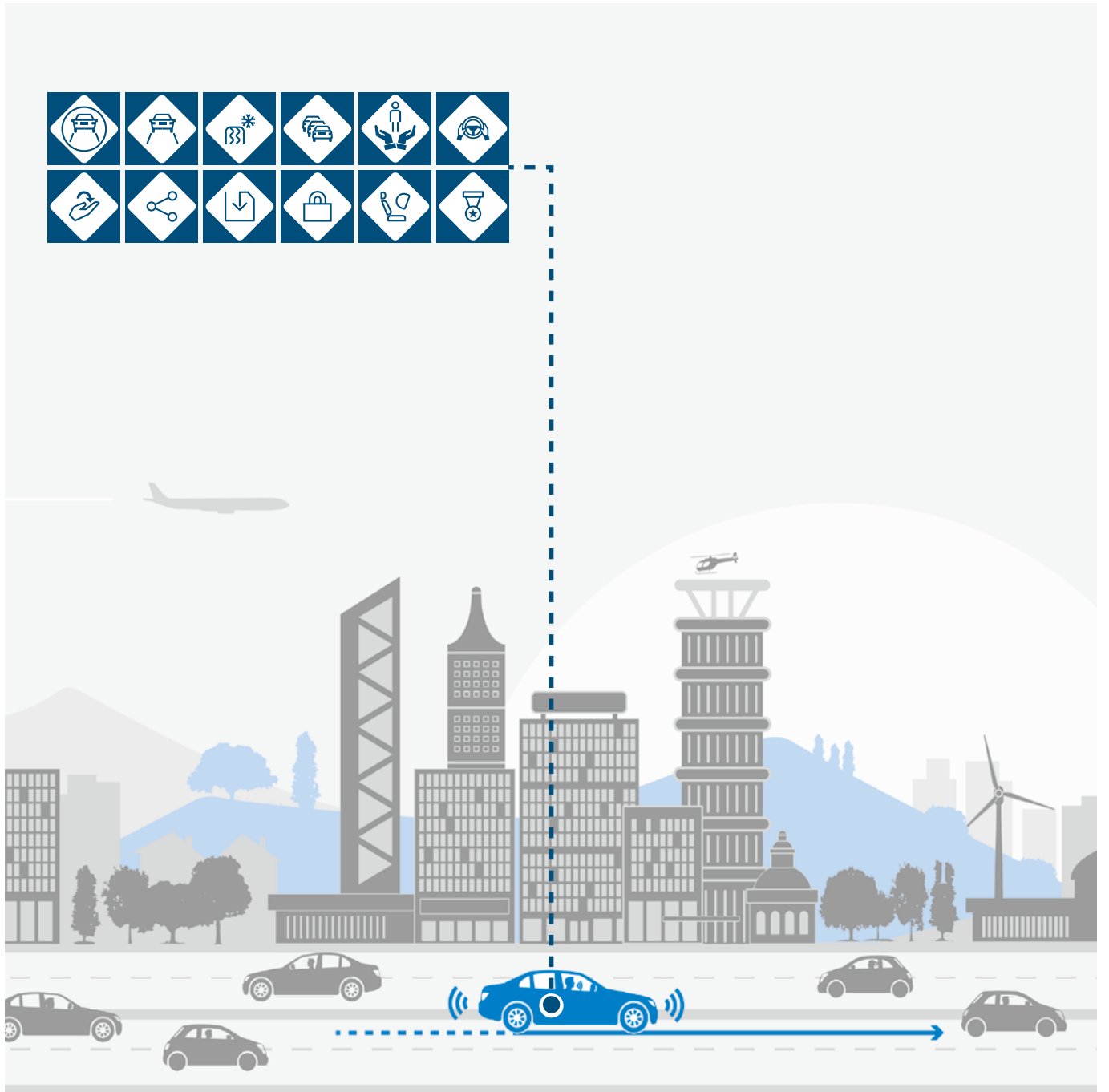


SAFETY FIRST FOR AUTOMATED DRIVING



AUTHORS

APTIV

Matthew Wood, M.Sc.

matthew.wood@aptiv.com

Dr. Philipp Robbel

philipp.robbel@aptiv.com

Dr. Michael Maass

Dr. Radboud Duintjer Tebbens

Marc Meijs, M.Sc.

Mohamed Harb, M.Sc.

Jonathon Reach, B.Sc.

Karl Robinson

AUDI

David Wittmann, M.Sc.

david.wittmann@audi.de

Toshika Srivastava, M.Sc.

Dr.-Ing. Mohamed Essayed

Bouzouraa

BAIDU

Siyuan Liu, BS, MBA

liusiyuan01@baidu.com

Yali Wang, MA

wangyali05@baidu.com

BMW

Dr.-Ing. Christian Knobel

christian.knobel@bmw.de

Dipl.-Inf. David Boymanns

david.boymanns@bmw.de

Dr.-Ing. Matthias Löhning

Dr. Bernhard Dehlink

Dirk Kaule, M.Sc.

Dipl.-Ing. Richard Krüger

Dr. Jelena Frtunikj

Dr. Florian Raisch

Dipl.-Math. Miriam Gruber

Jessica Steck, M.Sc.

Dipl.-Psych. Julia Mejia-Hernandez

CONTINENTAL

Dipl.-Ing. Sandro Syguda

sandro.syguda@continental-corporation.com

Dipl.-Ing. Pierre Blüher

Dr.-Ing. Kamil Klonecki

Dr. Pierre Schnarz

DAIMLER

Dr. Thomas Wiltshcko

thomas.t.wiltshcko@daimler.com

Dipl.-Inf. Stefan Pukallus

Dr.-Ing. Kai Sedlaczek

FCA

Neil Garbacik, M.Sc.

neil.garbacik@fcagroup.com

David Smerza, BSAE

Dr. Dalong Li

Dr. Adam Timmons

Marco Bellotti

HERE

Michael O'Brien, BS

michael.obrien@here.com

Michael Schöllhorn

INFINEON

Dipl.-Ing. Udo Dannebaum

udo.dannebaum@infineon.com

INTEL

Jack Weast, BS, M.Sc.

jack.weast@intel.com

Alan Tatourian, BS

VOLKSWAGEN

Dr.-Ing. Bernd Dornieden

bernd.dornieden@volkswagen.de

Dr.-Ing. Philipp Schnetter

Dr.-Ing. Dipl.-Wirt.Ing. Philipp

Themann

Dr.-Ing. Thomas Weidner

Dr. rer. nat. Peter Schlicht

ABSTRACT

This publication summarizes widely known safety by design and verification and validation (V&V) methods of SAE L3 and L4 automated driving. This summary is required for maximizing the evidence of a positive risk balance of automated driving solutions compared to the average human driving performance. There is already a vast array of publications focusing on only specific subtopics of automated driving. In contrast, this publication is a comprehensive approach to safety relevant topics of automated driving and is based on the input of OEMs, tiered suppliers and key technology providers. The approach of this publication is to systematically break down safety principles into safety by design capabilities, elements and architectures and then to summarize the V&V methods in order to demonstrate the positive risk balance. This publication is intended to contribute to current activities working towards the industry-wide standardization of automated driving.

REFERENCED STANDARDS

ISO/PAS 21448:2019	Road Vehicles – Safety of the intended functionality (SOTIF)
ISO 26262:2018	Road Vehicles – Functional safety
ISO/SAE CD 21434	Road Vehicles – Cybersecurity engineering
ISO 19157:2013	Geographic information – Data quality
ISO/TS 19158:2012	Geographic information – Quality assurance of data supply
ISO/TS 16949:2009	Quality management systems – Particular requirements for the application of ISO 9001:2008 for automotive production and relevant service part organizations
ISO/IEC 2382-1:1993	Information technology – Vocabulary – Part 1: Fundamental terms
ISO/IEC/IEEE 15288:2015	Systems and software engineering – System life cycle processes

© Copyright 2019 by Aptiv Services US, LLC; AUDI AG; Bayerische Motoren Werke AG; Beijing Baidu Netcom Science Technology Co., Ltd; Continental Teves AG & Co oHG; Daimler AG; FCA US LLC; HERE Global B.V.; Infineon Technologies AG; Intel; Volkswagen AG. All rights reserved.

The document and information contained herein is not a license, either expressly or impliedly, to any intellectual property owned or controlled by any of the authors or developers of this publication, and license to this document and information should not be considered to be have been made available to parties receiving and/or reviewing this document and information. The information contained herein is provided on an "AS IS" basis, and to the maximum extent permitted by applicable law, the authors and developers of this document hereby disclaim all other warranties and conditions, either express, implied or statutory, including but not limited to, any (if any) implied warranties, duties or conditions of merchantability, of fitness for a particular purpose, of accuracy or completeness of responses, of results, of workmanlike effort, of lack of viruses, of lack of negligence. THERE IS NO WARRANTY OR CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, OR NON-INFRINGEMENT.

