Considering Functional Safety - supporting the development of automated driving vehicles by the use of Model-Based Systems Engineering

Moritz Wäschle*, Kai Wolter*, Katharina Bause*, Matthias Behrendt*, Albert Albers*
*IPEK-Institute of Product Engineering at Karlsruhe Institute of Technology (KIT), Kaiserstraße 10, 76131 Karlsruhe
correspondence: moritz.waeschle@kit.edu

***Abstract*—Starting from first electric functions in vehicles like the braking function realized with the anti-lock braking system (ABS), some of today's vehicles have over 100 control units to achieve an increasing amount of functionality. Functions realize multiple use cases and depend on hardware and software components. This multiple domains development leads to an increase in the complexity of the overall development process. In order to cope with this functional complexity in automated vehicles, new methods for validation are necessary.**

In order to cope with this functional complexity in automated vehicles, new methods for validation are necessary. These new methods need to identify intended and unintended relationships of and within systems and consider safety elements like hazards. In the following contribution, the authors show a new method with an example of distributed test benches for automated trailer transport. The method supports the validation of safety aspects of stakeholder needs through use cases, scenarios, functions and components. Possibilities for automated generation and parameterization of test cases are presented. In order to validate the automated driving functions regarding safety, the IPEK-X-in-the-Loop approach for distributed validation environments is used. Hereby, the automated vehicle systems, as well as the geographically distributed test benches, are considered as System of Systems (SoS). Hence, SoS characteristics are taken into account by validation methods in the context of functional safety.

***Index Terms*—Validation, Model-Based Systems Engineering (MBSE), System of Systems (SoS), Safety of the intended functionality (SOTIF)**

I. Introduction

The exposure to a risk can be determined by considering system elements like functions and components. For instance, according to the functional safety standard ISO 26262, the risk is determined by the frequency of occurrence, the severity and controllability ([1] p.19). Different systems, maneuvers, vehicle- / road states and further elements must be considered to determine the risk [1]. Therefore, one approach is the modeling of elements with their relations to address functional safety needs. The need for linkage is mentioned in ISO 26262 that "for verification, a traceability-based argument can be used" ([2] p.16). This so called "traceability" can be supported by Model-Based Systems Engineering (MBSE), which supports system requirements, design, analysis, verification & validation throughout the whole product development process [3]. Hence, MBSE may help in verification and validation of safety relevant aspects in automated driving.

In this paper, a specific focus lies on how to handle damages in hardware and software. With redundancies in the fulfillment of functions, the possible damages can be compensated. In order to avoid increasing the vehicle's weight and cost by adding too many components, it should be the ambition to have redundancies smaller than two. Redundancy exist by having the same functionality achieved in different already included components. For example, the steering function could be fulfilled only by torque vectoring (i.e. the active drive torque distribution on different wheels) and not by the mechanical steering system. The use case of an unmanned, automated trailer transport (SAE Level 4) with the scenarios docking and undocking of a trailer on a flat plane acts as a representative example. Hereby, electrical and mechanical functions like steering, breaking or environmental perception can be realized in multiple ways. The safety of damaging mechanisms in systems with redundancies smaller than two can be realized with different standards and methods. The validation environment for this example consists of distributed test benches, to validate certain functionalities early in the development process with stationary test benches. These independently organized and operated facilities are connected and result in a System of Systems (SoS).

II. State of Research

This publication is part of the research project "SmartLoad", founded with the focus on new methods for reliability enhancement of highly automated electric vehicles. In previous investigations, the authors describe a toolchain for developing a concrete scenario from general use cases and a STPA modeled in SysML in the context of distributed validation environments. [4]

The following two sub-chapters cope with the previous and current research activities. Firstly, the validation activities in the context of SoS Engineering are introduced. Secondly, functional safety with its standards and further publications are described.

A. Understanding of Validation in (System of) Systems Engineering

According to the International Council of Systems Engineering (INCOSE), “Systems Engineering is a trans-disciplinary and integrative approach to enable the successful realization, use, and retirement of engineered systems, using systems principles and concepts, and scientific, technological, and management methods. [5]” The validation in Systems Engineering can be realized with the IPEK X-in-the-Loop approach (IPEK-XiL approach). The approach focuses on a system (System-in-Development (SiD)) in its environment and therefore describes that a validation of a system needs to consider its Connected Systems (i.e. remaining systems, environment). The “X” stands for this SiD, for instance on system-level a highly automated vehicle or on subsystem-level a clutch. Hence, the whole validation environment consists of the SiD, Connected Systems as well as of methods and resources to validate the SiD. [6]
As a result, the IPEK-XiL approach, with its physical, virtual or mixed systems, addresses this need of validation.
An autonomous vehicle with interaction to other systems like infrastructure or vehicles can be described as a SoS. According to Maier and Dahmann, SoS consists of the two core characteristics: operational and managerial independence.

