PROJECT of the PEGASUS FAMILY

VERIFICATION VALIDATION METHODS

Mid-Term Presentation 15 / 16 March 2022

An Approach for Decomposition and Analysis

Julian Pott, Ford Werke GmbH; Matthias Rauschenbach, Fraunhofer LBF;

Martin Mai, ZF; Tobias Braun, Fraunhofer IESE; Simon Kupjetz, Fraunhofer LBF;

Christian Wolschke, Fraunhofer IESE

Supported by:

Federal Ministry for Economic Affairs and Climate Action

on the basis of a decision by the German Bundestag

Content

VERIFICATION
VALIDATION
METHODS

- Part 1:

  From Capabilities to Requirements

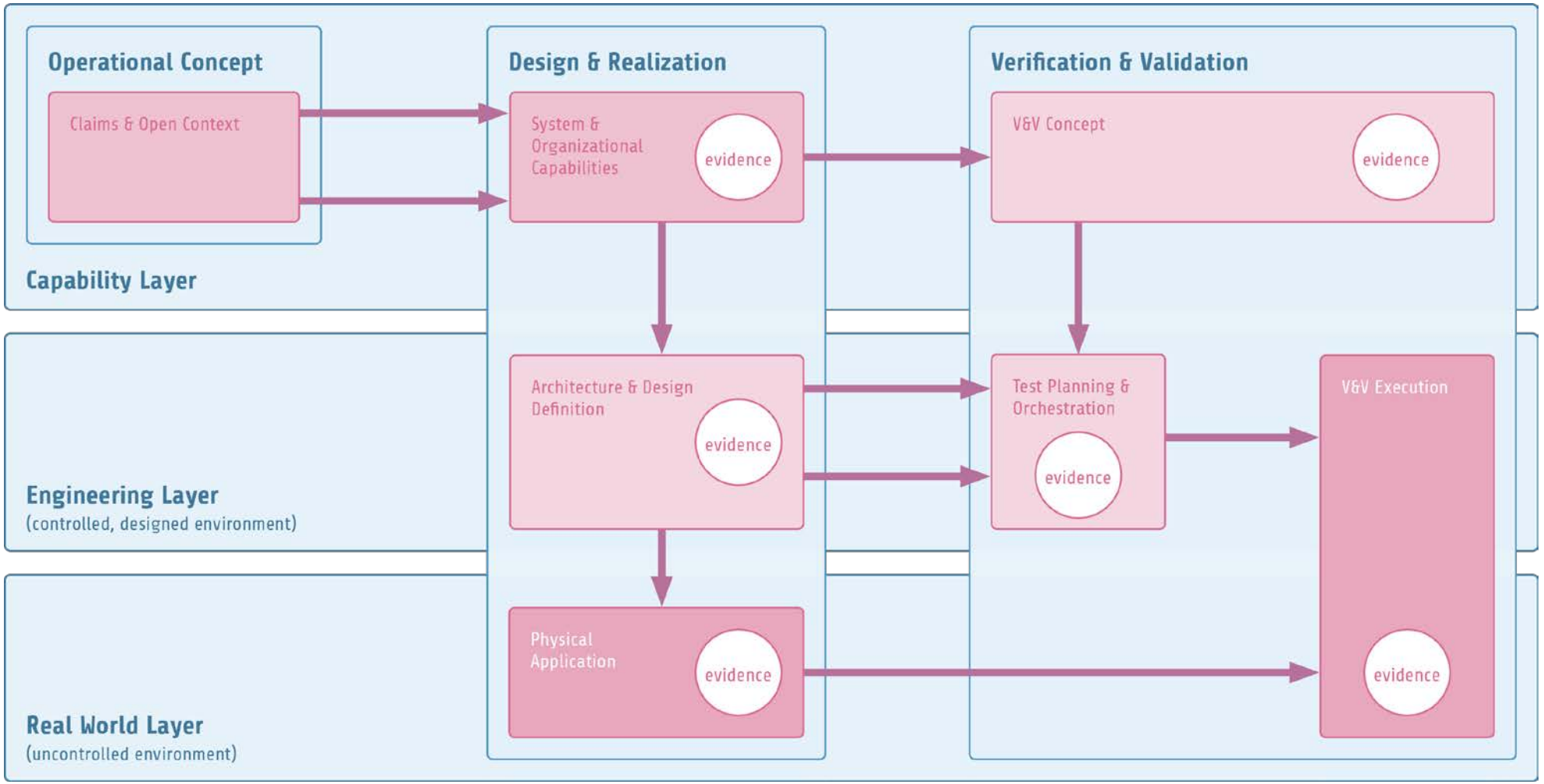
  Julian Pott, Martin Mai

- Part 2:

  A safety analysis method regarding capabilities, weaknesses and component failures

