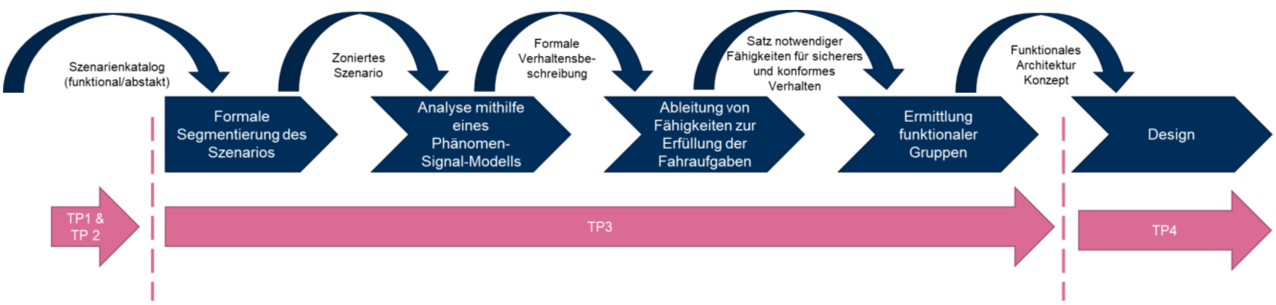


Figure 13: Extract from the continuous argumentation chain sub-project 3 (early work status in the sub-project)

For the implementation of the overall method in the traceability demonstrator, reference was also made to the results of the test described in chapter 2.1 and the project-internal release of results and documentation. In the early phase of the project, the procedure was also designed to ensure the consistency and traceability of the development artefacts along the process chain and served as a core element for the dissemination of the project results for the mid-term event.

The content provided by the project partners was broken down into various interlocking processes and sub-processes for the documentation of the method and enriched with specific development artefacts. Figure 14 shows an excerpt from the method documentation presented at the mid-term event. The overview tree on the left already shows the basic building blocks of the later "Development & Operations - Global" (see Figure 4 and Figure 6), as well as elements of the lower levels, for example "Specification of Target behaviour", "Test Orchestration" or "Test Execution".

The process flow, recognisable by the blue arrows in the diagram, has also been modelled in this illustration where available. The internal structure of the individual activities could not yet be fully and consistently described for all elements in this phase of the project. In this illustration, concrete work results of the project were primarily assigned to the respective activities, also with regard to the showcase for the mid-term event. This was done in order to classify them in the overall context of the assurance framework defined up to that point and to visualise the interrelationships. Figure 14 (left) shows, for example, that the "Formalisation of the PSM", the "PSM software" and the "Risk assessment" are parts of the "Specification of target behaviour".