The first independence results in each system interacting independently and with its own purpose. The latter consists of the independent organization of these individual systems, which are part of the SoS, to fulfill their purpose. Further characteristics are summarized in Nielsen et al. and include [7]–[10]:

- emergence of behavior: the combination of systems delivers results that cannot be achieved by individual systems
- evolution: changes in systems over time. ”The development of a SoS is never completed” [11].
- dynamic reconfiguration: SoS can change structure and its composition, sometimes even in real-time
- interdependence: links between systems with mutual dependencies of elements to achieve a common goal.
- interoperability: SoS integrate multiple heterogeneous systems with interfaces, standards and protocols.
- geographic distribution: systems are dispersed and need connections.

B. Functional Safety of automated vehicles

Leveson differs between three safety nomenclatures and focuses for future use on ”safety III” with the goal of handling hazards. [12] Furthermore, she introduces the method STPA to handle the whole safety process. In the White Paper on ”How to Perform Hazard Analysis on a “System-of-Systems””, Leveson extends the approach for SoS. By arguing that SoS are systems with some specific characteristics, she states that new safety methods like STPA need to be introduced for these new formed systems with emergent behavior. [12]

For functional safety, the twelve public standards of ISO 26262 provide an overview of the automotive product development.

The standard “Safety of the Intended Function” (abb. SOTIF, norm ISO/PAS 21448) extends the safety aspects with the focus on intended functionality and the division between four fields of safe and unsafe, respectively, known and unknown scenarios. For example, in the verification activity, the system needs to perform well in known safe and unsafe conditions. For validation, an accumulation of relevant test cases should be considered based on known behavior e.g. from accident numbers. In the concept phase, the first process “creation of the function and system specification” is realized with the language SysML. The documentation must be iteratively updated and consists of objectives of the described function, dependencies to functions and systems, environmental conditions and human machine interface interaction. The specification phase focuses on “planning of verification of validation”, evidence of the robustness of the system. The “identification and evaluation of critical use cases” and “measures to reduce SOTIF-risks”. In the proof phase, the “verification”, “validation” and “releases of SOTIF” are considered. [13]

Birch et al. apply the SOTIF standard for the highly automated driving functions. The authors introduce a safety argument structure and highlight the role of the operational design domain for assuring safety. [14]

In the context of ISO 26262 and ISO/IEC 15504 for Process Assessments - Automotive Spice (ASPICE) was developed to support the assessment of process quality. The VDA (Verband der Automobilindustrie) has published this guideline to give rules and recommendation for processes in automotive industry. [15] Especially the nomenclature, the classification of use cases, systems, functions and the suggested processes have got relevance for this publication.

The ASAM standards OpenX (X can stand for CRG, Scenario or Drive) and OSI (Open Simulation Interface) contribute to a uniform description of driving maneuvers and interfaces in the automotive development. Marko et al. tested version three of OSI for the integration into co-simulation. [16]

Within the Project “PEGASUS” [17], the testing of highly automated driving systems is defined and described in methods. Six scenario layers were identified to systematically describe a scenario. The main outcome is the PEGASUS method for the assessment of highly automated driving functions. [17]

In the whitepaper “Safety first for automated driving” conducted of multiple companies, the verification and validation for automated driving is described. Especially the following questions arise [18]:

- Why is the test necessary and how good does the test result need to be?
- How are the tests conduced?
- Where is the test performed?
- What are the test elements and System-under-Investigation?

While the described contribution consider aspects of SoS, (Model-Based) Systems Engineering and safety, this contribution combines these aspects of different domains.

III. Research Focus

The objective of this contribution is the development of a method to support the validation of automated vehicles. The following research questions are answered:

- How can functional safety approaches and methods be modeled in the validation environment to support the validation of automated vehicles with distributed test benches?
- How to consider SoS characteristics in the integration of functional safety approaches and methods to support the validation of automated vehicles with distributed test benches?

