VERIFICATION
VALIDATION
METHODS

VVM Deliverable 06

Refined qualities, system- and test requirements

Version 1.0

Editor Jürgen Nuffer

Project coordination Robert Bosch GmbH and BMW AG

Due date 31/12/2023

Creation date xxx

Date of publication xxx

PEGASUS FAMILY

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

aufgrund eines Beschlusses des Deutschen Bundestages

Document information

Authors

Jürgen Nuffer, Simon Kupjetz (LBF)
Martin Mai, Martin Dörr (ZF Friedrichshafen AG)
Björn Filzek (Conti)
Christoph Doyscher (IAV)

Reviewer

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	20.04.2023	Template prepared	Nuffer, Kupjetz	LBF
0.2	30.6.2023	Ch1	Nuffer	LBF
0.3	17.07.2023	Ch2	Nuffer, Mai, Doyscher	LBF, ZF, IAV
0.4	11.09.2023	Ch3	Filzek, Dörr, Mai, Nuffer	Conti, ZF, LBF
0.5	04.11.2023	Ch4, 5, 6	Filzek, Dörr, Mai, Nuffer	
1.0	21.12.2023	Final chapters and typo, Errata	Kupjetz, Nuffer, Mai	LBF, ZF

Table of Content

Table of figures

List of tables

1 Introduction

This deliverable is based on the previous deliverable D03, where first system and test requirements as well as several methodological tools within the V & V domain were presented.

In this deliverable, enhanced quality measures based on functional and behavioral safety requirements will be shown. also, the application of the methods resulting in the derivation of a test specification as main delivery artefacts to the test orchestration is described. Within the assurance framework already presented in the previous deliverable D03, this deliverable covers the flux of information from the functional architecture via test planning to test requirements which serve as a pre-requisite for the test orchestration. This is highlighted within the red box in fig. 3.1. In addition, it describes the input-output - Structures between these artefacts, which are represented in the assurance framework as bold purple arrows. These arrows summarize the main interconnection between these artefacts, ad in this deliverable it is highlighted, how these interconnects work in detail.

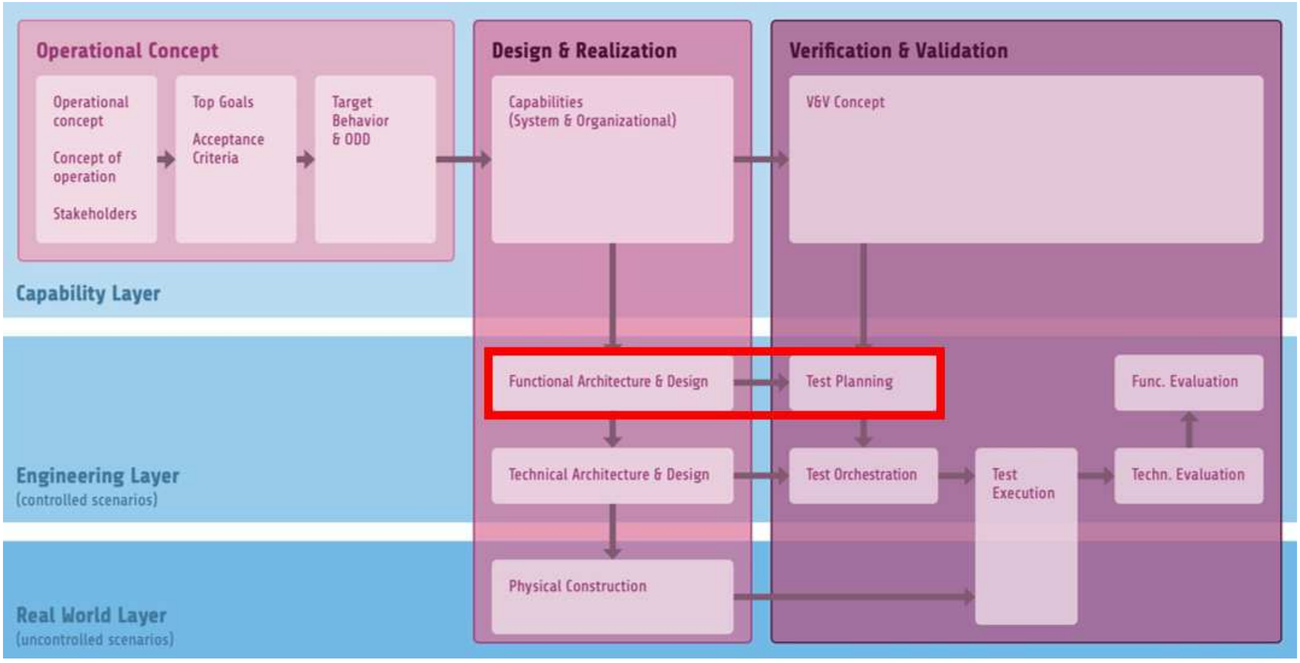


Figure 1: Assurance Framework

2 Detailing of test planning

2.1 Test planning artefacts

In the test planning, the set of all functional test cases is defined. A single or group of similar functional test cases is described by a functional test specification that bundles all functional test requirements relevant to that test case or test case group. Thus, the functional test specifications represent the functional test space as the central output of the test planning. In essence, it is a matter of selecting the corresponding functional test objects and scenarios and assigning them to each other with the desired test goals, i.e. connecting them to test cases.