  Matthias Rauschenbach, Tobias Braun, Simon Kupjetz, Christian Wolschke

V&V Process in Assurance Framework



Problem Statement

- How to…
  - ... get from the capability-based architecture to functional requirements
    - Including other sources e.g. item definition
  - ... develop quality measurements and create a catalogue
  - ... develop additional requirements based off quality measurements
  - ... document decomposition of quality measurements
  - ... document decomposition of requirements (solved by SE)

Prerequisites and exemplary solution

- Stakeholder needs
  - Delivers description of the systems potential to produce the target behavior (Cap.-Based Arch.)
  - Functional Features
  - Item definition
- Derive System-Requirements (FUC 2.3) according to the Capability-Based Architecture

ID	Title	System Requirement	Req. Type	Refines
SR-3.1.2	crosswalk marking perception	The system shall **perceive** broad stripes on the road for *crosswalk markings*.	Functional	

- Create or review related quality measurements
  - Goal, Question, Metric (GQM) application
- Decompose requirements and quality measurements
  - Perception example

Intro to Goal, Question, Metric [Basili]

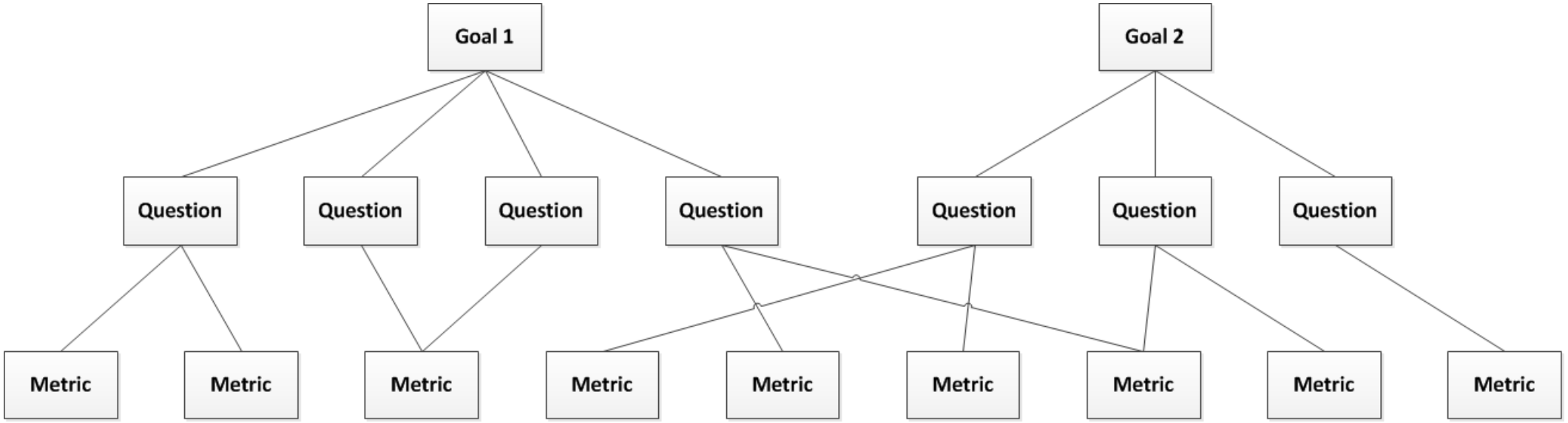
VERIFICATION
VALIDATION
METHODS

- Method to measure goals of Organizations and its Projects
- Examples:
  - Products: Specifications, Software, Designs, …
  - Processes: designing, developing, testing
  - Resources: People, Hardware, Software,..

- ➔ For our usage we only execute the first 3 steps of GQM
- Basili, Caldiera, Rombach (Encyclopedia of Software Engineering – 2 Volume Set, 1994) (Link)

GQM Model overview

- GQM model contains
  - Goals: e.g. fulfillment of the feature functions
  - Questions: Questions of stakeholders regarding a goal
  - Metrics: quantifiable answers to the questions



1. Goal Definition

- GQM goal Definition
  - Purpose
  - Issue
  - Object or Process
  - A Viewpoint (developer, tester, management, …)

- Example:

Goal	*Purpose*	Improve
	Issue	the understanding of
	Object	Goal Question Metric
	Viewpoint	from the audience‘s viewpoint.

2. Questions

- Definition of questions regarding a goal
- Represent the evaluation of success regarding a goal

Goal	*Purpose*	Improve
	Issue	the understanding of
	Object	GQM
	Viewpoint	from the audience‘s viewpoint.
Question	*Q1*	Does the web conference work?
Question	*Q2*	Is there enough time to ask questions?