IV. Method in the Context of Functional Safety

By using MBSE, different elements and relations can be modeled and describe the automated vehicle as well as its validation environment. Hereby, Table I gives an overview by the classification of the relevant models (c.f. [19]).

TABLE I
Classification of results of the project "SmartLoad" in the modeling structure according to [19]

Model	Description and application
Meta	SysML with profiles like "Safety & Reliability Analysis"
Reference	SmartLoad-structure describes element classes and interactions
Implementation	Implemented validation environment with distributed test-benches
Application	Demonstrator of an automated vehicle

In order to describe the interrelations of element types, Fig. 1 visualizes a reference model in the language SysML. The reference model uses the most frequently used elements in MBSE in the context of safety. Furthermore, methods and elements specific to safety approaches are considered (see ISO 26262&SOTIF block) and linked to the previous elements. Thus, the reference model visualizes the two main areas, SOTIF known and unknown, with the purpose to reduce the unknown area with safety consideration. The implementation was done in the tool Cameo Systems Modeler by means of adaptations of the approach ”Functional Architectures of Systems for Mechanical Engineers” [20]. Considering functional safety and safety of the intended function standards, the elements and their implementation in MBSE are explained in the next paragraphs. Here, the exemplary application of the trailer use case as the application model is mentioned in italic after each paragraph.

1) Stakeholder: Stakeholder are modeled in an use case diagram. For validation purposes, the stakeholders' need are necessary to validate the product. The stakeholders are therefore linked via an association connection to use cases in which they interact.

The vehicle with trailer needs an operator for parking processes and one tele-operator as a safety backup. Furthermore, passengers as well as people in the environment like pedestrians must be considered. Moreover, insurance companies, producers, service provider and stakeholders of interacting systems can be modeled.

2) References: According to the PGE – Product Generation Engineering a newly designed product is based on reference system elements with three possible variation types: "carry-over, embodiment and principle variation [21]. For instance, the use case of an automated docking process of a tugboat can be a helpful basis for vehicle docking with trailer. Hence, reference system elements not only for use cases but for all elements are considered.

The vehicle with trailer of SAE Level 4 is based on products of SAE Level 2 and 3 as well as on other products.

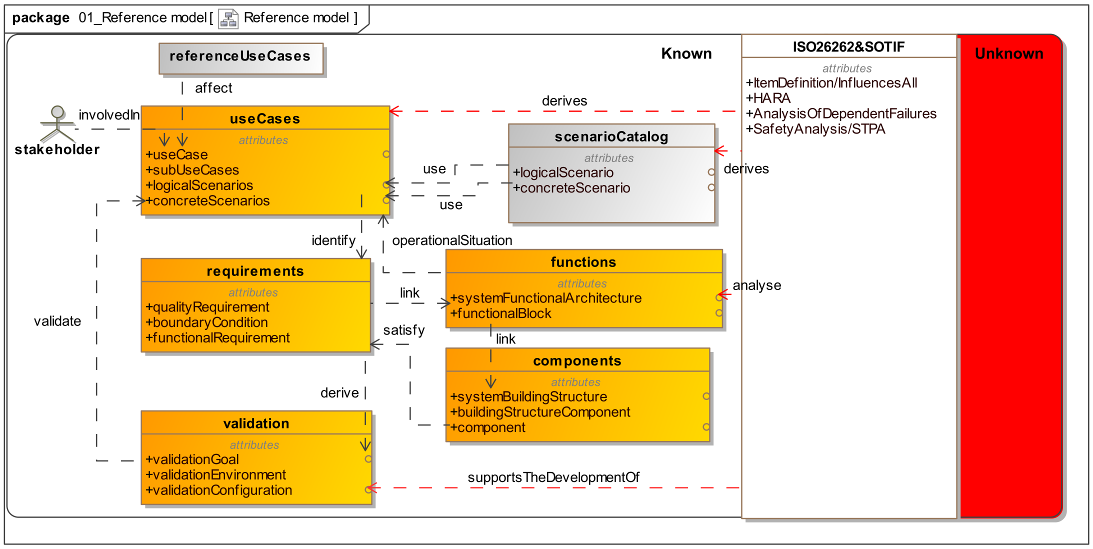


Fig. 1. Reference model in SysML

3) Use cases: Based on the reference system and stakeholder needs, use cases can be considered. Three main use cases for transportation are people, parcel and trailer transport. For these use cases, there can be more specific sub use cases derived. These sub use cases are linked with an include connection to logical and from there to concrete scenarios (c.f. [4], [17]).