The selection of functional test objects is based on the functional design, described in the form of a functional architecture at different levels of abstraction, also called functional system levels, and the associated functional requirements including their qualities.

The test scenarios can be selected from the total set of all evaluated scenarios. These, also defined as "Logical Scenario Instances" in the context of VVM, fully represent the ODD and are evaluated in their probability of occurrence.

The motivation for the test case is expressed in the test goals. First, a basic distinction is made here between verification and validation. While verification checks whether the product meets the requirements, validation pursues the question of the extent to which the product fulfills its intended use. In both cases, it is concretely described what is to be measured to generate the evidence needed for argumentation.

In real-world, the possible combinations of test objects, environments and goals are endless. To limit oneself to a set of test cases that can be implemented in practice, weighting is carried out according to the areas that are sensitive to safety. For this purpose, a model is needed that evaluates and quantitatively describes the individual functions and scenarios from the decomposition of functional design and ODD with regards to their significance for safety.

A special case in this initially top-down structured approach are the concerns. These are indications of a subjective (e.g. assessments) or objective (e.g. measurement results) nature, which must be given special consideration with regard to the scope of the test. Concerns can be divided into different classes. For example, functional concerns refer to functional design, while scenarios concerns refer to scenarios that need to be considered. According to their classification, such concerns are received at the appropriate point to weight their relevance for safety in the context of the functional design or the logical scenario instances.

The core process of test planning outlined so far deals with how single test specifications are created. Through repeated application, a total set of test specifications is formed, which, as stated above, can only span a limited test space at a time. For the argumentation, it is of great importance to be able to assess the coverage of this test space. Since the functional design, the scenarios and the different types of safety have been linked in test cases, the statement on test coverage must always include all 3 dimensions. Other criteria such as feasibility and efficiency are in contradiction. They are in direct conflict of objectives with the preferably highest test coverage and complete the picture of the extent to which these requirements for the test concept are also met.

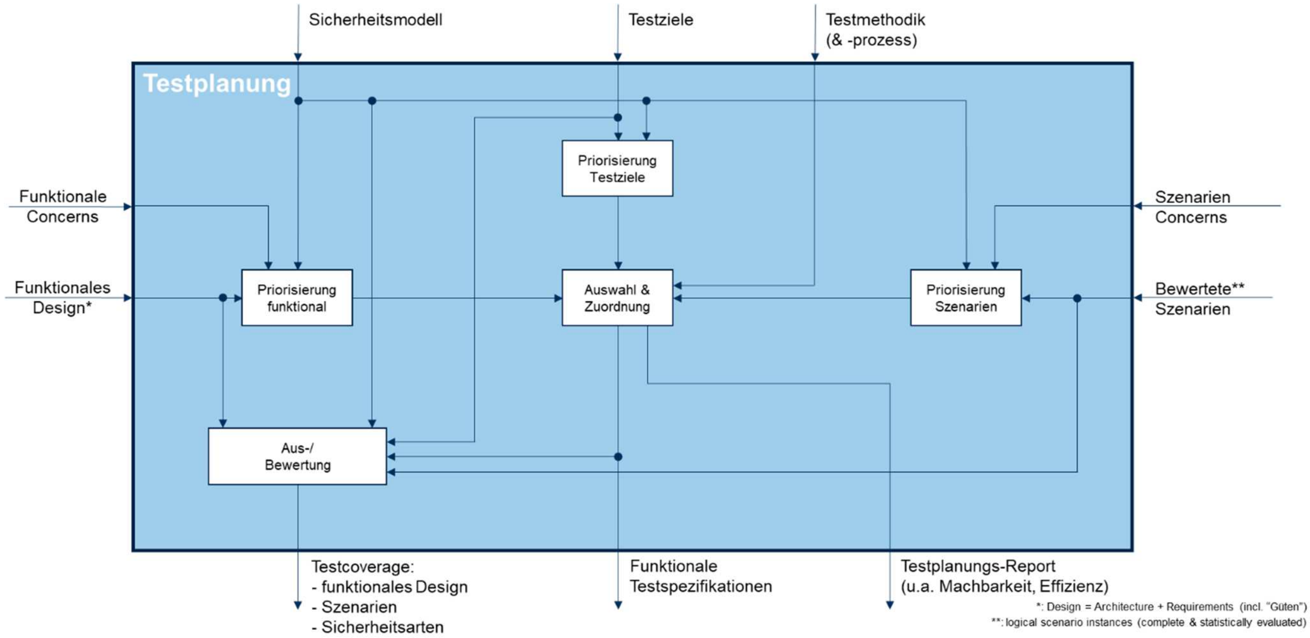


Figure 2: Details of Test planning

Test planning receives strategic guidelines and framework conditions from the V&V concept. Such requirements and constraints include, among other things, test methods to be applied (e.g. verification and validation) or an overarching test process in which test planning is integrated.

3 Functional Design as Input for Test Planning

One of the fundamental inputs of test planning is a functional design. In VVM we therefor collected exemplary requirements and defined a simple functional architecture of an AD system.