Contents

1	INTRODUCTION & MOTIVATION	2
1.1	Scope of this Publication	2
1.2	Structure of and Development Examples Used in this Publication	4
1.3	Safety Vision	6
1.3.1	Background	6
1.3.2	The Twelve Principles of Automated Driving	6
2	SYSTEMATICALLY DEVELOPING DEPENDABILITY TO SUPPORT SAFETY BY DESIGN	12
2.1	Deriving Capabilities of Automated Driving from Dependability Domains	13
2.1.1	Legal Frameworks for Automated Driving Vehicles	13
2.1.2	Applying the Related Safety Standards	14
2.1.3	Safety of the Intended Functionality	17
2.1.4	Functional Safety	20
2.1.5	Automotive Cybersecurity	21
2.1.5.1	Why is Cybersecurity so Important for Safety?	22
2.1.5.2	Cybersecurity Approach and Measures	24
2.1.6	Capabilities of Automated Driving	27
2.1.6.1	Initial Derivation of Capabilities	27
2.1.6.2	Overview of the Capabilities	30
2.1.7	Minimal Risk Conditions and Minimal Risk Maneuvers	34
2.2	Elements for Implementing the Capabilities	36
2.2.1	Implementing the Capabilities	36
2.2.1.1	FS_1: Determine location	37
2.2.1.2	FS_2: Perceive relevant static and dynamic objects in proximity to the automated vehicle	38
2.2.1.3	FS_3: Predict the future behavior of relevant objects	39
2.2.1.4	FS_4: Create a collision-free and lawful driving plan	40
2.2.1.5	FS_5: Correctly execute and actuate the driving plan	41
2.2.1.6	FS_6: Communicate and interact with other (vulnerable) road users	41
2.2.1.7	FS_7: Determine if specified nominal performance is not achieved	42
2.2.1.8	FD_1: Ensure controllability for the vehicle operator	43
2.2.1.9	FD_2: Detect when degraded performance is not available	44
2.2.1.10	FD_3: Ensure safe mode transitions and awareness	44
2.2.1.11	FD_4: React to insufficient nominal performance and other failures via degradation	45

2.2.1.12	FD_5: Reduce system performance in the presence of failure for the degraded mode.....	46
2.2.1.13	FD_6: Perform degraded mode within reduced system constraints	46
2.2.2	Elements	47
2.2.2.1	Environment Perception Sensors	47
2.2.2.2	A-Priori Perception Sensors	48
2.2.2.3	V2X	51
2.2.2.4	Sensor Fusion	51
2.2.2.5	Interpretation and Prediction	52
2.2.2.6	Localization	53
2.2.2.7	ADS Mode Manager	53
2.2.2.8	Egomotion	54
2.2.2.9	Drive Planning	55
2.2.2.10	Traffic Rules.....	56
2.2.2.11	Motion Control	56
2.2.2.12	Motion Actuators	57
2.2.2.13	Body Control with Secondary Actuators	58
2.2.2.14	Human-Machine Interaction	58
2.2.2.15	User State Determination.....	61
2.2.2.16	Vehicle State	64
2.2.2.17	Monitors (Nominal and Degraded Modes).....	64
2.2.2.18	Processing Unit	64
2.2.2.19	Power supply	65
2.2.2.20	Communication Network.....	65
2.3	Generic Logical Architecture.....	65
3	VERIFICATION AND VALIDATION	72
3.1	The Scope and Main Steps of V&V for Automated Driving Systems	72
3.2	Key Challenges for V&V of L3 and L4 Systems	75
3.3	V&V Approach for Automated Driving Systems.....	76
3.3.1	Defining Test Goals & Objectives (Why & How Well).....	77
3.3.2	Test Design Techniques (How).....	77
3.3.3	Test Platforms (Where)	78
3.3.4	Test Strategies in Response to the Key Challenges	79
3.4	Quantity and Quality of Testing	83
3.4.1	Equivalence Classes and Scenario-Based Testing	84
3.5	Simulation	85
3.5.1	Types of Simulation	87
3.5.2	Simulation Scenario Generation	88

3.5.3	Validating Simulation	89
3.5.4	Further Topics in Simulation.....	89
3.6	V&V of Elements	90
3.6.1	A-Priori Information and Perception (Map).....	91
3.6.2	Localization (Including GNSS)	92
3.6.3	Environment Perception Sensors, V2X and Sensor Fusion	92
3.6.4	Interpretation and Prediction, Drive Planning and Traffic Rules.....	93
3.6.5	Motion Control	93
3.6.6	Monitor, ADS Mode Manager (Including the Vehicle State)	93
3.6.7	Human-Machine Interaction	94
3.7	Field Operation (Monitoring, Configuration, Updates).....	94
3.7.1	Testing Traceability	94
3.7.2	Robust Configuration and Change Management Process	95
3.7.3	Regression Prevention	95
3.7.4	Security Monitoring and Updates	96
3.7.5	Continuous Monitoring and Corrective Enforcement.	97
4	CONCLUSION AND OUTLOOK	100
5	APPENDIX A: DEVELOPMENT EXAMPLES	104
5.1	L3 Traffic Jam Pilot (TJP)	104
5.1.1	Nominal Function Definition	104
5.1.2	Minimal Risk Conditions	104
5.1.3	Minimal Risk Maneuver.....	104
5.2	L3 Highway Pilot (HWP).....	104
5.2.1	Nominal Function Definition	104
5.2.2	Degraded Mode/Minimal Risk Conditions	104
5.2.3	Minimal Risk Maneuvers	104
5.3	L4 Urban Pilot (UP)	104
5.3.1	Nominal Function Definition	105
5.3.2	Degraded Mode/Minimal Risk Conditions	105
5.3.3	Minimal Risk Maneuvers.....	105
5.4	L4 Car Park Pilot (CPP)	105
5.4.1	Nominal Function Definition	105
5.4.2	Degraded Mode/Minimal Risk Conditions	105
5.4.3	Minimal Risk Maneuver.....	105
5.5	Selection of the Discussed Elements.....	107
5.5.1	Sensing Elements for FS_1 Localization.....	107
5.5.2	Sensing Elements for FS_2 Perceive Relevant Objects	108

5.5.3	Interpretation and Prediction in FS_3 Predict Future Movements.....	109
5.5.4	Acting Elements in FS_5 Execute Driving Plan and FD_6 Perform Degraded Mode.....	110
5.5.5	ADS Mode Manager in FS_7 Detect Nominal Performance and FD_4 React to Insufficient Performance	111
5.5.6	User State Determination in FD_1 Ensure Controllability for Operator	112
5.5.7	HMI in FD_1 Ensure Controllability for Operator and FD_6 Perform Degraded Mode.....	113
5.5.8	Monitors in FS_7 and FD_2.....	113
6	APPENDIX B: USING DEEP NEURAL NETWORKS TO IMPLEMENT SAFETY-RELATED ELEMENTS FOR AUTOMATED DRIVING SYSTEMS	116
6.1	Motivation and Introduction: Machine Learning in Automated Driving	116
6.2	Define (What and Why).....	118
6.3	Specify (How).....	120
6.3.1	Defining and Selecting the Data.....	120
6.3.2	Architecture Design for DNNs.....	123
6.4	Develop and Evaluate	125
6.5	Deploy and Monitor	128
6.6	DNN Safety Artifacts	130
7	GLOSSARY.....	134
8	REFERENCES.....	142

List of Abbreviations

ADAS	Advanced Driver Assistance System
ADS	Automated Driving System
ASIL	Automotive Safety Integrity Level
AUTO-ISAC	Automotive Information Sharing and Analysis Center
AUTOSAR	AUTomotive Open System Architecture
CERTS	Computer Emergency Response Team
CPU	Central Processing Unit
CPP	Car Park Pilot
CRC	Cyclic Redundancy Check
DDT	Dynamic Driving Task
DESTATIS	(Statistisches Bundesamt) Federal Statistical Office of Germany
DFMEA	Design Failure Mode and Effect Analysis
DiL	Driver-in-the-Loop
DNN	Deep Neural Network
E/E	Electrical/Electronic
ECU	Electronic Control Unit
EPS	Electric Power Steering
EU	European Union
FMEA	Failure Mode and Effects Analysis
FMVSS	Federal Motor Vehicle Safety Standards
FUSA	Functional Safety
GDPR	European General Data Protection Regulation
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GPU	Graphics Processing Unit
HiL	Hardware-in-the-Closed-Loop
HMI	Human-Machine Interaction
HW	Hardware
HW REPRO.	Hardware Reprocessing
HWP	Highway Pilot
I/O Port	Input/Output Port
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IMU	Inertial Measurement Unit
IPsec	Internet Protocol Security
ISO	International Organization for Standardization
ISTQB	International Software Testing Qualifications Board
LIDAR	Light Detection and Ranging
MCU	Microcontroller Unit
MRC	Minimal Risk Condition

MRM	Minimal risk maneuver
NDS	Naturalistic Driving Study
NHTSA	National Highway Traffic Safety Administration
NTSB	National Transportation Safety Board
ODD	Operational Design Domain
OEM	Original Equipment Manufacturer
OR	Open Road
OTP	One True Pairing
OUT	Object Under Test
PG	Proving Ground
RAMSS	Reliability, Availability, Maintainability, Safety and Security
RMA	Reliable Map Attribute
SDL	Secure Development Lifecycle
SiL	Simulation-in-the-Closed-Loop
SoC	System on Chip
SOTIF	Safety of the Intended Functionality
STVG	(Straßenverkehrsgesetz) German Road Traffic Act
SW	Software
SW REPRO.	Software Reprocessing
TJP	Traffic Jam Pilot
TLS	Transport Layer Security
UNECE	United Nations Economic Commission for Europe
UP	Urban Pilot
V&V	Verification and Validation

Chapter

01

INTRODUCTION & MOTIVATION

1 Introduction & Motivation

Automated driving is one of the key modern technologies. In addition to offering broader access to mobility, it can also help to reduce the number of driving-related accidents and crashes. When doing so, the safety of automated driving vehicles is one of the most important factors. Designed to supplement existing publications on various aspects of safety, this publication presents a more technical-based overview of the requirements during development to avoid safety-related hazards and thus emphasizing the importance of safety by design. Furthermore, this publication aims to provide a sound discussion of the verification and validation of such systems, which is currently still missing from existing literature.