The main use case is the trailer transport with sub use cases for the docking process in a rural area. Hereby, multiple sub use cases can be defined like the docking and undocking of the trailer. The concrete scenario "docking on a flat plane" contains events like maneuvering with concrete parameter sets.

4) Scenario Catalog: In the scenario catalog, predefined scenarios from different reference elements like previous system models are used as well as scenarios resulting from methods like STPA. Concrete scenarios are selected to validate functions based on the reference model. With respect to the SOTIF standard, validation of the uncertain control actions against these scenarios is important to demonstrate that the uncertainties can be managed except for a previously defined residual risk. For instance, the control action of the transmission of the data set for trajectory planning needs to be managed by considering the timing, order of the data or degradation of systems.

In modeling, the derivation of a generalized scenario from a loss scenario is clarified using the derive relationship. In this way, the traceability can be ensured. [22]

The scenario catalog is based on PEGASUS scenario catalog [23]. In our example, different scenarios are described by linking them to characteristics of each of the six layers in the Project PEGASUS [17].

5) Requirements: The connection between requirements and further elements in the system structure is described in [4]. Further publications argue that requirements must be considered with their relation to the system infrastructure (c.f. [15]).

According to ISO 26262, the requirements of the exemplary use case must contain the operation mode, fault tolerance, time interval, safe states, emergency operation time interval, functional redundancies. Furthermore, each requirement should specify multiple strategies like fault tolerance. The developed method within the Project "SmartLoad" uses tables in SysML to specify every part of the requirement. The norm suggests the use of methods like FMEA to support the generation of a complete set of requirements. The requirements are derived from goals. In the reference model, the goal is integrated in the requirements. This is based on the safety goal definition as a a top-level safety requirement ([2] p. 10).

E.g. the goal is described for the hazardous event of the crash during the docking of the trailer with the ASIL Level A considering the controllability of the event, the exposure and the severity. The safety goals are determined for each hazard by an assessment of hazards and risks. ([2] p.10)

6) Functions: According to the approach Functional Architectures for Mechanical Engineering (FAS4M) the elements functional architecture and functional blocks are necessary to describe the functions and their relations [20]. In addition, an aggregation of several functions and use cases is considered in order to evaluate groupings of elements. The standard ISO 26262 focuses on safety related functions, which have got the potential to be part of a violation or achievement of a safety goal ([2] p. 23).

The function automated backwards parking can be realized. For evaluation of safety aspects, the aggregation of all parking functions needs to be considered.

7) Components: According to FAS4M the system building structure is defined to show the links between components [20].

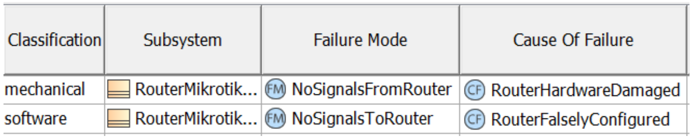
The components of a steering actor and the engine with gear and connection to wheel can realize the function steering for the trailer parking.

8) Validation: In the context of ISO 26262, the term "safety validation" is defined as an "assurance based on the examination and tests, that the safety goals are adequate and have been achieved with a sufficient level of integrity ([2] p.24)".

If we consider safety validation in the IPEK-XiL approach, safety methods must consider not only the System-in-Development (SiD), but also the residual system with its interacting environment. The distributed validation environment in the Project "SmartLoad" consists of geographically distributed test benches with the objective to validate different degrees of maturity of stakeholder needs. The modeling contains a dependency matrix and a Block Definition Diagram to provide an overview of the distributed and networked test benches with the system building structure in a logical and physical architecture. In this way, a flexible validation depending on the current SiD can be achieved. In order to establish a linkage and communication between the distributed test benches so called Koppelsystems are needed [6]. Real Koppelsystems may have some influence on the connected test benches. In the system model, it is possible to evaluate which Koppelsystem is required between different locations. For instance, a Koppelsystem between a physical and virtual model consists of a connection between a vehicle and a virtual environment.

A process how to use the elements in a validation process is realized with a previously published toolchain (c.f. [4]). Besides FMEA, further methods like STPA or FTA can be used as well (c.f. control structure of a uphill drive according to STPA in [24]).