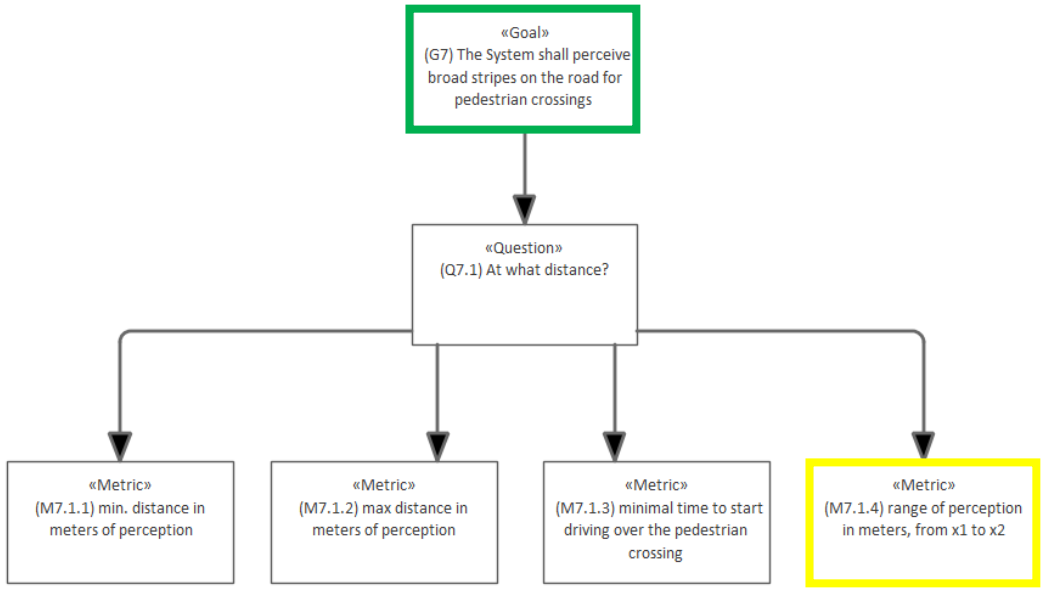
3. Metrics

- Find quantitative answers to questions

Goal	*Purpose*	Improve
	Issue	the understanding of
	Object	Goal Question Metric
	Viewpoint	from the audience‘s viewpoint.
Question	*Q1*	Does the web conference work?
Metrics	*M1*	# of disconnects per hour due to web conf sw failures
	M2	Screen forwarding latency
	M3	Packet loss of each participant
Question	*Q2*	Is there enough time to ask questions?
Metrics	*M4*	presentation time actual <= planned time for presentation

System Requirements refined by Performance Requirements

- Functional Requirements are examined
  - With respect to the goals they relate to in the GQM Model
  - Goals can be added to the GQM Model and analyzed
- New Performance Requirements are created
  - Based on GQM Model content



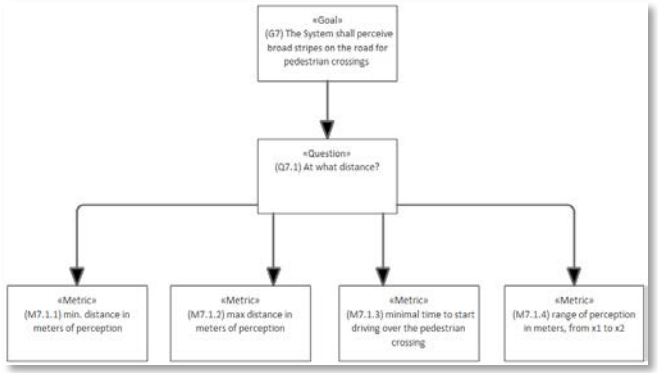
ID	Title	System Requirement	Req. Type	Refines
SR-3.1.2	crosswalk marking perception	The system shall **perceive** broad stripes on the road for *crosswalk markings*.	Functional	
SR-3.1.2a	crosswalk marking perception **range**	The system shall **perceive** crosswalk markings on the vehicle's traffic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

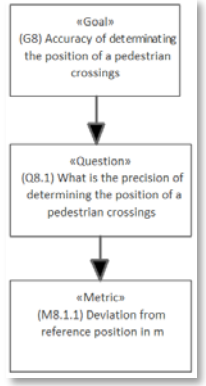
Decomposition of Requirements and Quality Measurements