A functional design is a process and result of defining the working relationships among the components of a system. In this case, the term components of a system refers to functional elements of a system. (1)

A functional architecture is a hierarchical arrangement of functions, their internal and external (external to the aggregation itself) functional interfaces and external physical interfaces, their respective functional and performance requirements, and their design constraints. (1)

3.1 System Requirements

System Requirements are based on stakeholder needs and behavioral requirements.

Stakeholder Needs are collected from various sources, like e.g., functional use cases, an item definition or customer functions.

Behavioral Requirements specify stakeholder requirements towards behavior in a scenario-specific context.

Applying the **Goal/Question/Metric (GQM) method** (2) on selected functional requirements results in quality measures. They are formulated as performance requirements, refining the functional requirements.

Functional Requirements are statements that identify what results a product or process shall produce. (1)

Performance requirements are measurable criteria that identify […] how well functional requirements shall be accomplished. (1)

3.2 Functional Architecture and Design

A **Functional Architecture** satisfies the system requirements by defining sufficient functionality. While elaborating the functional architecture, the system requirements are refined in parallel. By linking requirements to functions and functional interfaces, one can track the progress and fill gaps.

In the VVM project the resulting functional hierarchy is based on the **Sense-Plan-Act Methodology** (3) known from robotics and automation in general. The functional architecture in this example consists of a hierarchical arrangement of functions, their internal and external functional interfaces, and links to their respective functional and performance requirements. The relationship between functional elements is partially described by diagrams.

.

3.3 Handover to Test Planning

The functional elements of the functional architecture are handed over as test objects to test planning. The functional elements bundle up their related system requirements. The contained performance requirements are goals when specifying pass/fail criteria for testing.

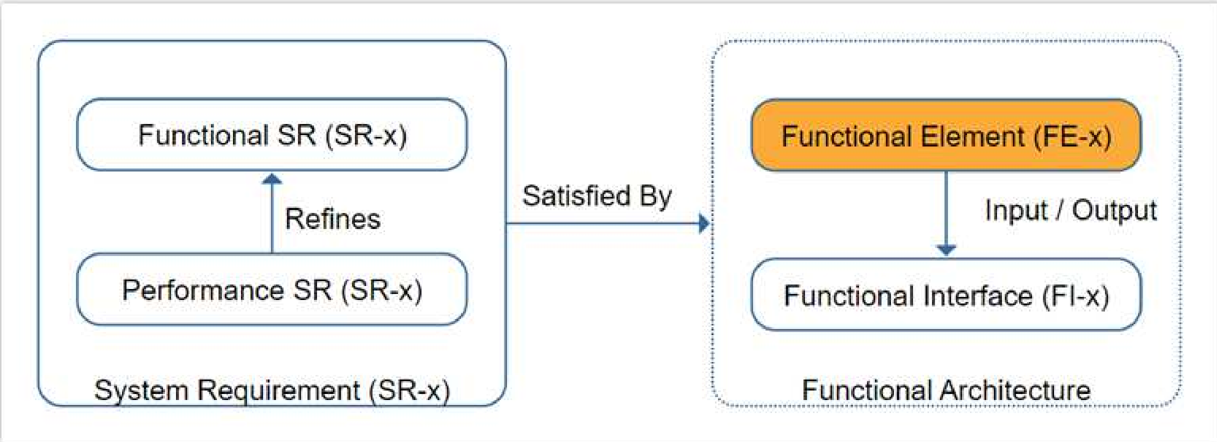


Figure 3: Functional Elements (FE), their Functional Interfaces (FI) and related System Requirements (SR)

Example

Regarding e.g., the classification of pedestrian crosswalks, the functional element *(FE-1.1.3.1) classification of road networks and traffic guidance objects* shall be tested by planning the verification of its related requirements.

Title	System Requirement	Type
(SR-4.1.2) crosswalk classification	The system shall classify crosswalks.	Functional
(SR-4.1.2.1) crosswalk marking classification	The system shall classify broad stripes on the road for crosswalk markings.	Functional
(SR-4.1.2.2) crosswalk sign classification	The system shall classify traffic signs for crosswalks.	Functional
(SR-4.1.2a) crosswalk position accuracy	The system shall classify crosswalks with a deviation to the exact position of less than 0.3 meters.	Performance
(SR-4.1.2b) crosswalk detection rate	The system shall classify crosswalks during free sight within 50 meters in more than 99% of the cases.	Performance
(SR-4.1.2c) crosswalk detection distance	The system shall classify crosswalks within a distance of at least 50 meters.	Performance

(SR-4.1.2d) crosswalk detection range	The system shall classify crosswalks within a distance of more than 0.3 meters and at least 50 meters.	Performance
(SR-4.1.2e) crosswalk reaction time	The system shall classify crosswalks at least 5 seconds before reaching it.	Performance
(SR-4.1.2f) crosswalk detection time	The system shall classify crosswalks within maximum 0.5 seconds.	Performance