An example for the modeling benefit of MBSE is shown in Fig. 2 with an extract of a FMEA. The table contains the linked elements: classification, item, subsystem, failure mode, local effect of failure, cause of failure and prevention control. The FMEA helps to systematically identify failures and to protocol failures for future need. In total, the table has over thirty *identified failures in context of the validation of connected test benches.*



Classification	Subsystem	Failure Mode	Cause Of Failure
mechanical	RouterMikrotik...	FM NoSignalsFromRouter	CF RouterHardwareDamaged
software	RouterMikrotik...	FM NoSignalsToRouter	CF RouterFalselyConfigured

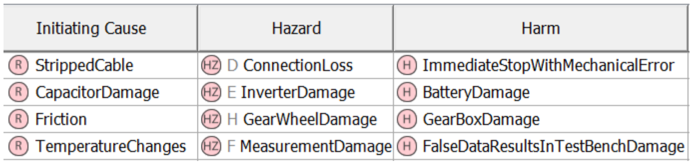
Fig. 2. Extract of the FMEA of distributed validation environments

9) ISO 26262 & SOTIF: Considering functional safety, the authors include multiple SysML profiles, which contain predefined element types and diagrams to extend a reference model. These element types have an impact on element types like the scenario catalog. For instance, the risk table in the plug in Safety Analysis allows to show necessary information about safety aspects in a predefined way and supports the derivation of new scenarios.

ISO 26262 especially mentions the necessity of feedback loops with the terms “cascading failures” and “dependent failures”:
A cascading failure of an element or item can cause further failures of other elements or items. An item is a system or a combination of systems that implements a function. A (statistically) dependent failure is for instance the probability of single combined failures which are not equal to the combination of all failures. ([2] p. 4)

Further needs are mentioned in ISO 26262-1 - like bidirectional traceability or different functions implemented with identical hardware - reinforce the idea of the use of MBSE [2].

Metrics can be derived eg. in the context of the method HARA (see Fig. 3). The method identifies and categorizes risks based on three factors. These factors are severity (S) of the potential damage, exposure (E) of the operating situation, and controllability (C) of the hazardous event. An Automotive Safety Integrity Level (ASIL) is assigned to a hazard based on the analysis of driving situations with respect to the factors S, E, and C. The ASIL is then assigned to the hazard. From this, a safety objective is formulated to avoid an unreasonable risk and requirements for risk reduction are derived. [25]



Initiating Cause	Hazard	Harm
R StrippedCable	HZ D ConnectionLoss	H ImmediateStopWithMechanicalError
R CapacitorDamage	HZ E InverterDamage	H BatteryDamage
R Friction	HZ H GearWheelDamage	H GearBoxDamage
R TemperatureChanges	HZ F MeasurementDamage	H FalseDataResultsInTestBenchDamage

Fig. 3. Extract of a risk table considering hazards of gear test bench and electric motor test bench

By using safety elements, the hazards can be identified and linked to other elements in the system. The authors examine the SoS characteristics of the connected test benches. This results in emergent behavior with new causes and dependent failures (see Fig. 4). The hazards are pulled up of specific test bench configurations. But new causes need to be considered. These SoS-causes are partly shown in the FMEA in Fig. 2.

Further SoS characteristics like managerial independence and interoperability are taken into account by overviews to improve the communication between different independent parties. Hence, Block Definition Diagrams, contracts (c.f. [4]) and tables are used to discuss the interfaces between different test bench environments. Hereby, Koppelsystems need to be modeled.

The SoS characteristics of evolution and dynamic reconfiguration are applied by using reference system elements and modular structures in MBSE. For instance, reference use cases help to identify similarities and carryover variations between use cases. Hereby, factors like SoS characteristics, as well as functional architectures need to be taken into account. Especially the modeling of functions in a functional architecture realizes a high degree of carryover variation. Thus, the architectures with its partly abstract functional elements, like the general function of steering, support the reconfiguration of the SoS. Furthermore, in the context of validation, IPEK-XiL uses modularity to support the reconfiguration of validation elements.