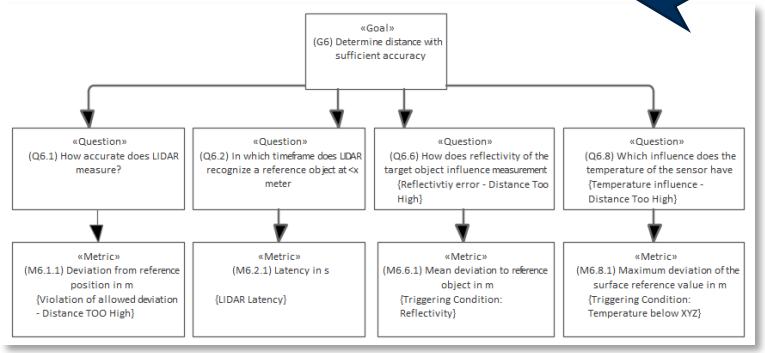
ID	Title	System Requirement	Req. Type	Refines
SR-3.1.2	crosswalk marking perception	The system shall perceive broad stripes on the road for *crosswalk markings*.	Functional	
SR-3.1.2a	crosswalk marking perception range	The system shall perceive crosswalk markings on the vehicle's traffic lane in a distance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

ID	Title	ADS Perception Requirement	Req. Type	Derived From (System Requirements)	Refines
PR-4.1.1.1	crosswalk marking object delivery	The system shall deliver an object of category *CrosswalkMarking* for crosswalk markings on the road.	Functional	SR-3.1.2 crosswalk marking perception	
PR-4.1.1.1a	crosswalk marking object delivery range	The system shall deliver objects of category *CrosswalkMarking* for crosswalk markings on the vehicle's traffic lane in a distance at least between *?* and *?* meters in the direction of driving.	Performance	**SR-3.1.2a** crosswalk marking perception range	PR-4.1.1.1
PR-4.1.1.1b	crosswalk marking position accuracy	The system shall deliver objects of category *CrosswalkMarking* for crosswalk markings on the vehicle's traffic lane with a maximum deviation from the ref position in ? m	Performance		PR-4.1.1.1

ID	Title	Component Requirement	Req. Type	Derived From (ADS P Req)	Refines
CR	**…**	*(work in progress)*			







*Decomposition included in GQM Model but not shown

Conclusion and Outlook

- Based on a Capability-based architecture and first functional System Requirements
  - How to derive Qualitiy Measurements via GQM and catalogue them
  - Document discovered decomposition of qualitiy measurements via the GQM Model
  - Established a knowledge base for Requirements and Test Requirements

- CFT, probFMEA as consumer of GQM Model to perform Safety Analysis

Content

- Part 1:

  From Capabilities to Requirements

  Julian Pott, Martin Mai

- Part 2:

  A safety analysis method regarding capabilities, weaknesses and component failures

  Matthias Rauschenbach, Tobias Braun, Simon Kupjetz, Christian Wolschke

New Challenges for highly automated driving

VERIFICATION
VALIDATION
METHODS

How to…

- **… perform a methodological safety analysis** considering
  - Fully automated driving function
  - Very large number of variations of driving scenarios and boundary conditions (open context)
  - Large range of possible interactions of the vehicle with its environment

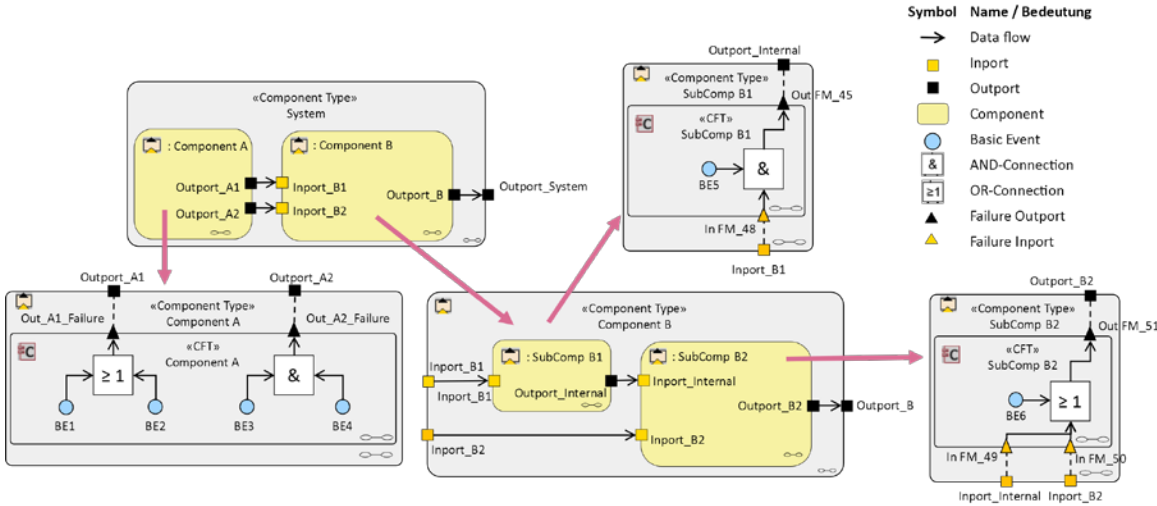
- **… identify gaps and shortcomings of the implementation** on each level of aggregation
  - Behavioral layer
  - Engineering layer