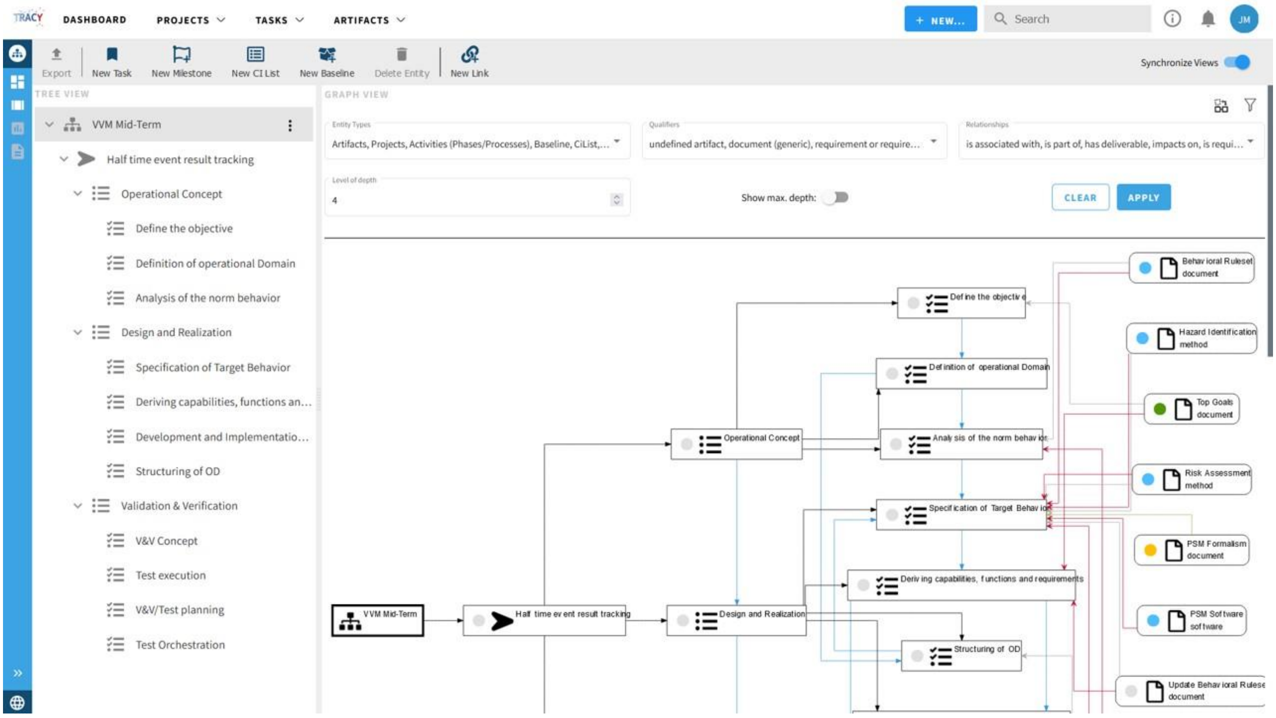


Figure 14: Extract from the implementation in TRACY presented at the half-time event

The implementation of the Traceability Demonstrator at the half-time event was given functions specially adapted to the event. On the one hand, elements of the graphical interface were revised and thus simplified in order to streamline it and make it easier to get started, as certain standard functionalities were not required for this presentation. On the other hand, a read-only functionality was implemented to prevent third parties from making changes to the content at the event.

For the final implementation in the Traceability Demonstrator, the focus was less on the individual concrete work results, as in the half-time event. As in the SysML model, the focus here was also on the representation and documentation of the consistency of the overall VVM method. For this reason, a completely new implementation of the contents of the VVM method was carried out for the final implementation.

The structure of the documentation is also based on the four main levels, "Development & Operation - Global", "Development & Operation - Scenarios", "Argumentation" and "Risk Management", of the overall method (see Figure 11 and Figure 16 top left). Within this level, a decomposition was carried out into the individual process steps belonging to the level. This results in a tree-like top-down structure that documents which process steps make up certain elements of the method and which inputs and outputs they have. Figure 15 shows an example of the internal structure of the method element "Capability Identification Process" with the associated inputs and outputs.