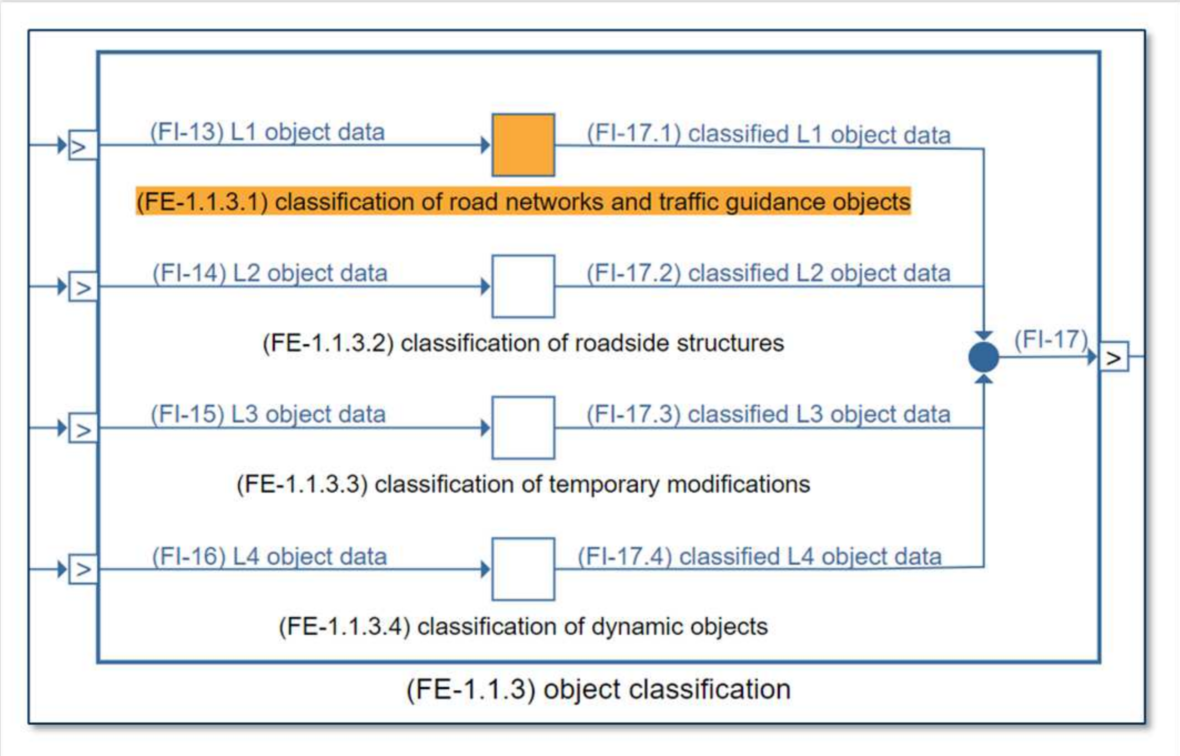


Figure 4: Example relationship of Functional Element FE-1.1.3.1

3.4 Functional Safety Aspects

Many requirements for automated driving are motivated by safety concerns.

Functional safety requirements (FSRs) in our case on system level are a subset of the system requirements (SRs). Be aware of the two different meanings of “S” in the abbreviations, standing once for “Safety” and once for “System”.

In the VVM project we focus on two sources for functional safety requirements:

Safety goals (SGs) are **top-level safety requirements** as a result of the hazard analysis and risk assessment at the vehicle level. (1)

Behavioral safety requirements (BSRs) specify solution-independent safety-related behavior or solution-independent safety measures including their safety-related attributes.

The concept of behavioral safety requirements to derive functional safety requirements was introduced in VVM towards the end of the project. As defining and analyzing safety goals is a well-known and established way to derive functional safety requirements, both approaches had been followed in parallel.

The following example shows, how safety goals, behavioral safety requirements and functional safety requirements could be aligned.

Example

1) Safety Goal

SG-001	<u>**Prevent**</u> the collision of an ego vehicle with ***vulnerable road user*** in the crosswalk influence area.

Figure 5: Safety Goal SG-001

2) Behavioral Safety Requirement

BSR-001 crosswalk detection

Nominal Risk Reduction: An autonomous driving vehicle must <u>be capable to detect</u> crosswalks to take any <u>relevant</u> crosswalk ahead into account when making driving decisions.

Integrity: An autonomous driving vehicle must be able to reliably detect crosswalks to sufficiently cover implementation uncertainties.

Figure 6: Behavioral Safety Requirement BSR-001

3) Functional Safety Requirement with Performance Requirements

Title	**System Requirement**	**Type**
(SR-4.1.2) crosswalk classification	The system shall classify crosswalks.	Functional **Safety**
(SR-4.1.2a) crosswalk position accuracy	The system shall classify crosswalks with a deviation to the exact position of less than 0.3 meters.	Performance
(SR-4.1.2b) crosswalk detection rate	The system shall classify crosswalks during free sight within 50 meters in more than 99% of the cases.	Performance
(SR-4.1.2c) crosswalk detection distance	The system shall classify crosswalks within a distance of at least 50 meters.	Performance
(SR-4.1.2d) crosswalk detection range	The system shall classify crosswalks within a distance of more than 0.3 meters and at least 50 meters.	Performance
(SR-4.1.2e) crosswalk reaction time	The system shall classify crosswalks at least 5 seconds before reaching it.	Performance