- **… evaluate and measure sufficiency of risk mitigation** and safety measures

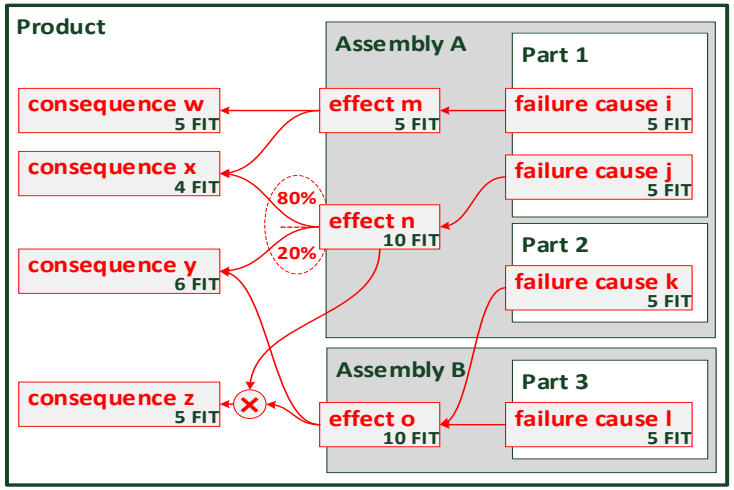
- **… contribute to the specification of testing criteria and strategies**

Combination of advanced state-of-the-art methodical concepts for safety verification

- **Component Fault Trees “CFT” (Fraunhofer IESE)**
  - Top-Down approach:
    Safety goal violation → Causes in parts
  - Modular, hierarchic approach
  - Modelling of failure propagation
  - Tight alignment and traceability between failure propagation and architecture



- **Probabilistic FMEA “probFMEA“ (Fraunhofer LBF)**
  - Bottom-Up approach:
    causes in parts → Safety goal violation
  - Logical network with multiple failure effects (conditional)
  - Modelling and calculation of consequential probabilities in Bayesian Networks

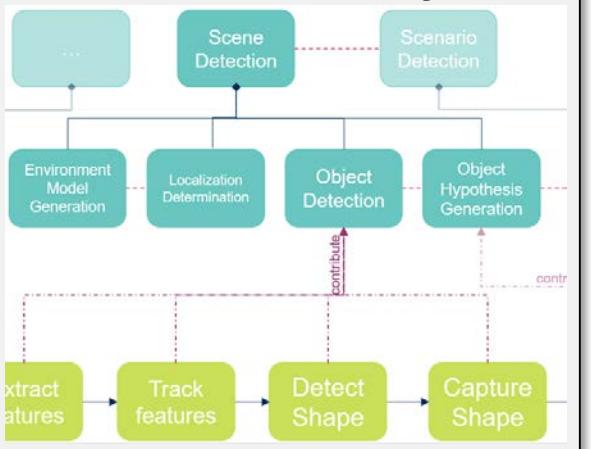


- **Combination of both approaches to establish**
  - one coherent and systematic methodology
  - one consistent and wholistic information model

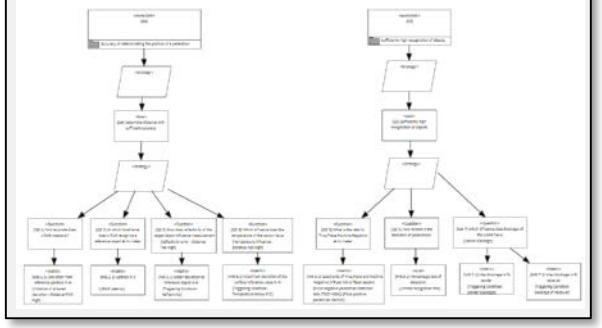
Approach related to levels of specifications of a HAV

VERIFICATION VALIDATION METHODS

System capability-based taxonomy



GQM



Input → Analysis

Capability Layer

Functional Use-Case

Design & Realization

Goals

ODD Def.

System Capabilities

System Capa-bilities

Functional Features

Failure Analysis CFT + probFMEA — unintended behaviour ?

Failure Analysis CFT + probFMEA — insufficient capability ?

Engineering Layer

Functional Architecture & Design Definition

Functional Design

Technical Architecture & Design Definition

Technical Design

Failure Analysis CFT + probFMEA — weakness ?

Failure Analysis CFT + probFMEA — system- and component failures ?