This publication is intended to contribute to current activities working towards the industry-wide standardization of automated driving. This effort will also contribute toward a deeper understanding by developing a framework or guideline for the safety of automated driving systems for all companies in the automotive and mobility world – from technology startups through to established OEMs and the tiered suppliers of key technologies.

1.1 Scope of this Publication

The goal of this publication is to provide an overview of the generic steps for developing and validating a safe automated driving system. The starting point for this is the defining of guidelines or principles taken from different regulatory publications (various legal frameworks from around the world, ethics reports, etc.). These principles are the foundation of this publication, forming the basis from which the safety by design, methods and V&V strategies are derived. The constant focus hereby is on the development that is required in addition to existing SAE L1 and L2 driver assistance systems. It is important to consider security in conjunction with safety, as security is concerned with active adversaries whereas safety deals with passive adversaries. This difference warrants the use of additional analysis tools and technical mechanisms that in turn impact safety. Thus, safety and security should work together, and so Section 2.1.4 explores this concept in greater detail.

This publication further aims to develop a generic baseline that tackles the risks introduced by automated vehicles. This approach comprises a common consensus of the contributing companies and must be expanded upon for every specific system introduced onto the market. This publication was written based on the state-of-the-art automated driving technology at the time of publication. As such, it is not a complete work and its contents should be continuously revisited and revised whenever advances are made in the areas of social acceptance, technology and legislation.

Devising an explicit technical solution is not included in the scope of this publication, as several possibilities exist regarding the definition of the automated driving system, its operational design domain and technical advancements, etc. Due to its focus on safety, this publication does not address topics such as unsupervised machine learning, misuse, data privacy or advanced driver assistance system either. Finally, non-safety-relevant elements that are normally part of a customer function, such as a comfortable driving strategy or the fastest navigation from point to point, are also not included in the scope of this publication.

The intended audience of this publication includes entities such as the media and press, regulators, individuals from the automated driving industry, insurance companies and any persons involved in later standardization efforts.

For the avoidance of doubt, this publication is not intended and shall not be construed to establish, create, or deem to have created a maximum requirement regarding dependability for automated driving, nor shall it prevent, hinder, or restrict a Party or an Affiliated Company of a Party to deviate in any way from the content of this Publication. In case any statement of this Publication may be perceived as conflicting to this vision, the non-binding vision shall always prevail.

1.2 Structure of and Development Examples Used in this Publication

This publication is structured as interconnected topics which build upon one another to achieve an overall safety vision. Figure 1 visualizes this structure:

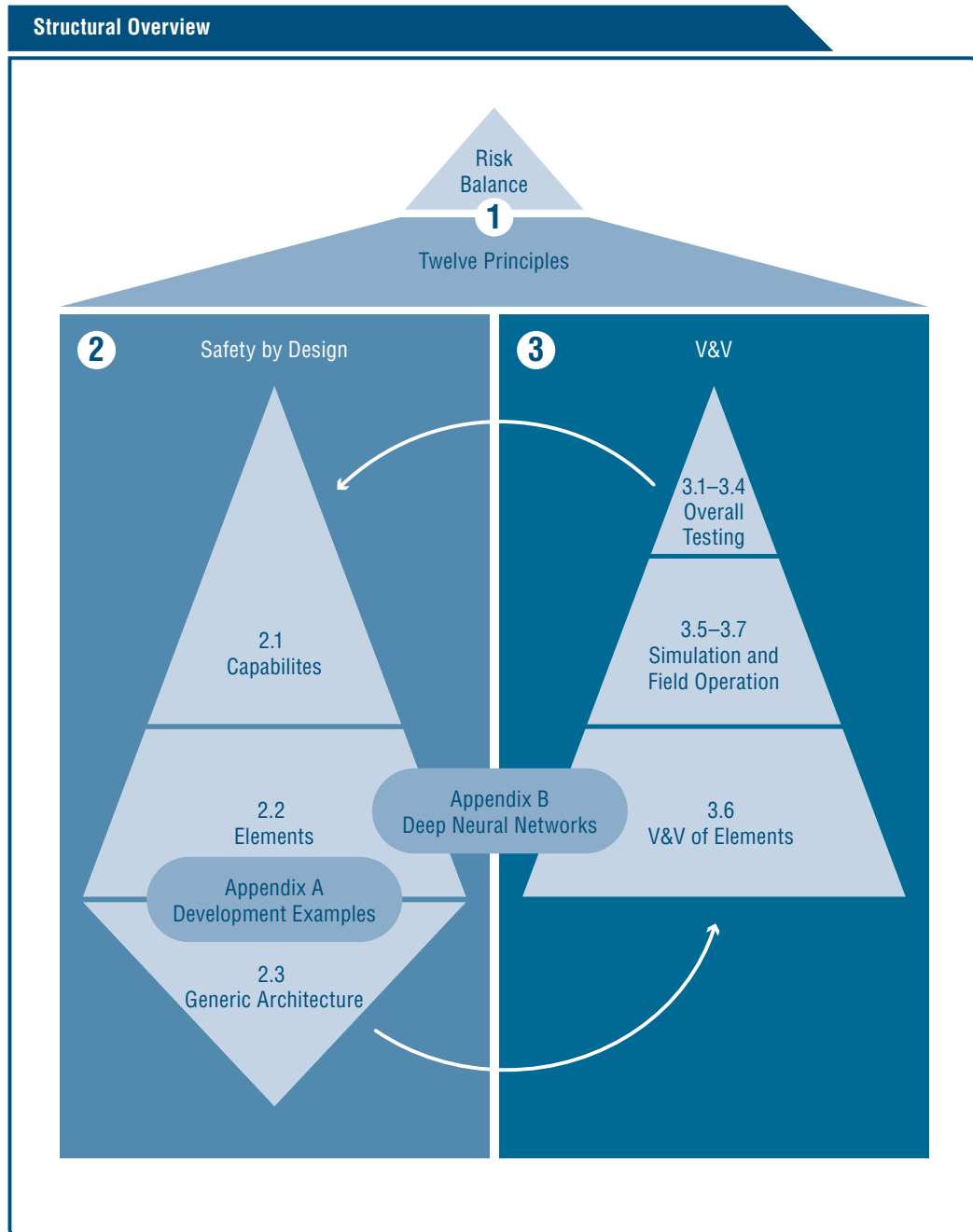


Figure 1: Structural Overview

The roof ridge in the figure represents the positive risk balance as an initial starting point and the overall goal. The roof directly underneath represents the twelve principles for automated driving. Together, the roof ridge and roof represent Chapter 1. This overall roof structure is supported by the two pillars, safety by design (Chapter 2) and verification and validation (Chapter 3). The first pillar introduces the three domains of dependability for automated driving: Safety of the intended functionality (Section 2.1.3), functional safety (Section 2.1.4) and automotive cybersecurity (Section 2.1.5). Capabilities for automated driving are then derived out of the twelve principles and the three dependability domains. Section 2.2 introduces elements to implement the capabilities, and a generic architecture connecting the elements together is introduced in Section 2.3. The architecture of the development examples (Appendix A) forms the last element of this pillar.

The second pillar begins by introducing the approach in Section 3.1 to Section 3.3 before discussing the quantity of testing (Section 3.4) and simulation (Section 3.5). Section 3.6 presents the verification and validation of the elements introduced in Section 2.2. The final block of the second pillar comprises the discussion relating to field operation in Section 3.7. Both pillars are then linked together via the V&V approach outlined throughout Chapter 3, which combines safety by design and testing with the main strategies applied in V&V to solve the challenges discussed throughout this publication. Finally, Appendix B discusses the use of DNNs to realize safety-relevant elements for automated driving.