(SR-4.1.2f) crosswalk detection time	The system shall classify crosswalks within maximum 0.5 seconds.	Performance

4) Constraints

- The ego vehicle speed is **30 km/h** (see VVM Functional Use Case "FUC2 – Straight Passing of a T-Crossing with Pedestrian Crossing")
- With a maximum allowed speed of 30 km/h, a pedestrian crossing has to be visible in Germany at a distance of at least **50 meters** (see German guideline R-FGÜ 2001 "Richtlinien für die Anlage und Ausstattung von Fußgängerüberwegen" by the German Federal Ministry of Transport, Building and Housing)

5) Requirements to achieve nominal performance

- Detection of pedestrian crossing shall take a maximum of 0.5 seconds (see SR-4.1.2f)
  - 0.5 seconds at a speed of 30 km/h is equivalent to a driven distance of **4.2 meters**.
- After detection, the pedestrian crossing shall be at least 5 seconds away (see SR-4.1.2e)
  - seconds at a speed of 30 km/h is equivalent to a driven distance of **41.7 meters**.

This sums up to a driven distance of a maximum of

```
   4.2  meters      max. distance for crosswalk detection time
+ 41.7  meters      max. distance for crosswalk reaction time
= 45.9  meters      max. driven distance before crosswalk
```

6) Requirements to achieve integrity

- Pedestrian crossing shall be detected with a deviation of a maximum of **0.3 meters** (see SR-4.1.2a)
- Ego vehicle's speed shall be detected with a deviation of a maximum of 5 percent (see SR-2.1a)
  - percent of 30 km/h equals 1.5 km/h deviation.
  - 0.5 seconds for detection of a pedestrian crossing at a speed deviation of 1.5 km/h is equivalent to an additionally driven distance of **0.2 meters**.
  - 5 seconds distance to the pedestrian crossing at a speed deviation of 1.5 km/h is equivalent to an additionally driven distance of **2.1 meters**.

This increases our driven distance calculated above to a maximum of

```
  45.9  meters      max. driven distance before crosswalk
+  0.3  meters      max. deviation for crosswalk position
+  0.2  meters      max. deviation for crosswalk detection
+  2.1  meters      max. deviation for crosswalk reaction
= 48.5  meters
```

7) Safety considerations

- We are on the safe side at a speed of 30 km/h, as the above calculated **48.5 meters** are slightly below the required distance of 50 meters in 99% of the cases.
- The safety could be further increased by the detection of traffic signs indicating pedestrian crossings (see SR-3.2.1 and SR-4.1.2.2).
- The safety could be further increased by the usage of a highly accurate road map, including all pedestrian crossings (see SR-3.7).