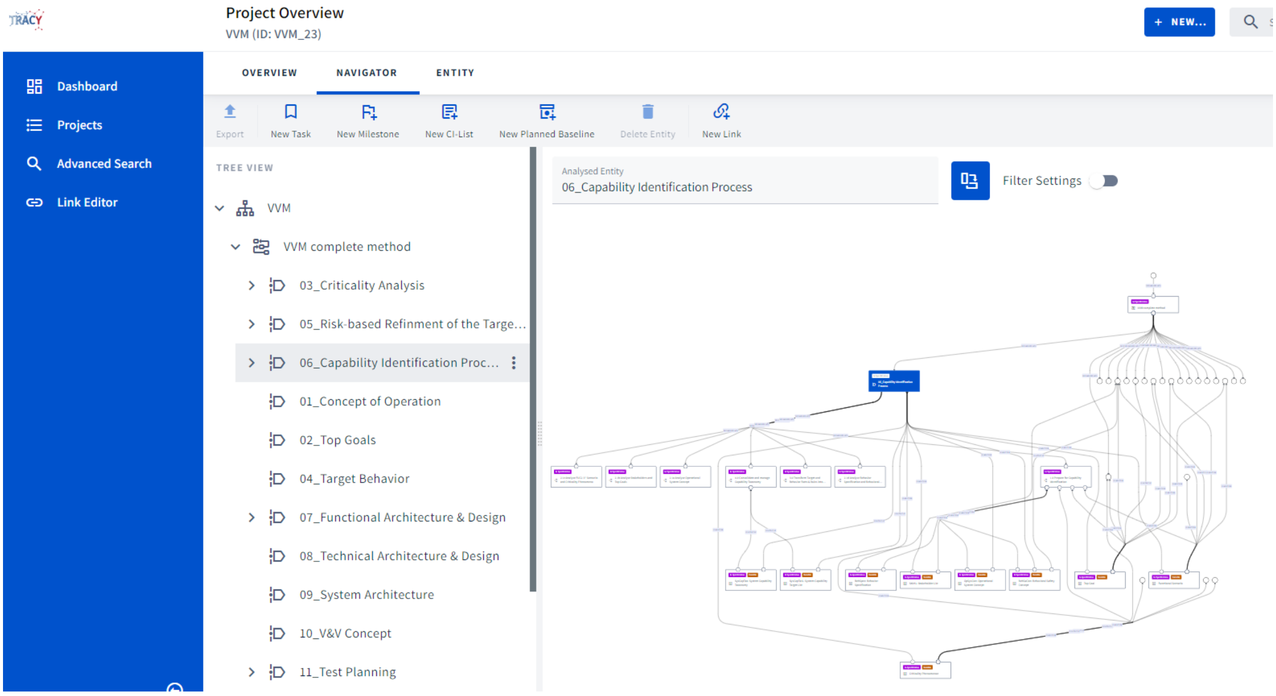


Figure 15: Structure of the method element "Capability Identification Process" in TRACY for the final event (final software version)

TRACY also offers selectable filter criteria in the user interface, based on which, for example, only the structure of individual process steps or only their specific inputs and outputs can be displayed. You can also set how many levels of detail should be displayed. This makes it easy to navigate through the overall VVM method in order to display exactly the content at the level of detail required by the user.

Figure 16, top right, shows the structure of the "Test Planning" method element. As all other dependencies have been removed by the filter, a clear representation of the individual sub-processes that make up the method element is created. At the bottom left, a filter was applied to the inputs and outputs. It can be clearly seen which transfer elements are generated and/or required by the "Identify hazards" sub-process. This corresponds exactly to the elements on the outer edges of an activity in the SysML model. TRACY also offers the option of carrying out impact analyses. Figure 16, bottom right, shows an example of this for the artefact/transfer object "Hazard". The connections depicted show all artefacts that have an influence on the creation of the "Hazard" object. This means that all artefacts are listed that are input variables in processes in the overall model that have "Hazard" as an output object. The opposite representation, which shows, for example, what "Hazard" has an influence on, is also implemented as a function. This makes it possible to quickly identify which elements could be affected by changes.