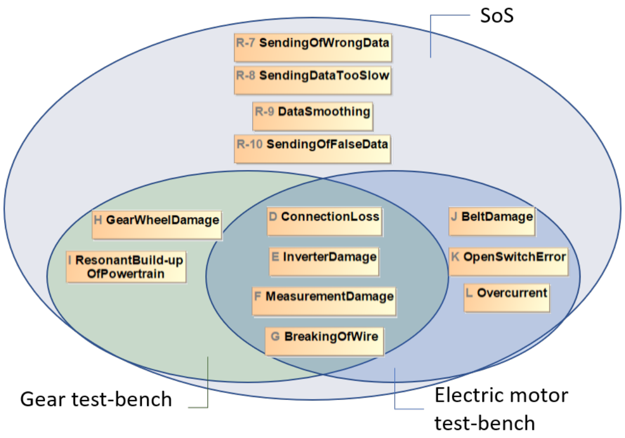


Fig. 4. Extract of hazards of the test benches (gear and electric motor) as well as shared causes considering the SoS character of the connected test benches

One possible outcome of the dependency modeling is shown in the Safety and Reliability Analysis Map (see Fig. 5). The figure visualizes as an example stripped cable as a safety relevant item and their linkage to harm. Hence, cross-systems interdependence of SoS can be modeled and include multiple elements. Furthermore, the map legend starts with FMEA items and has the terminology to comply with SOTIF standard from low information to more information and more critical of hazard, hazardous situation and harm. The functions are linked to use cases and support the identification of hazardous events.

An exemplary consideration of an hazard is the trailer which sends out wrong location information to the docking vehicle. This results in a cascading failure. Hence, measures should be specified during the development phase.

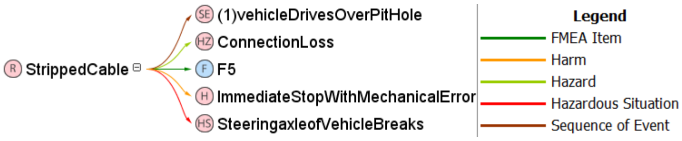


Fig. 5. Safety Analysis Item “StrippedCable” in a Reliability Analysis Map

V. Discussion

This contribution considers multiple safety related standards of the automotive domain. Artificial Intelligence specific use cases or functions are not considered and will be dealt with in the future standard “Safety and artificial intelligence” (ISO/AWI PAS 8800). A comprehensive use of the MBSE can be achieved with connected federated systems like safety specific expert tools. In this contribution, the modeling support was achieved with different diagrams during the product development process. However, some parts are only modeled retrospectively.

With regard to modeling technique, Ulrich identified that in an automobile, the complexity is high with over 10,000 parts and five levels of decomposition [26]. To handle complexity, one element should have less than six interfaces. Thus, only the necessary links should to be directly modeled. The number of relevant links is also depending on further topics such as security. Hence, this contribution with the focus on functional safety contemplates one important topic and should not result in a very complex structure already. However, multiple standards focus on safety and especially with automated System of Systems, safety becomes an important topic with multiple approaches which support each other. [29]

VI. Summary

This contribution focuses on safety aspects and develops a reference model for the comprehensive validation of automated vehicles. It takes relevant functional safety publications and standards like ISO 26262 into consideration. With the support of MBSE, the authors develop a reference model to link elements and validate the exemplary trailer transport with distributed test benches. The reference model consists of safety-related SysML profiles and further extensions like the table based scenario catalog. Metrics and views in diagrams and tables support functional safety in automated vehicles. Hereby, metrics like ASIL as well as SoS characteristics are taken into account and result in new diagrams and links between elements.

To conclude, there is an effort to set up the model with its linkages. However, it supports the consideration of safety relevant methods and standards, as well as the identification of effective chains and decisions about validation environment.

Acknowledgment

This work has been supported by the Federal Ministry for Education and Research (BMBF) in the project “New methods to increase the reliability of highly automated electric vehicles (SmartLoad)” with the funding reference number: 16EMO0363.

References

[1] ISO, “ISO 26262-3:2018(en) road vehicles — functional safety — part 3: Concept phase,” 2018-12.

[2] ’ISO, “ISO 26262-1:2018(en) road vehicles — functional safety — part 1: Vocabulary,” 2018-12.

[3] International Council on Systems Engineering (INCOSE) - Technical Operations, “Incose systems engineering vision 2020.” [Online]. Available: http://www.ccose.org/media/upload/SEVision2020_20071003_v2_03.pdf

[4] M. Wäschle, K. Wolter, C. Han, U. Pecha, K. Bause, and M. Behrendt, “Validation concept for scenario-based connected test benches of a highly automated vehicle,” in *Automatisiertes Fahren 2021*, T. Bertram, Ed. Wiesbaden: Springer Fachmedien Wiesbaden, 2021, pp. 95–109.