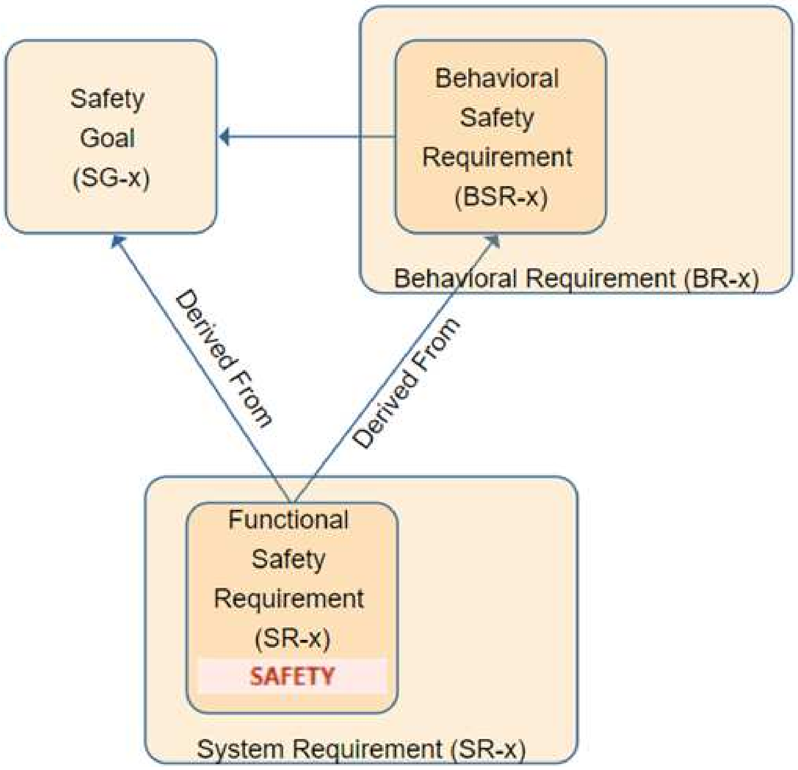


Figure 7: Deriving FSRs from SGs and BSRs in parallel

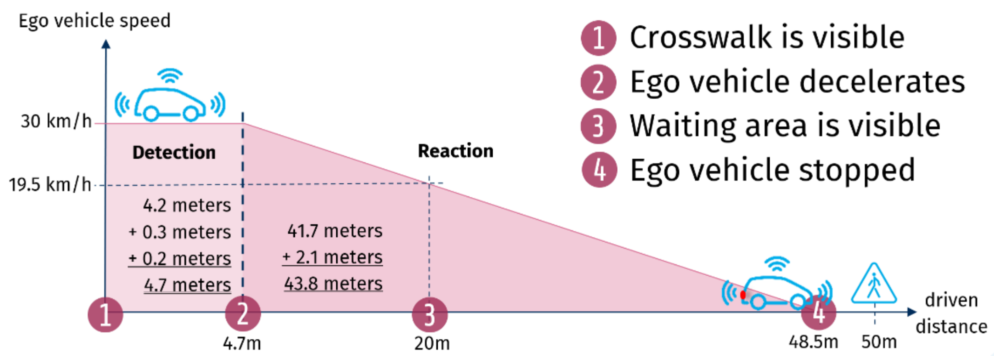


Figure 8: Vehicle slowing down when approaching crosswalk with pedestrian

A challenge is the provisioning of safety goals and behavioral safety requirements with quality measures like “fatalities per operating hour”. It is expected that these numbers are transferred into design constraints e.g., in necessary distances in meters like in the example above. This conversion of quality measures will stay a matter of experience and research.

4 Functional test specification as main input for test orchestration

As already described, the functional test specification is the central output from test planning that is transferred to test orchestration. There, it is then converted into the technical test specification with the help of the technical design. Only the technical test specification is actually executable.

4.1 Verification

Verification is intended to prove that the product meets the agreed requirements. With regard to the functional test specification, this means that it must be described in which test scenarios which functional test object must be examined with regard to which target variables. Figure 1 shows the concrete example of a functional test specification designed in VVM for verification.

After the identifier and unique designation of the functional test specification, the test scenarios are described via the next attributes. Two pieces of information are essential here: the logical scenario and the variation rule. First, the underlying functional scenario is described as well as the declared parameters including the spaces of possible parameter values and their limits. This is followed by the scenario variation information. In this simple example, with one exception, all parameter values are fixed and only the visibility is varied. The step size through this parameter space is given a constant value. It is also conceivable that variation algorithms are specified at this point. On this basis, it is subsequently possible to generate concrete scenarios during test execution.

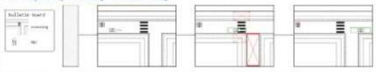
When describing the functional test object, reference is made to the functional architecture and to the system requirements. Specifically, reference is made to the functional element and its description in the architecture and specification (see attribute group 4 in Figure 1). In the example given here, the functional element (FE) is "classification of road networks and traffic guidance objects". The underlying system requirements specify the functional element and inform about the grades to be verified. For information on how these grades were arrived at, reference is made here to the GQM.

In addition to the quality to be verified, it is essential for the test planning with which confidence the verification of the quality is to be carried out. This confidence requirement comes from the safety modeling and is available to the test planning as input. Safety targets are not mentioned explicitly in the context of verification, because here the qualities are substantial, which can be led back over the system requirements to the safety targets.

The test objectives are formed from the grades to be verified and the confidence requirement. An example is given in attribute 6 in Figure 1. Implicit at this point are test termination criteria for the case that a test objective cannot be achieved due to insufficient performance.

The functional test specification ends with attribute 7, which offers the possibility of providing information on test classes and methods.

Beispiel für eine Funktionale Testbeschreibung

		Erläuterung, Anmerkung	Beispiel
1	**ID**	Eindeutige ID	FT-001
2	**Titel**	Eindeutige Bezeichnung	Erkennung Fußgängerüberweg in eigener Fahrspur voraus im Kontext FUC2.3
3	**Logisches Szenario**	Ausgangssituation, dynamische Bewegungen	FUC2 - Straight Passing of a T Crossing with Pedestrian Crossing  An urban intersection with three arms (directions west, south, east and angles 90°, 90°, 180° respectively) in a 30 km/h zone. There are no traffic signs, thus right has right of way. The arm to the west has a pedestrian crossing close to the junction. There are bicycle lanes on all sidewalks.
3	**Parameter**	deklarierte Parameter	Deklarierte Parameter • Breite der Straße (Fahrspuren, Fahrbahn): 3-12 Meter • Größe eines einzelnen Streifens: (2,5 bis 8 Meter) x (0,3 bis 0,5 Meter) • Verschmutzungsgrad und Abnutzungsgrad des Zebrastreifens: ? • Sichtweite (Niederschlag, Staub, ...): ? • Lichtverhältnisse (Blendung, ...): ? • Qualität der Ego Positionsbestimmung (GPS, ...): ?