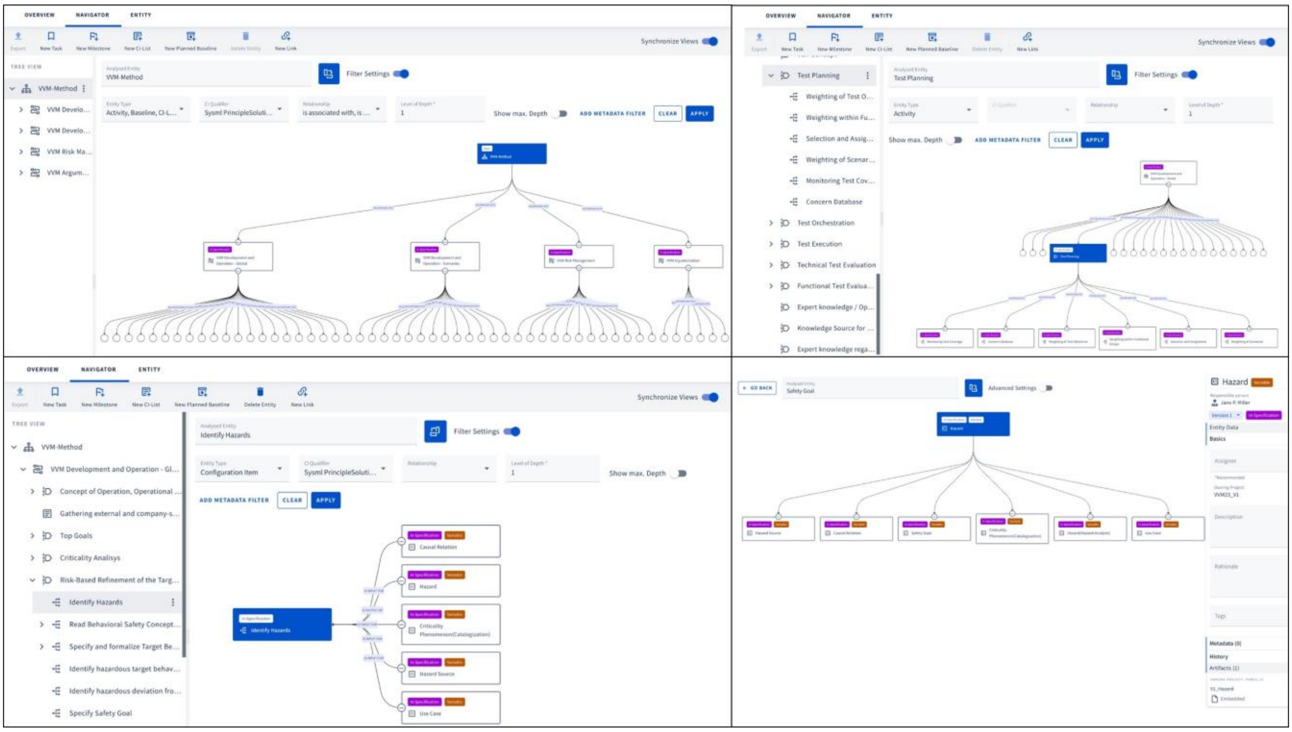


Figure 16: Overview of different display and filter options in TRACY

The implementation in TRACY therefore offers a useful addition to the SysML model, as individual relationships are represented differently or more simply here than is possible in the SysML model. For similar representations in the SysML model, the software used for modelling would sometimes have to be used directly or new diagrams need to be inserted, which is not possible in the HTML derivation.

4 Conclusion and Outlook

This document for Deliverable 14 describes the procedure and the results of the documentation of the overall VVM method in SysML and in the PROSTEP Traceability Demonstrator. Both documentation methods support the industrial instantiation of the VVM project results through the consistent and integrated description of the method on which they are based. In addition, the documentation supports the dissemination of the project results and the exchange with external experts from outside the project, as it creates a formalised basis for discussion through the formal, domain-independent description on the one hand, and on the other hand creates an easy introduction to the overall VVM method through the graphical representation and the top-down approach with different levels of granularity.

For this purpose, the overall VVM method was broken down into interrelated activities and documented in this form. In addition, the transfer and exchange objects for the individual activities are described consistently across the entire method. They thus serve as a "common thread" along the process chain and establish the individual connections between the elements of the method.

In close cooperation with the VVM method project partners, an instrument was created in the form of the SysML model and the mapping of the method in the Traceability Demonstrator, which sustainably supports the accessibility and traceability of the overall VVM method even beyond the project boundaries. It also supports the following points:

- Link and trace information cross domain.
- Steer projects according to top process requirements.
- Collaborate with partners.
- Ensure consistency via formalised description.
- Build a digital thread.
- Analyse impacts.
- Provide a graphical representation of the VVM-Method.
- Establish a navigable "blueprint" for industrial application of the VVM-Method.

At the end of the project, the results of the documentation will be made available to the partners in various ways, both in proprietary and non-proprietary formats, for further use in the organisations. In addition, the final version of the SysML documentation will be made available to the public as an HTML export on the VVM homepage in order to actively promote the exchange with experts and interested parties described above.

5 References

[1] Strategie automatisiertes und vernetztes Fahren; Bundesregierung Deutschland; https://bmdv.bund.de/SharedDocs/DE/Publikationen/DG/broschuere-strategie-automatisiertes-vernetztes-fahren.pdf?__blob=publicationFile; last visted 02.11.2023

[2] Geissen M.: Traceability of Simulation Tasks - Bausteine für die simulationsbasierte Entscheidungsfindung, prostep ivip ProduktDaten Journal 2020-1, 2020

[3] Wachenfeld, W. und Winner, H.: The release of autonomous vehicles, In Autonomous driving, (pp. 425-449), Springer, Berlin, Heidelberg, 2016

[4] DIN ISO 10007:2020-10, Qualitätsmanagement - Leitfaden für Konfigurationsmanagement, Beuth-Verlag, Berlin

[5] Mäder P. und Egyed A.: Do Developers Benefit from Requirements Traceability When Evolving and Maintaining a Software System?, Empirical Software Engineering, Volume 20, Issue 2, pp 413–441, 2015, https://doi.org/10.1007/s10664-014-9314-z