Various methods are used to aid the reader of this publication. Each of the twelve principles described in Section 1.3 is assigned a pictogram, which is then used as a visual reference in subsequent chapters. This publication also uses the following four development examples and their pictograms throughout for further clarity:



L3 Traffic Jam Pilot (TJP) as an option for vehicle customers: Vigilant driver with driver's license, driving only on structurally separated roads, typically no pedestrians or cyclists, 60 km/h max, only with leading vehicles, no lane changing, no construction sites, only during daylight, without rain, only temperatures higher than freezing point



L3 Highway Pilot (HWP) as an option for vehicle customers: Vigilant driver with driver's license, driving only on structurally separated roads, 130 km/h max, with and without leading vehicles, lane changing, construction sites, at night and during daylight, moderate rain and snow



L4 Urban Pilot (UP) in fleet operation in urban areas: Non-vigilant driver, not capable of driving, no driver's license necessary, 70 km/h max, large ODD with safety driver, very limited ODD without safety driver, allows for indirect teleoperation if necessary



L4 Car Park Pilot (CPP) as an option for vehicle customers and in fleet operation: Driverless movement within certified parking structures or areas (no vigilant driver, no driver's license necessary), 10 km/h max, ODD focus on off-street parking and logistic areas, scalable use of infrastructure (infrastructure not mandatory but possible up to teleoperation)

1.3 Safety Vision

1.3.1 Background

According to the German traffic accident statistics published by the Federal Statistical Office of Germany (Destatis, 2018), over 98% of traffic accidents are caused, at least in part, by humans (Destatis, 2018, p. 146). Similarly, the US National Highway Traffic Safety Administration (NHTSA) reports that 94% of serious vehicle crashes in the US are caused by human error (NCSA, 2015, p. 142). Therefore, introducing automated driving poses great potential for reducing crash rates. However, there are also major challenges in realizing the full safety benefit of automated driving in order to achieve the target of a “positive risk balance compared to human driving performance”, as recommended by the German Ethics Commission in June, 2017 (BMVI, 2017).

Taking a deeper look at the statistics published by Destatis, which also serve as an indicator of human driving performance, it can be argued that human beings are a reasonable factor for traffic safety (Destatis, 2018). There is an average distance of 300,000 km between two crashes of any severity with respect to a lifetime mileage of 700,000 km. Statistically, the average distance between two fatal crashes in the USA is 228 million km and as many as 661 million km on the German highway or “Autobahn” (Destatis, 2018, p. 146; NCSA, 2015, p. 142).

1.3.2 The Twelve Principles of Automated Driving

Automated driving will improve performance in most situations compared to that of human drivers. However, it will not completely eliminate the risk of accidents or crashes. The goal of this publication is to present a generic approach for tackling the risks introduced by automated vehicles. While this common generic approach should be interpreted as a baseline for safe automated driving, it does not define a specific product that is complete and safe.



SAFE OPERATION

DEALING WITH DEGRADATION

If safety-related functions or system components become hazardous (e.g. unavailable), the automated driving system shall:

- Be capable of compensating and transferring the system to a safe condition/state (with acceptable risk).
- Ensure a sufficient time frame for the safe transition of control to the vehicle operator.

FAIL-OPERATIONAL (limited to the safety-related function or component)

The loss of safety-related functions or system components shall not lead to a safety-related situation.



OPERATIONAL DESIGN DOMAIN

ODD DETERMINATION

As soon as system limits that restrict the safe functionality of the automated system are recognized, the system shall react to compensate or shall issue a driver takeover request with a sufficient time frame for the takeover.

MANAGE TYPICAL SITUATIONS

The automated driving system shall take into account situations that can typically be expected in the ODD and address possible risks.



VEHICLE OPERATOR-INITIATED HANDOVER

Engaging and disengaging the automated driving system shall require an explicit interaction from the vehicle operator, indicating a high confidence of intent.



SECURITY

When providing an automated driving system, steps shall be taken to protect the automated driving system from security threats.



USER RESPONSIBILITY

To promote safety, the user's state (i.e. state of alertness) must be suitable for a responsible takeover procedure. The system should be able to recognize the user's state and keep them informed about their responsibilities concerning the required user's task. It should also be able to inform the respective operator about safety-relevant driving situations in unmanned driving services.

RESPONSIBILITIES

The aspects of the driving task which remain under the user's responsibility must be clear to the user.

MODE AWARENESS

The automated function must ensure that the currently active driving mode can be recognized explicitly and unmistakably at any time. In addition, a change in driving mode must be clearly apparent to the user as well.



VEHICLE-INITIATED HANDOVER

MINIMAL RISK CONDITION

If the vehicle operator does not comply with a takeover request, the automated driving system must perform a maneuver to minimize risk, resulting in a minimal risk condition. This maneuver depends on the situation and the current performance of the automated driving system.

TAKEOVER REQUESTS

Vehicle-initiated handovers shall be clearly understandable and manageable for the vehicle operator.



INTERDEPENDENCY BETWEEN THE VEHICLE OPERATOR AND THE ADS

The overall evaluation of system safety needs to take effects on the driver due to automation into account, even when they occur immediately after the period of automated driving has ended and when a direct link to the automated driving part of the journey can be drawn.



SAFETY ASSESSMENT

Verification and validation shall be used to ensure that the safety goals are met so as to reach a consistent improvement of the overall safety.



DATA RECORDING

Automated vehicles shall record the relevant data pertaining to the status of the automated driving system when an event or incident is recognized in manner that complies with the applicable data privacy laws.



PASSIVE SAFETY

CRASH SCENARIOS

The vehicle layout should accommodate modifications to crash scenarios resulting from vehicle automation.

ALTERNATIVE SEATING POSITIONS

Occupant protection shall be ensured even when the customer has new uses for the interior that are made possible through automated driving systems.



BEHAVIOR IN TRAFFIC

MANNERS ON THE ROAD

The behavior of the automated function needs to not only be easy-to-understand for surrounding (vulnerable) road users, but also predictable and manageable.

CONFORMING TO RULES

The applicable traffic rules are to be taken into account by the automated driving system. The above principle „User Responsibility“ describes the remaining user responsibilities.



SAFE LAYER

The automated driving system shall recognize system limits, especially those that do not allow the safe transition of control to the vehicle operator, and react to minimize the risk.

The generic approach of this publication is based on the twelve principles presented above, comprising a collection of publications and recommendations from mainly public authorities or consumer associations (IWG, 2017; ABI & Thatcham Research, 2017; NTSB, 2017; NCSA, 2015; BMVI, 2017; StVG, 2018). These principles provide a foundation for deriving a baseline for the overall safety requirements and activities necessary for the different automated driving functions under consideration of a positive risk balance.

The purpose of this publication is to highlight safety and security-relevant aspects of developing, producing, operating and maintaining automated driving vehicles; the combination of which lead to a safe product on the road. The aspects brought forward should contribute toward a foundation for the safety of automated driving vehicles. The consortium partners of this publication share the common goal of their automated driving vehicles being better than the average human driver during automated guidance and slightly before or after transitioning to human guidance within the same ODD in terms of avoiding or mitigating related hazards with elevated severity, e.g. collisions or roadway departure crashes. At the same time, a slightly negative safety balance of the automated driving system in rare improbable scenarios may still be acceptable, providing a positive risk balance is maintained across all situations.

Chapter

02

**SYSTEMATICALLY DEVELOPING
DEPENDABILITY TO SUPPORT
SAFETY BY DESIGN**

2 Systematically Developing Dependability to Support Safety by Design

This chapter describes how the three dependability domains safety of the intended functionality (SOTIF), functional safety and cybersecurity work together and how to combine them to create a dependable system. The chapter begins by introducing each domain and deriving automated driving capabilities from dependability. It then provides elements that can implement these capabilities. Lastly, it combines all elements by introducing a generic logical architecture (see Figure 2).