[5] INCOSE, “Systems engineering and system definitions,” 2019.

[6] A. Albers, T. Pinner, S. Yan, R. Hettel, and M. Behrendt, “Koppelsystems: Obligatory elements within validation setups,” in *Proceedings of DESIGN 2016*, 2016.

[7] M. W. Maier, “Architecting principles for systems-of-systems,” *INCOSE International Symposium*, vol. 6, no. 1, pp. 565–573, 1996.

[8] M. W. Maier, “Architecting principles for systems–of–systems,” *Systems Engineering*, vol. 1998, no. 4, pp. 267–284, 1998.

[9] J. Dahmann, “Integrating systems engineering and test evaluation in system of systems development,” in *2012 IEEE International Systems Conference SysCon 2012*, 2012, pp. 1–7.

[10] C. B. Nielsen, P. G. Larsen, J. Fitzgerald, J. Woodcock, and J. Peleska, “Systems of systems engineering,” *ACM Computing Surveys*, vol. 48, no. 2, pp. 1–41, 2015.

[11] A. Albers, N. Peglow, J. Powelske, C. Birk, and N. Bursac, “Coping with complex systems-of-systems in the context of pge – product generation engineering,” 2018.

[12] N. G. Leveson, “White paper on how to perform hazard analysis on a “system-of-systems”,” 2018.

[13] L. Schnieder and R. S. Hosse, *Leitfaden Safety of the Intended Functionality*. Wiesbaden: Springer Fachmedien Wiesbaden, 2020.

[14] J. Birch, D. Blackburn, J. Botham, I. Habli, D. Higham, H. Monkhouse, G. Price, N. Ratiu, and R. Rivett, “A structured argument for assuring safety of the intended functionality (sotif),” in *Computer Safety, Reliability, and Security. SAFECOMP 2020 Workshops*, ser. Lecture Notes in Computer Science, A. Casimiro, F. Ortmeier, E. Schoitsch, F. Bitsch, and P. Ferreira, Eds. Cham: Springer International Publishing, 2020, vol. 12235, pp. 408–414.

[15] VDA QMC Working Group 13 / Automotive SIG, “Title: Automotive spice process assessment / reference model: Version 3.1,” 2017.

[16] N. Marko, J. Ruebsam, A. Biehn, and H. Schneider, “Scenario-based testing of adas - integration of the open simulation interface into co-simulation for function validation,” in *SIMULTECH*, 2019.

[17] “Scenario description: Requirements & conditions – stand 4,” o.J. [Online]. Available: https://www.pegasusprojekt.de/files/tmpl/PDF-Symposium/04_Scenario-Description.pdf

[18] “Safety first for automated driving,” Whitepaper, 2019.

[19] S. Muschik and A. Albers, “Development of systems of objectives in early product engineering, entwicklung von zielsystemen in der frühen produktentstehung,” *1615-8113*, 2011.

[20] G. Moeser, “Kurzversion fas4m methodenguideline: Methodisches vorgehen im fas4m-ansatz; kurzversion der 24-seitigen methodenguideline (konkrete handlungsanleitungen) zur veröffentlichung,” 2016.

[21] A. Albers, N. Bursac, and E. Wintergerst, “Produktgenerationsentwicklung – bedeutung und herausforderungen aus einer entwicklungsmethodischen perspektive,” in *Stuttgarter Symposium für Produktentwicklung 2015 SSP 2015*, B. Binz, Ed., 2015.

[22] M. Kozok, “Modellierung von abhängigkeiten in einer validierungsumgebung für hochautomatisierte elektrische fahrzeuge mit hilfe von mbse-ansätzen,” Unveröffentlichte Bachelor-/arbeit, Karlsruher Institut für Technologie (KIT), Karlsruhe, 2020.

[23] “Pegasus,” 26.11.2020. [Online]. Available: https://www.pegasusprojekt.de/de/

[24] M. Wäschle, K. Wolter, K. Bause, and M. Behrendt, “A new safety oriented approach in the validation of highly automated electric vehicle subsystems,” 2020.

[25] ISO, “ISO 26262-2:2018(en) road vehicles — functional safety — part 2: Management of functional safety,” 2018-12.

[26] K. T. Ulrich and S. D. Eppinger, *Product design and development*, 5th ed. New York, NY: McGraw-Hill, 2012.