Analysis steps across the different development views

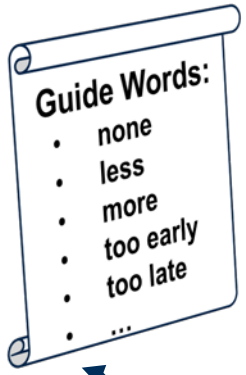
VERIFICATION VALIDATION METHODS

Behavioral layer

Vehicle behavior in the specific scenario (ego-perspective)	Behavioral deviations and behavioral errors
System capability-based taxonomy	Capability deficits, incapabilities

Engineering layer

System architecture	Weaknesses and deviations from intended function
specific components	Defects and failure of individual elements

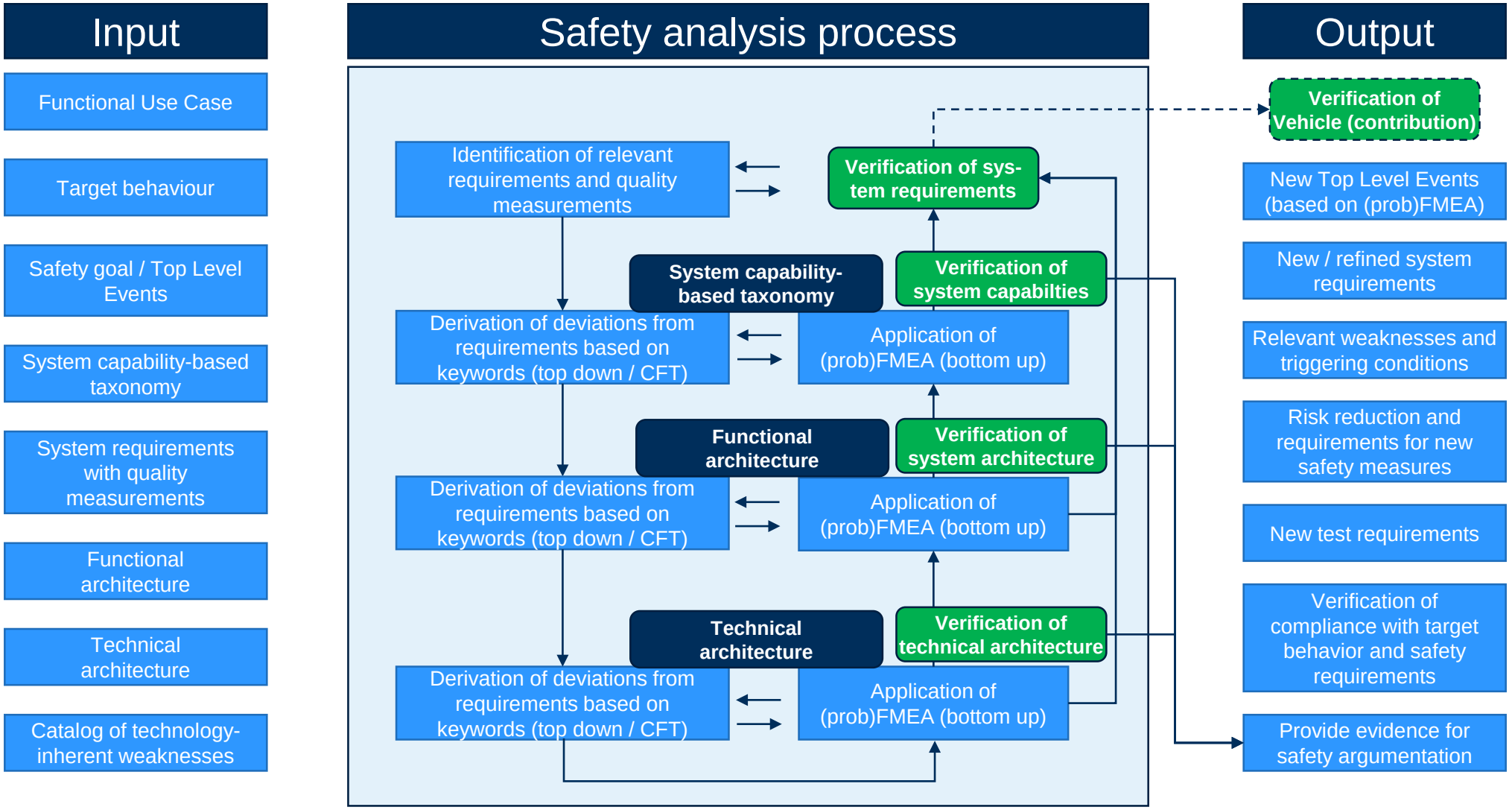


Possible causes for assumed deviation

Possible consequences of assumed deviaton

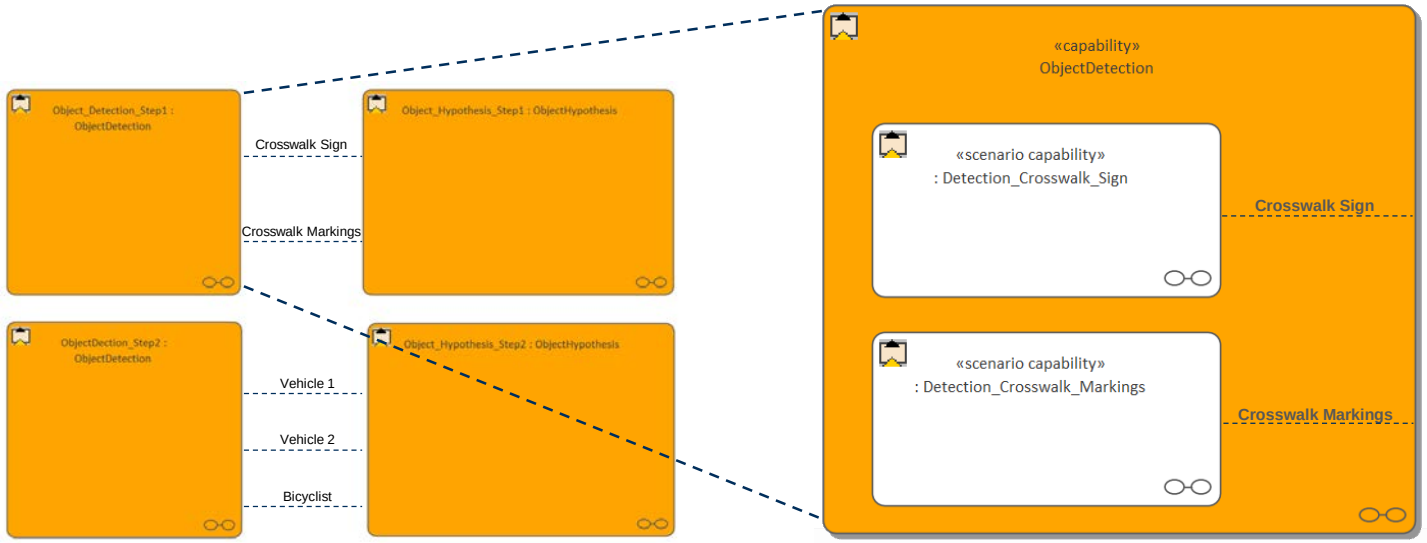
which capability deficits and deviations can lead to certain behavioral deviations?	which undesired behavior and effects can possible capability deficits lead to in the specific scenario?
which functional weaknesses and deviations can lead to specific capability deficits?	which capability deficits do functional failures and weak-nesses lead to?
Which failure in parts and data processing errors could occur to cause the assumed system failure?	What functional deviations can possible component and software failures lead to?