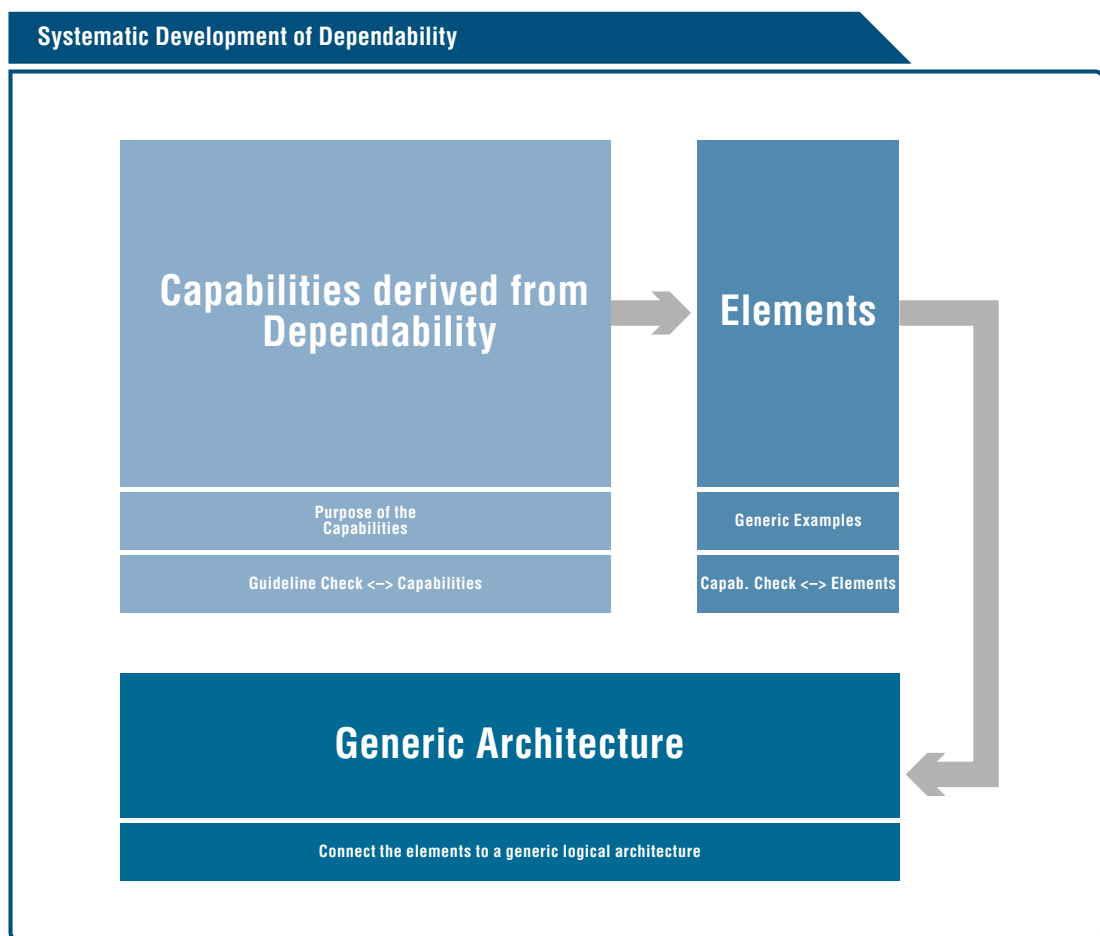


Figure 2: Systematic Development of Dependability

2.1 Deriving Capabilities of Automated Driving from Dependability Domains

Deriving capabilities from dependability domains begins with an overview of different international legal frameworks for automated driving vehicles to identify the requirements that capabilities should cover in addition to the twelve principles. The capabilities cover both SOTIF, which deals with human factors, and functional safety. Security works on the logical and technical architecture and provides input requirements for both. As there is currently no approved legislation or international standardization on automotive cybersecurity available, this section provides advice on security approaches and measures.

2.1.1 Legal Frameworks for Automated Driving Vehicles

Rules that explicitly address automated vehicles need to be fulfilled as well as those that apply to vehicles and road users in general, e.g. road traffic laws. All automated driving systems should comply with the legal regulations applicable to their ODD. This may include a set of federal, national and international regulations such as the following:

THE EU, JAPAN, REST OF THE WORLD (UN REGULATIONS)

The Vienna Convention of 1968 states that the driver must be in control of their vehicle at all times (United Nations, 1969). In 2014, the UNECE amended the regulation to include highly automated systems, provided that these continue to have a driver who is ready to take over driving functions and who can override the system and switch it on and off. However, this still presupposes that every vehicle must have a driver. UNECE WP.1 has already affirmed that the 1949 and 1968 Conventions apply to all driving situations, except those situations where the vehicle is moved exclusively by vehicle systems without the driver assuming any role at all.

UNECE WP.1 is currently working on a draft resolution regarding the deployment of highly and fully automated vehicles in road traffic, which includes recommendations to contracting parties of the 1949/1968 Conventions on how to safely deploy such new technology (ECE/TRANS/WP.1/165, 2018).

US

The U.S. DOT *Federal Automated Vehicles Policy* of 2016 (U.S. DOT, 2016), replaced with the *Automated Driving Systems: A Vision for Safety* (NHTSA, 2017), provides the industry and states with a framework to analyze and communicate a safety strategy using a Voluntary Safety Self-Assessment on automated driving systems for SAE Automation Levels 3–5. This framework highlights a wide range of demands on the development and safety verification and validation of automated driving systems that are also based on twelve principles. In order to realize the appropriate legal frameworks, automated driving system regulations at the international, national, regional and local levels must coexist and coordinate with

minimal conflict while taking into account existing automobile legal frameworks. There is a need for one legal framework at the national level that may provide the base framework for automated driving system regulation. This would form the foundation for the development of new Federal Motor Vehicle Safety Standards (FMVSS) by the National Highway Traffic Safety Administration (NHTSA).

There is currently an opportunity and the need for the governments of the world to analyze their present automobile legislation to understand the areas that require adaptation. Doing so will promote and enable the mass production of the different levels of automated driving systems and will, in particular, facilitate regulation of the safety of near-future SAE L3 and L4 technology.

CHINA

The Ministry of Industry and Information Technology of the People's Republic of China released the *Guidelines for the Construction of the National Internet of Vehicle Industry Standard System (Intelligent & Connected Vehicles)* in 2018 to comprehensively strengthen the top-level design and to promote the Intelligent and Connected Vehicle industry research and development. Moreover, China is developing extensive new regulations (30 new standards by 2020, 100 new standards by 2025) that adapt to China's national conditions and international practices.

2.1.2 Applying the Related Safety Standards

In addition to regulations, it is not possible to achieve the necessary degree of safety unless the system can control safety relevant use cases that result either from the intended use or from unlikely E/E faults. Furthermore, foreseeable misuse should be considered where this is not a deliberate manipulation of the system. Partially automated systems (SAE L2) that are in series production today increasingly support the driver in lateral and longitudinal control tasks. The advancement of driver assistance systems will see users relieved from the driving task, and the introduction of highly automated driving (SAE L3 and higher) where the driver will be discharged from the driving task altogether. A holistic systems perspective is utilized to safeguard safety throughout this technological development. This perspective is covered in part by established development standards but also requires the development of new ones.

There are currently no development standards or a state of the art for automated driving systems since such systems do not yet exist and because the solutions that are available lack maturity and are not deployed. The automotive industry presently uses other resources in addition to ISO 26262 to define the safety design of an automated driving system. Experience, literature, studies and standards developed by other industries are relevant resources for the automotive community for defining safety standards and processes for the domain of automated driving systems. The second revision of ISO 26262 has matured to include more rigor and structure to support more complex automotive electronic systems. The recently released ISO/PAS 21448 standard specifies a development process for the analysis, verification and validation of non-faulted scenarios and use cases of a system. However, ISO/PAS 21448 looks only at L1 and L2 automated systems. Thus, this publication attempts to extrapolate this standard to L3 and L4 applications.

The above standards complement each other and may be used primarily to define the design risk of an automated driving system, equip the engineering teams to design safety mechanisms and augment the intended functionality of an automated driving system to mitigate the risk identified.

Existing standards do not present solutions to some of the most problematic topics of automated driving systems, such as the safety assurance of artificial intelligence (the most relevant algorithms derive from the fields of machine learning and neural networks, see Appendix B), human factors and psychology, and the technological capability of the sensory devices used as inputs to the automated driving system. Nevertheless, safety-related use cases should be analyzed to ensure the necessary levels of safety. These analyses systematically assess the functional descriptions for possible hazards arising from the intended use and from foreseeable misuse. In addition to a safe design and development process, assessment progresses iteratively from verification to validation and comprises expert appraisals, safety analyses and experiments. Depending on their scope, the different standards support this procedure.

Initially, a safe functionality should be defined. ISO/PAS 21448 was developed to address the level of risk and hazards caused by the intended functionality, including foreseeable misuse. Danger stemming from E/E malfunctions of the system is addressed by functional safety using the globally established ISO 26262 standard, whereas danger as a result of deliberate manipulation is assessed from an ISO/SAE 21434 security point of view. Implementing the safety standards ISO/PAS 21448, ISO 26262 and ISO/SAE 21434 would allow the combining of their procedures and methods. Depending on the development organization's needs, it may be necessary to develop the standards independently, taking their dependencies into account while doing so.

When implementing automated vehicle functions, the risks based on the functional and system boundaries should be evaluated. Performance limitations (e.g. based on sensor limitations) could result in malfunctioning behavior, potentially introducing hazardous situations. Development standards will aid developers in managing the complexity of the systems, estimating possible risks and addressing adequate measures. Finally, there is no standard available that addresses an adequate correlation of a system's safety and availability. Using ISO 26262 for availability aspects has its limits for the majority of applications. Designing a system to be safe is a balancing act of risk and availability of the application. Being too risk adverse leads to a system that is overly conservative, and the system availability becomes too low (safety mechanisms to reach sufficient integrity could endanger availability goals), which in turn will not provide the benefits of a safer and more comfortable customer experience. On the contrary, if the system safety design is too liberal, it will have the effect of a system that is not safe enough but may be available all the time as shown in Figure 3:

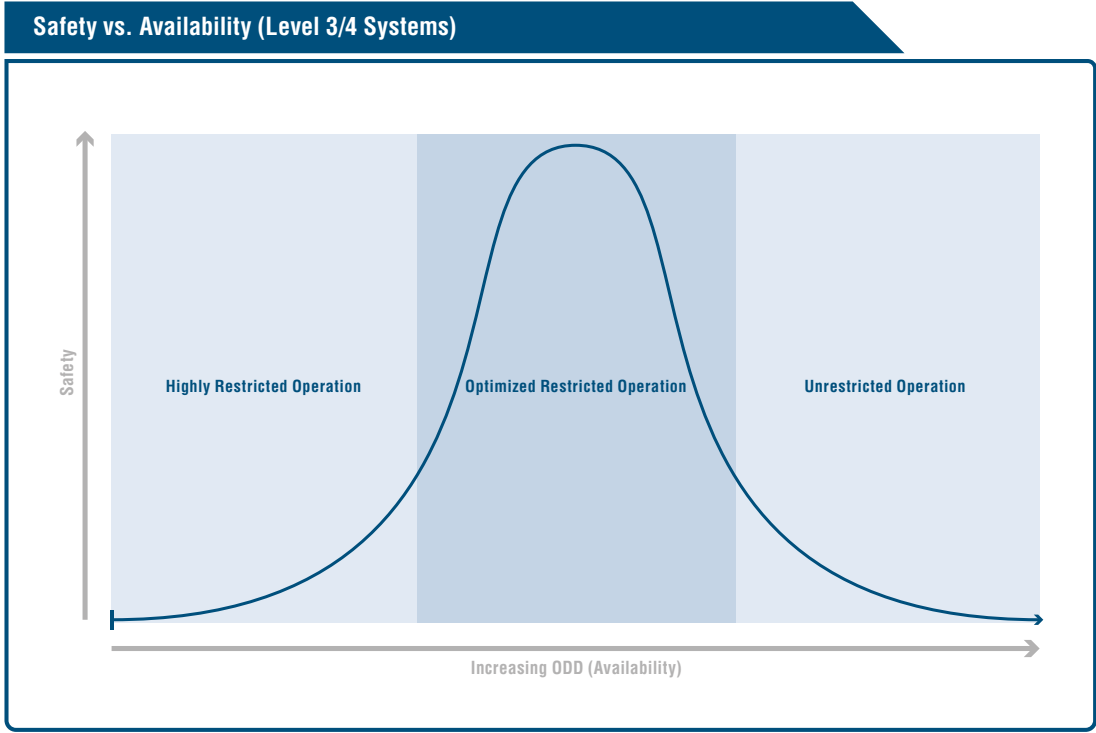


Figure 3: Safety vs. Availability (Level 3-4 systems)

Thus, it becomes increasingly important to approach the problem from an analytical perspective, addressing the safety of the system by designing scenario-based system behaviors and addressing the technological capability through the analysis of use cases and scenarios to design a robust and safe system. This is what this publication defines as the concept of safety by design, and this is the approach that is detailed throughout this document.

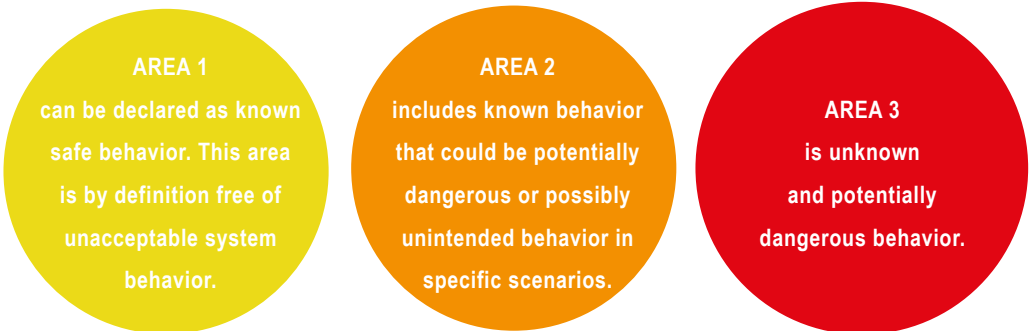
Availability is not only affected by ODD restrictions; it can also be compromised by an overly conservative, overly complex or flawed safety mechanism design, flawed system design, insufficient sensor diversity regarding modality and redundancy, environmental factors, human mode confusion arising from poorly designed HMI, or automation effects (interdependency between the vehicle operator and the ADS).

To achieve the balance between fail-safe and availability, the design is analyzed and built from the top down. The first analysis is carried out irrespective of the generic logical architecture. The process includes risk assessments to determine the safety requirements of the system being designed. Ultimately, this evolves into a safety concept, defining safety mechanisms to support the safety goals.

2.1.3 Safety of the Intended Functionality

The basic concept of the safety of the intended functionality (SOTIF) approach is to introduce an iterative function development and design process that includes validation and verification and that leads to an intended function that could be declared safe. Several activities will be derived based on an approach that argues that these activities are adequate for developing an automated functionality that is safe.

This approach assumes that there is an area of known scenarios with safe system behavior and an unknown area with potential harm. In reality, these areas overlap as visualized in Figure 4. The areas depicted in the Venn diagram are defined as follows:



The remaining white area is by definition safe and unknown.

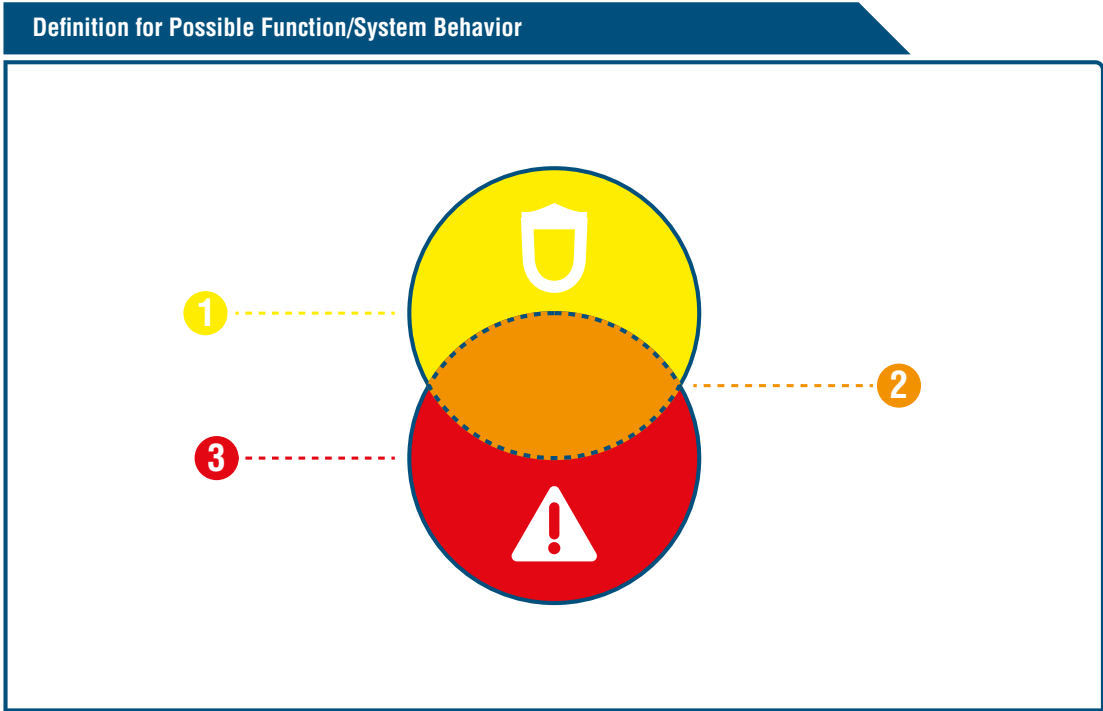
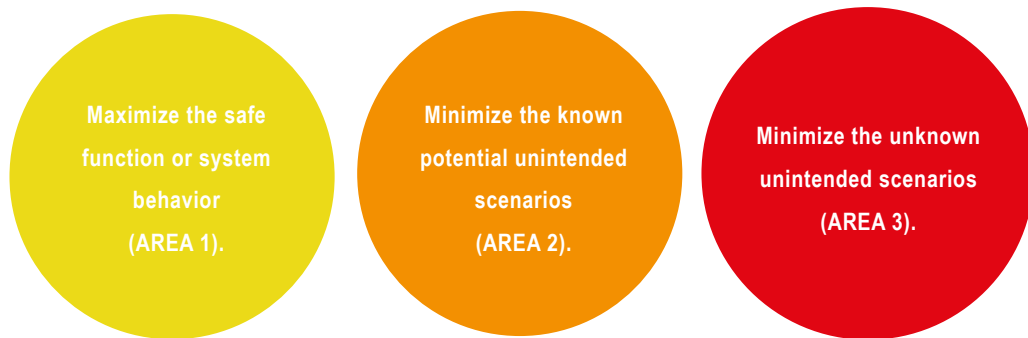


Figure 4: Definition for Possible Function/System Behavior

The automotive development goal is to reduce the known potentially unintended behaviors and the unknown potential behavior to an acceptable level of residual risk. Using the above model, the following development goals can be derived:



The white area is not relevant for the argument of this publication, because it is already safe. In any case, the white area could be also reduced through measures for reducing Area 3 or expanding Area 1. Figure 5 visualizes the result by depicting the development goals:

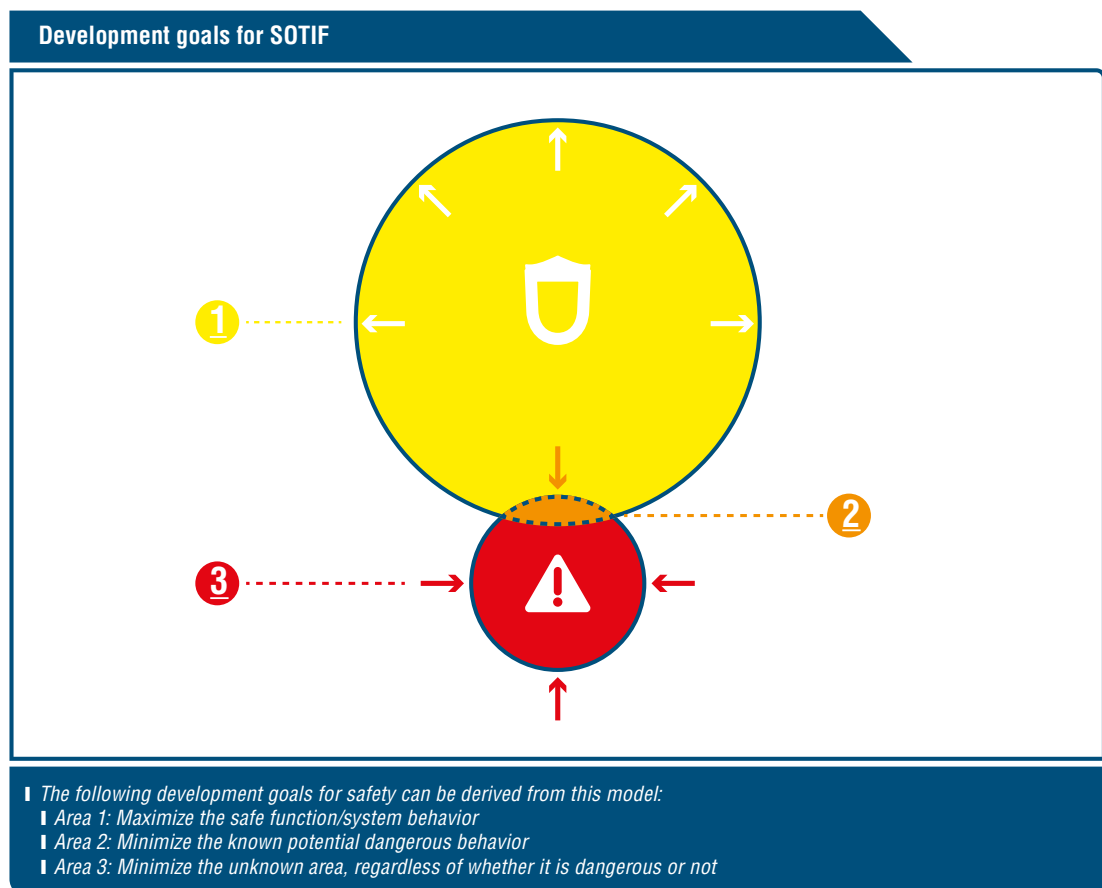


Figure 5: Development goals for SOTIF

The following measures contributes to the development goals using this concept. The measures will lead to an iterative change and improvement process:

MEASURES FOR AREA 1

CLEARLY DEFINE THE FUNCTION TO BE DEVELOPED TO:

- Identify potential risks via analyses
- Improve the definition where weaknesses are discovered

The functional & technical specifications will be analyzed using a risk analysis similar (but not identical) to that of ISO 26262. If weaknesses are detected, the functionality or system will be improved.

MEASURES FOR AREA 2

VERIFY THE FUNCTION, INCLUDING ITS SYSTEM COMPONENTS TO:

- Simulate the function, including its scenarios
- Test the system component and the overall system
- Identify where improvements can be made to the functions or the system in the event of weaknesses
- Determine a basis for acceptance for residual risk

MEASURES FOR AREA 3

VALIDATE THE FUNCTION SYSTEM TO:

- Reduce Area 3 to an acceptable level, e.g. via endurance testing, driving tests, simulation

Chapter 3 can be referred to at this point since the main evidence to demonstrate that the system is safe enough for the customer is provided via verification and validation activities.

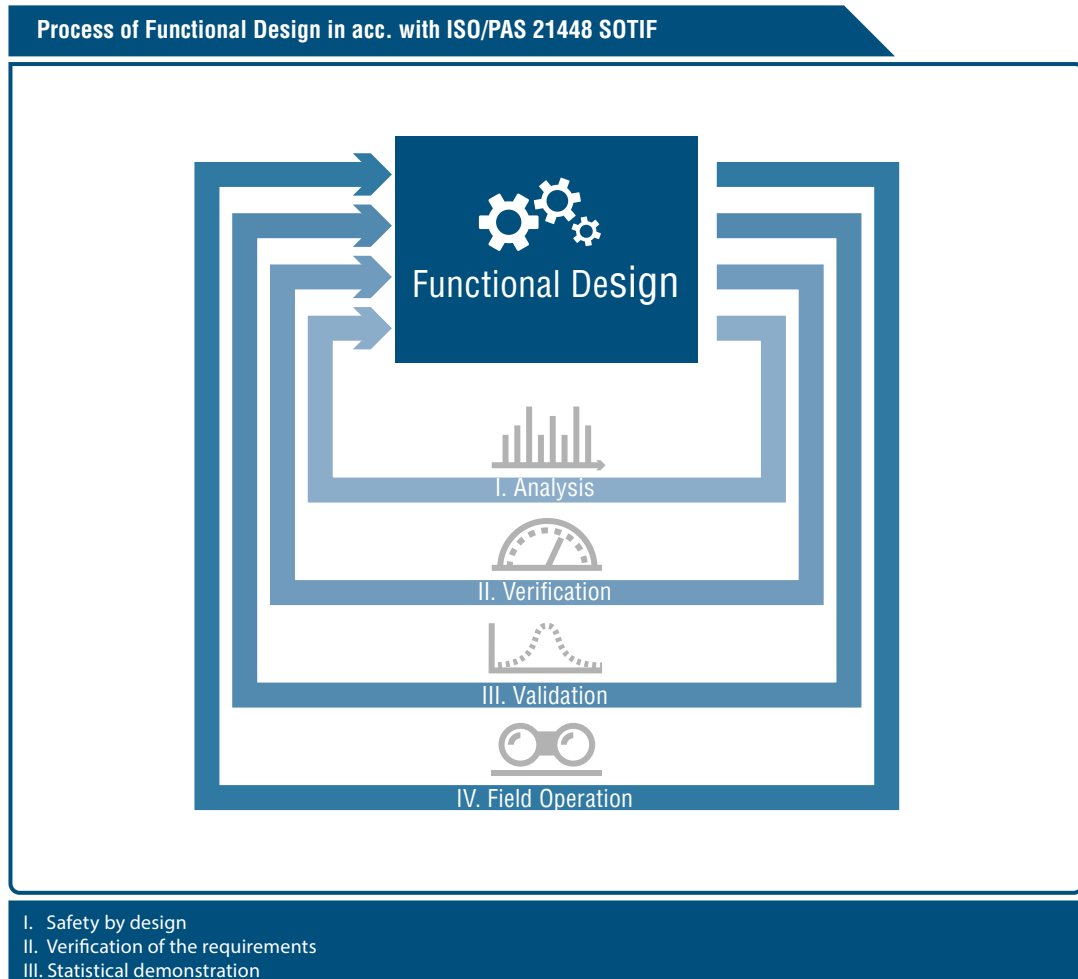


Figure 6: Process of Functional Design in acc. with ISO/PAS 21448 SOTIF

2.1.4 Functional Safety

The application of ISO 26262 is specific to applications for passenger vehicles, motorcycles and commercial motor vehicles, and more specifically to the practice of functional safety. In this standard, standard risk is determined and communicated or mapped using Automotive Safety Integrity Levels (ASIL). This publication will not attempt to directly trace and decompose ASIL values in a traditional application of functional safety, but these may be indirectly inferred from the architectural examples that are derived in the sections below. However, as practitioners of functional safety, it is understood that an E/E fault leading to a failure at the system-level capabilities and functions will contribute to incorrect steering or braking, which are considered the highest safety-related risk of ASIL D.

Defining the safe function and the system, including a first preliminary architecture, via the function developed in accordance with the ISO PAS 21448 SOTIF approach is the first step toward applying functional safety in accordance with ISO 26262 – the creation of the item definition. This item definition should include a definition of the functions, including their dependencies on and interaction with the environment and other items/vehicles. Based on the item definition, a hazard analysis and risk assessment can then be carried out to find the root requirements or safety goal for the function and its related system. The next steps are developing functional and technical safety concepts.

The first edition of ISO 26262 was created based on the knowledge of state-of-the-art systems in automotive industries (such as steering, braking and airbag systems, etc.) and does not adequately address very complex and distributed systems. Furthermore, there is no clear methodology that covers the need for availability to uphold safety. The second edition resolves some of these issues but fails to address many others. Thus, interpretations are needed, and some of the issues that should be resolved include:

- Addressing the gaps in the first and second editions of ISO 26262 to devise solutions for availability requirements
- Missing automotive architecture models in ISO 26262, e.g. for failure rates estimation as described in other standards such as IEC 61508
- Exiting and reused architecture elements are designed with fail-safe behavior. Thus, the new system designs should create fail-operational or fail-degraded behavior
- Decomposition of the given architecture elements to achieve the required ASIL
- Definition of the functional and architecture elements necessary to achieve the required ASIL

Meeting all challenges will result in the definition of a safe function and mean that weaknesses of the technologies have been considered (SOTIF) and that possible E/E faults can be controlled by the system or by other measures (ISO 26262). Consequently, it will be possible to declare the automated system safe without manipulation, which is currently not covered by ISO/PAS 21448 or ISO 26262.