3	**Vorgaben bzgl. Szenarienvariation**	Erzeugung der konkreten Szenarien durch:
		• Variation der statischen Parameter
		• Anfangswerte der dynamischen Parameter
		• Definition von Agentenverhalten
		Berücksichtigung Repräsentativität der Parameterabdeckung wird vorausgesetzt und nicht in der AG10 erarbeitet!
			Beispiel:
			• Variation statischer Parameter:
			  ◦ FUC2 - Straight Passing of a T Crossing with Pedestrian Crossing
			  ◦ Breite der Straße (Fahrspuren, Fahrbahn): 2 Spuren (1 Spur je Richtung), Breite 3,5 m
			  ◦ Größe eines einzelnen Zebra Streifens: 3 Meter x 0,5 Meter
			  ◦ Verschmutzungsgrad und Abnutzungsgrad des Zebrastreifens: keine Verschmutzung
			  ◦ Lichtverhältnisse (Blendung, ...): Tag, kein Gegenlicht
			  ◦ Qualität der Ego Positionsbestimmung (GPS, ...): 100%
			  ◦ Sichtweite (Niederschlag, Staub, ...): **min. 10m; max10km+; Schrittweite 1m**
			• Anfangswerte der dynamischen Parameter
			  ◦ Startposition: 200m vor Fußgängerüberweg
			  ◦ v Ego = 30 km/h an Startposition
			  ◦ a Ego = 0 m/s² an Startposition
			• Definition des ego Verhalten
			  ◦ v Ego bleibt konstant
			  ◦ a Ego bleibt 0 m/s² (+/- Toleranzwert x m/s²)
			• Definition von Agentenverhalten
			  ◦ entfällt, solange kein VRU betrachtet wird

4	**Testobjekt**	Element der Funktionalen Architektur	(FE 1.1.6.1) classification of road networks and traffic guidance objects **recognize** the classes of detected traffic signs and traffic lights, road markings, roads including shoulders, sidewalks, parking spaces etc.

4	**Anforderungen inkl. Güten**	Zu testende Anforderungen an das Testobjekt	• ~~(SR 4.1.2.1) crosswalk marking classification~~
			~~The system shall **classify** broad stripes on the road for crosswalk markings~~
			• (SR 4.1.2) crosswalk classification FUNCTIONAL (Related to G201)
			The system shall **classify** crosswalks.
			  ◦ (SR 4.1.2a) crosswalk position accuracy PERFORMANCE (Derived from G201)
			  The system shall **classify** crosswalks with a deviation to the exact position of less than 0.3 meters.
			  ◦ (SR 4.1.2b) crosswalk detection rate PERFORMANCE (Derived from G201)
			  The system shall **classify** crosswalks during free sight within 50 meters in more than 99% of the cases.
			  ◦ (SR 4.1.2c) crosswalk detection distance PERFORMANCE (Derived from G201)
			  The system shall **classify** crosswalks within a distance of at least 50 meters.
			  ◦ (SR 4.1.2d) crosswalk detection range PERFORMANCE (Derived from G201)
			  The system shall **classify** crosswalks within a distance of more than 0.3 meters and at least 50 meters.
			  ◦ (SR 4.1.2e) crosswalk reaction time PERFORMANCE (Derived from G201)
			  The system shall **classify** crosswalks at least 5 seconds before reaching it.
			  ◦ (SR 4.1.2f) crosswalk detection time PERFORMANCE (Derived from G201)
			  The system shall **classify** crosswalks within maximum 0.5 seconds.

4	**zur Information: Güten in der GQM**	rein informativ: Link zur GQM, die zu den Level 1 Güten geführt haben; entscheidend bleibt aber die Zeile oben "Systemanforderungen inkl. Güten"	G201 Recognizing a pedestrian crosswalk in the ego vehicle path:

G201 Recognizing a a pedestrian crosswalk in the ego vehicle path

Goal	Quality	Measure of Quality	Unit	Quality Criteria
G201_1.1	Genauigkeit der Positionsbestimmung eines Zebrastreifens	Abweichung von Referenzposition	Meter	Abweichung < 0,3m
G201_2.1	Erkennungsrate Zebrastreifen	Rate	1	>99% bei freier Sicht unter 50m Entfernung
G201_3.1	Minimale Distanz zur Erkennung unter Berücksichtigung Erkennungsrate	Abstand	m	Minimal geforderter Abstand zur Erkennung 50m
G201_3.2	Erkennungsfenster des Zebrastreifens	Minimaler und Maximaler Abstand unter Berücksichtigung Erkennungsrate	m	Minimaler abstand >0.3m bis >50m
G201_3.3	Minimale Zeit bis erreichen des Zebrastreifens bei Erkennung	Zeit bis erreichen des Zebrastreifens bei gegebener Geschwindigkeit zum Zeitpunkt des erkennens	s	>5s
G201_4.1	Maximale Verarbeitungszeit zur Zebrastreifenerkennung	Maximale Verarbeitungszeit zur Zebrastreifenerkennung	s	<0.5s

5	**Konfidenz**	geforderte Konfidenz für den Nachweis der Güten; kommt aus der Sicherheitsmodellierung	Beispiel: • alle Güten sind mit einer Konfidenz von 95% nachzuweisen, siehe hierzu auch das Beispiel unter Testziele
6	**Testziele**	2022 09 30 AG10 • Testziele werden aus Güten und Konfidenz gebildet	ein Beispiel: • "99 von 100 Zebrastreifen werden erkannt auf Basis von 500 Messungen"
7	**Testklassen, Testmethoden**	Abhängig von Testklasse: • Verifikation vs. Validierung • Known vs. Unknown • Cross Cutting Concerns 2022 09 30 • Cross Cutting Concerns kommen wohl erst ins Spiel bei der Realisierung und Vorgabe an Lieferanten	• Verifikation • Knowns (*Wir testen hier Szenarien, die wir kennen und erwarten*) *Diese Zeile nicht direkt relevant für die TestSpec, siehe Protokoll 7.11.22: 2022_11_07 AG10 - VVM - Confluence (vdali.de)*

Figure 9: Functional Test Specification - Example in the context of Verification

4.2 Validation

Validation pursues the goal of providing evidence of the extent to which the product fulfills its intended purpose. VVM also focuses on proving safety in this context. For this purpose, the functional test specification must describe in concrete terms what must be measured in order to gather the evidence required for the safety argumentation.