Process overview



Exemplary demonstration of tool-supported method application

- Instantiated model the capability-based architecture for one specific use-case (as a prerequisite)
  [for more background on this, see presentation on capability-based taxonomy (by T. Hofmann Stream 1)]
- Exemplary representation in Enterprise Architect (EA)
- Tool support implemented in **SafeTBox** (EA plug-in)



Guide-word-based analysis

Generic Failure Type System Specification

Name	Description
NONE	Indicates that no failure types is defined
No	Service or Signal is not delivered
Less	Service or Signal provides a lower value than expected
More	Service or Signal provides a higher value than expected
Too early	Service or Signal is delivered earlier than expected
Too late	Service or Signal is delivered later than expected
Non existent	Elemente refered to does not exist
Too large	Service or Signal provides too large values resp results
Too small	Service or Signal provides too small values resp results

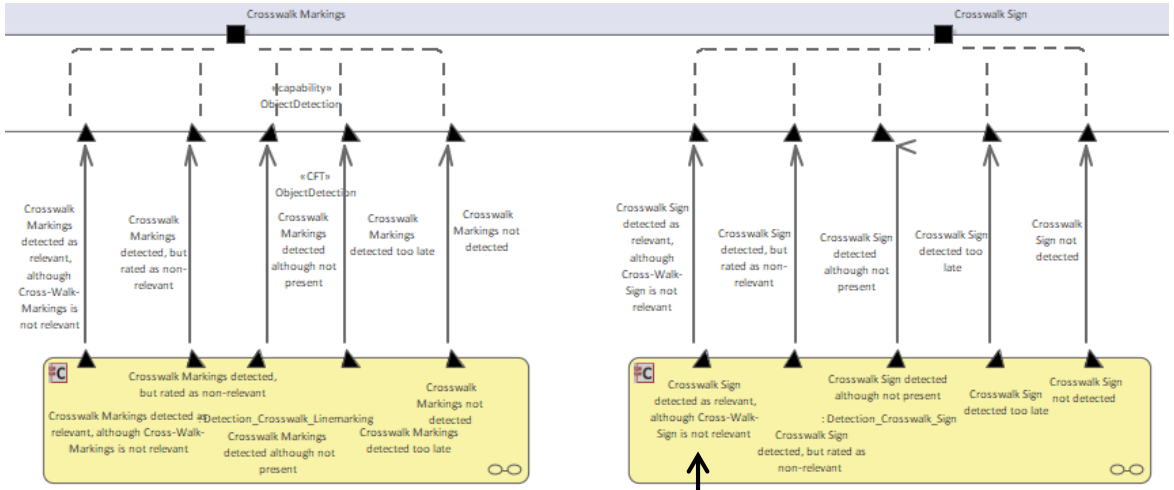
Possibilities for insufficency of the capability

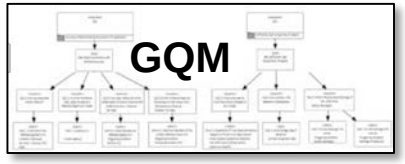
„Detection of crosswalk sign“ (Use Case Step 1):

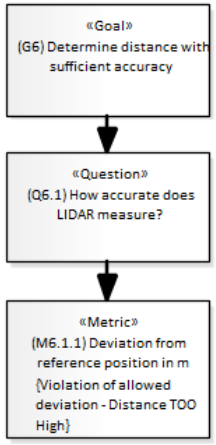
- Crosswalk sign not identified
- Crosswalk sign identified too late
- other sign identified instead
- Crosswalk sign identified in wrong position

…

Exemplary failure model for „crosswalk sign detected as relevant, although not relevant“

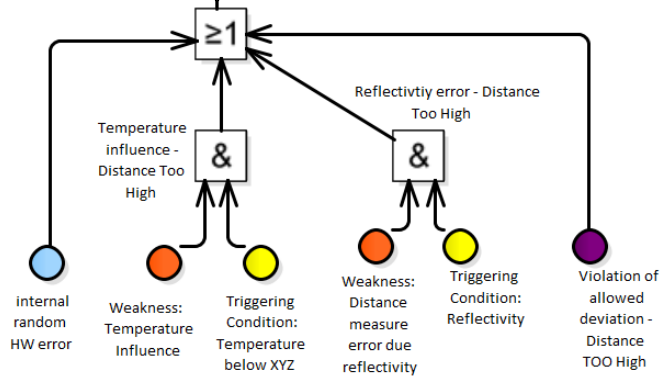


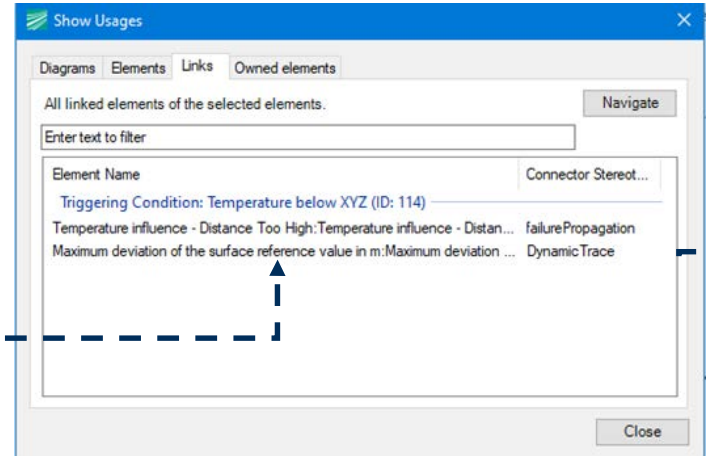




Tracing back to the specified quality measurements

For further details and explanations, please also refer to the dedicated poster





Summary and outlook

- Advanced methodology for analytical verification of highly automated driving
  - Integral approach combining fault trees and FMEA (reduction of effort and inconsistencies)
  - Scenario-based verification of behavior against situationally required capabilities
  - Analysis of component weaknesses (SOTIF) and failures (functional safety)

- Coherent evaluation in a consistent database:
  - Qualitative: Determination of relevant triggering conditions and weaknesses (minimal cut-sets)
  - Quantitative: Consistent evaluation of safety-criteria for automated vehicles (scenario-specific)

- Future research topics and challenges for the 2nd half of the VVM-project:
  - Usage of the GQM and failure analysis methodologies in the framework of a consistent safety argumentation
  - Refinement, verification of the formalism for notation and modeling of failure model across the boundary between capability and engineering layer
  - Requirements concerning the tool support to handle the complexity of HAD

VERIFICATION
VALIDATION
METHODS

Thank you!

Julian Pott, Ford Werke GmbH; Matthias Rauschenbach, Fraunhofer LBF;

Martin Mai, ZF; Tobias Braun, Fraunhofer IESE; Simon Kupjetz, Fraunhofer LBF;

Christian Wolschke, Fraunhofer IESE

PROJECT of the PEGASUS FAMILY

A project developed by the
VDA Leitinitiative
autonomous and connected driving

Supported by:

Federal Ministry
for Economic Affairs
and Climate Action

on the basis of a decision
by the German Bundestag