The functional test specification for validation basically follows the same structure that is used for verification. Validation must also be controlled via functional test object, functional target variables and the environment, i.e. test scenarios. However, there are differences in the content. Figure 2 shows the concrete example of a functional test specification designed in VVM for validation.

Where verification covers the known space via relevant test scenarios that can be specified in detail, validation must venture into the unknown space and find critical unknown scenarios. The functional test specification therefore requires random and versatile traversal of the ODD.

The functional test object generally moves at high system levels, i.e., as for example in Figure 2, at the level of the ADS function.

Regarding the functional target variables, we must speak of the validation criteria in the context of validation. In the present example, it is assumed that critical braking can be identified in the context of collision avoidance via a quotient of necessary deceleration to maximum possible deceleration. The functional test specification must provide information on which values are to be measured, which values must be calculated from measured values, which threshold values apply and which statistical evaluation is to be formed.

Beispiel für eine Funktionale Testbeschreibung

		Erläuterung, Anmerkung	Beispiel
1	**ID**	Eindeutige ID	FT-101
2	**Titel**	Eindeutige Bezeichnung	Einhalten unkritischer lateraler Abstände im Kontext FUC2.3
3	**Umgebung**	Ziel im Rahmen der Validierung ist es, die ODD zufällig und vielseitig zu durchfahren, um kritische Unknowns finden zu können.	
4	**Testobjekt**	ADS-Funktion (Damit ist das gesamte funktionale System gemeint.)	TP4 AG9 AD Funktionale Systemarchitektur
4	**Validierungskriterien (streggenommen sind Validierungs-Maße gemeint)**	Es muss nachgewiesen werden, welche Sicherheitsreserve im Falle einer Kollisionsvermeidung besteht. Als KPI müssen Kritikalitätsmaße herangezogen werden. Für dieses Beispiel ziehen wir folgenden KPI heran: • benötigte Verzögerung als Funktion von Geschwindigkeit und Abstand im Verhältnis zur max. möglichen Verzögerung • max. mögliche Verzögerung hängt vom Reibwert ab → wenn der Reibwert nicht mit gemessen werden kann, muss hier eine Vorgabe gemacht werden	zu messen: • Geschwindigkeit des Fahrzeugs in X-Richtung • Abstand zum Fußgänger in X-Richtung • Bezugsgröße: gefahrene Strecke, gefahrene Zeit, durchfahrene Szenarien zu berechnen: • benötigte Verzögerung, um Kollision zu vermeiden • benötigte Verzögerung / max. möglicher Verzögerung
4	**Gütekriterien**	1. Gütekriterium (z.B. Schwellwert für die Kritikalität). 2. Statistische Auswertung (inkl. Konfidenzbestimmung): Überschreitung des Gütekriteriums (z.B. Schwellwert) je km, h, Anzahl Szenarien.	1. Schwellwert liegt bei 0,3 (30% der max. möglichen Verzögerung) 2. tbd

4	**Messdaten/Messgrößen**	2023_07_17: Zeile neu eingefügt
		Für die Auswertung benötigte Informationen, zB:
		• um die Ursachen analysieren zu können, warum das Gütekriterium eines Validation Criteria verletzt wurde
			Bsp:
			• Kameramaterial/Bildaufzeichnungen
			• Daten aus dem Ego-Fahrzeug zu
			  ○ Bewegung
			  ○ Umfeld
			  ○ etc.

Figure 10: Functional Test Specification - Example in the context of Validation

4.3 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.

5 Outlook: Handover to test orchestration

As explained above, the functional test specification is developed based on the V&V concept and forms an essential input for the technical test orchestration. It is based on the functional architecture and provides information about which quality criteria must be demonstrated by means of which metrics (the functional target metrics) for each functional block (the functional test objects). The relevant test scenarios are also mentioned there. The functional test specification is not executable but provides a stable framework for planning the technical test orchestration. Only with the technical design can quality criteria and relevant metrics then be broken down to the subsystems and components. Together with variation rules that lead to concrete scenarios, the technical test specification can then be executed. Decisions on the concrete distribution of test cases to test instances are then largely determined by the capabilities and validity of the test instances. Figure A depicts the relation between Test Planning and Test Orchestration and shows the relevant Inputs.

The Test Orchestration itself is described in VVM's internal deliverable D07 and will be published in an updated version with the public deliverable D12.

6 References

(1) VVM Glossary

(2) Solingen, Rini & Berghout, Egon. (1999). *The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development.* London: McGraw-Hill.

(3) Srivastava, Ankit. (2019). *Sense-Plan-Act in Robotic Applications.* 10.13140/RG.2.2.21308.36481.