2.1.5 Automotive Cybersecurity

People often group safety and security, even if they cannot articulate why or how these topics relate to each other. This grouping is natural due to the overlapping properties that the topics are built upon. However, their focuses are subtly different, because safety focuses on the proper functioning of a system, and security focuses on the system's ability to resist some form of intentionally malicious action. In particular, these center around safety worries about risks presented by passive adversaries, randomness in nature and human-caused accidents or crashes and security worries about risks presented by active adversaries in the form of creative, determined and malicious human beings acting intentionally. This leads security to utilize additional analysis tools and technical mechanisms that nevertheless also affect safety.

For example, safety and security both focus heavily on data integrity. Safety often relies on CRCs to detect corruption, but CRCs are not robust against malicious actors. Thus, security instead relies on secure hashing algorithms and secrets to detect corruption and intentional tampering while resisting attack. Furthermore, security should grapple with the possibility that the data may be entirely forged by an

unexpected source and should therefore verify the data's source and integrity to achieve an acceptable level of risk. Availability is similar: Where safety emphasizes fail safes and degraded modes, security focuses on avoiding unavailability where possible, because a fail-safe or degraded mode may provide attackers an advantage.

Security makes heavy use of cryptography, which is often resource-intensive, but active safety mechanisms should be deterministic. Safety-related data often comes with requirements for short processing deadlines, which makes it difficult to ensure required levels of data authenticity, confidentiality, etc. Satisfying both safety and security will impact resources and the architecture.

2.1.5.1 WHY IS CYBERSECURITY SO IMPORTANT FOR SAFETY?

The automotive industry is facing new challenges in automated driving due to the extreme connectivity within automated driving vehicles and between those vehicles and their operating environment. These challenges range from fulfilling regulatory requirements and ensuring safety to protecting fleets and customers from cybersecurity attacks. Connectivity additions include new interfaces between the control functions of connected vehicles, IT backend systems, and other external information sources (see Figure 7). This rich attack surface creates considerable interest for malicious actors with various goals. In short, we have advanced to a level where vehicles cannot maintain a safe state unless they also operate securely. Most importantly, cybersecurity principles and practices should be applied to ensure that attackers cannot gain arbitrary control of a vehicle's movement and that attacks are exceptionally difficult to scale to the point of simultaneously exploiting multiple vehicles.

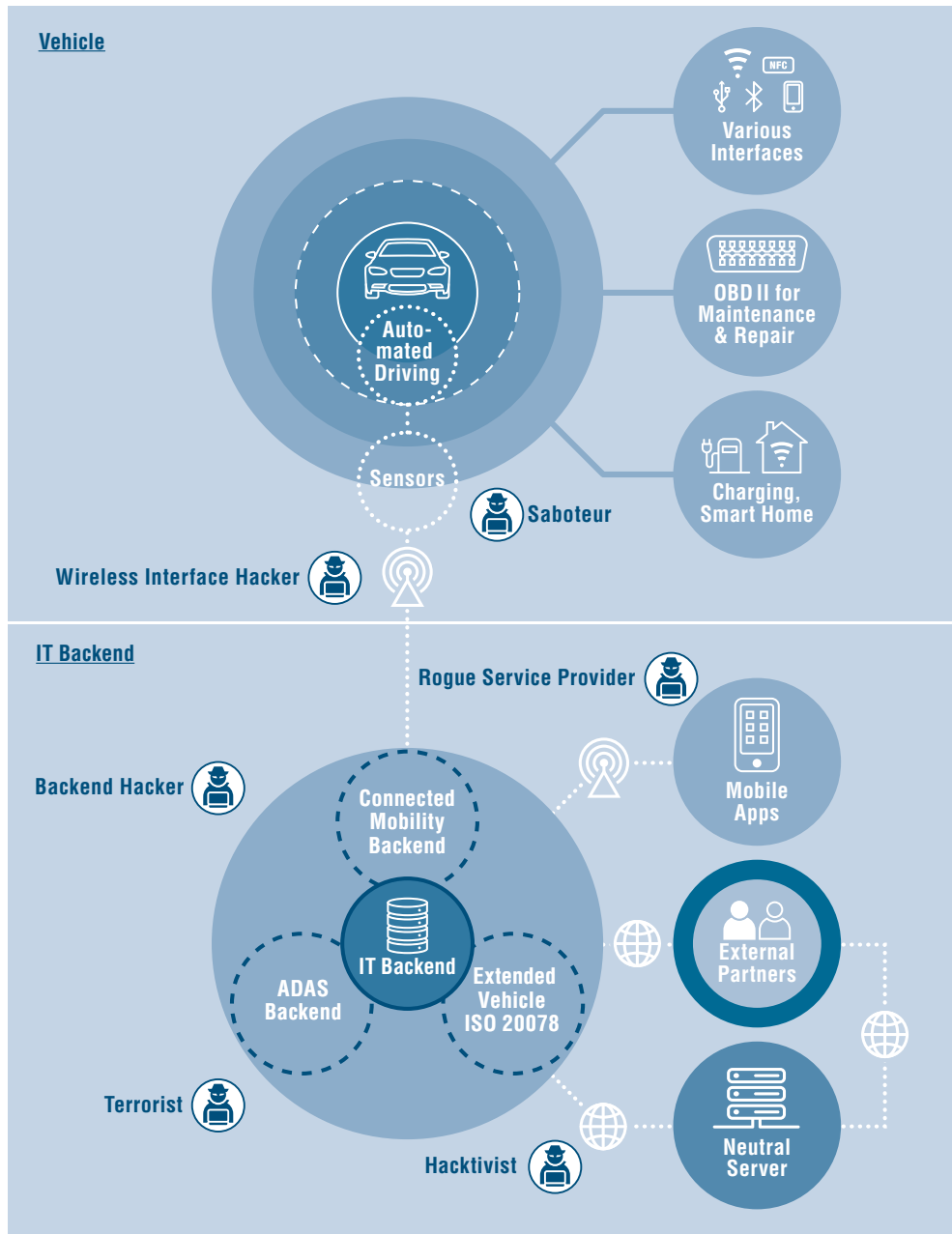


Figure 7: Automotive Cybersecurity

As the degree of vehicle automation increases, the security measures protecting vehicle functions should defend against unauthorized access and manipulation to guarantee the integrity of the vehicle, its components and the safe operation of its functions, especially vehicle control functions. It is in our fundamental interest to ensure the highest safety standards and to protect vehicle safety in the best possible way, taking into account the state of the art in technology when doing so.

This publication focuses on cybersecurity to protect the safety of an automated driving vehicle. Therefore, some examples are given of how to secure the additional components required for safe automated driving, required external information and new external interfaces necessary for automated driving.

The challenge for cybersecurity when extending from an L2 to L3 or L4 vehicles is that the automated driving functionalities are critically reliant on external data, e.g. sensor information, maps, positioning information, etc. If the integrity or authenticity of this data is compromised, the building blocks of the automated driving functions (Sense – Plan – Act) will use faulty data to maneuver the vehicle, which might result in inaccurate driving or other deviations from correct operation. If an automated vehicle is attacked, the impact will be much higher as the person inside the vehicle will not be able to take control in time, if at all. Therefore, cybersecurity measures should be introduced sufficiently to protect automated driving from malicious actors.

2.1.5.2 CYBERSECURITY APPROACH AND MEASURES

This section discusses the approach to address the threats described above. The approach begins with an overview of the development process used to build automated driving systems that resist attack. Security has to be designed into a system to achieve ubiquitous coverage. A rigid and fully integrated security engineering process is the basis for creating secure – and therefore safe – systems.

The process helps to tightly integrate various security controls, which are described after the process. Traditional computing security often focuses on establishing what is known as defense-in-depth. In this publication, defense-in-depth is adopted to ensure controls are layered throughout the system to prevent reliance on the perimeter alone to withstand attack.

2.1.5.2.1 SECURE DEVELOPMENT LIFECYCLE

A Secure Development Lifecycle (SDL) is a process for building in security. In particular, an SDL prescribes security practices to be carried out at a specific phase in a development process. The practices are diverse and aim to either pro-actively prevent attacks or to find and fix vulnerabilities early on. SDLs are tailored to fit into the development process used as part of product development and product maintenance.

Regardless of the development process used, SDLs typically bucket practices broadly into one of three categories: Preliminaries, development, and sustainment. Preliminaries include training to ensure a knowledge baseline within the development organization, and policy, procedure and guideline creation to ensure the rest of the process has the required grounding. Development includes practices that are familiar in software engineering, such as security requirements definition, threat modeling, static and dynamic analysis, fuzzing, code review and penetration testing. Finally, sustainment includes incident response, update sign-off procedures and other practices that ensure the product continues to operate after release.

A uniform SDL is not yet utilized across the industry, because SDLs are most effectively adopted when they align with the way each organization delivers their product to the market (Microsoft, 2019). However, the use of these lifecycles ensures a structured approach for pro-actively addressing security concerns in the development process.

When applying various SDL practices, there are many trade-offs to make. Risk assessments are used to help decide where to spend limited resources and how to prioritize approaches. At least three dimensions are important when considering trade-offs during development (as depicted in Figure 8): Risk treatment strategy, system state and risk treatment manifestation. The treatment strategy provides options for handling risk (e.g. avoiding, transferring, mitigating or accepting risk), system states help to define appropriate mitigations, and the treatment manifestation helps to understand how the selected approach alters the resulting risk.

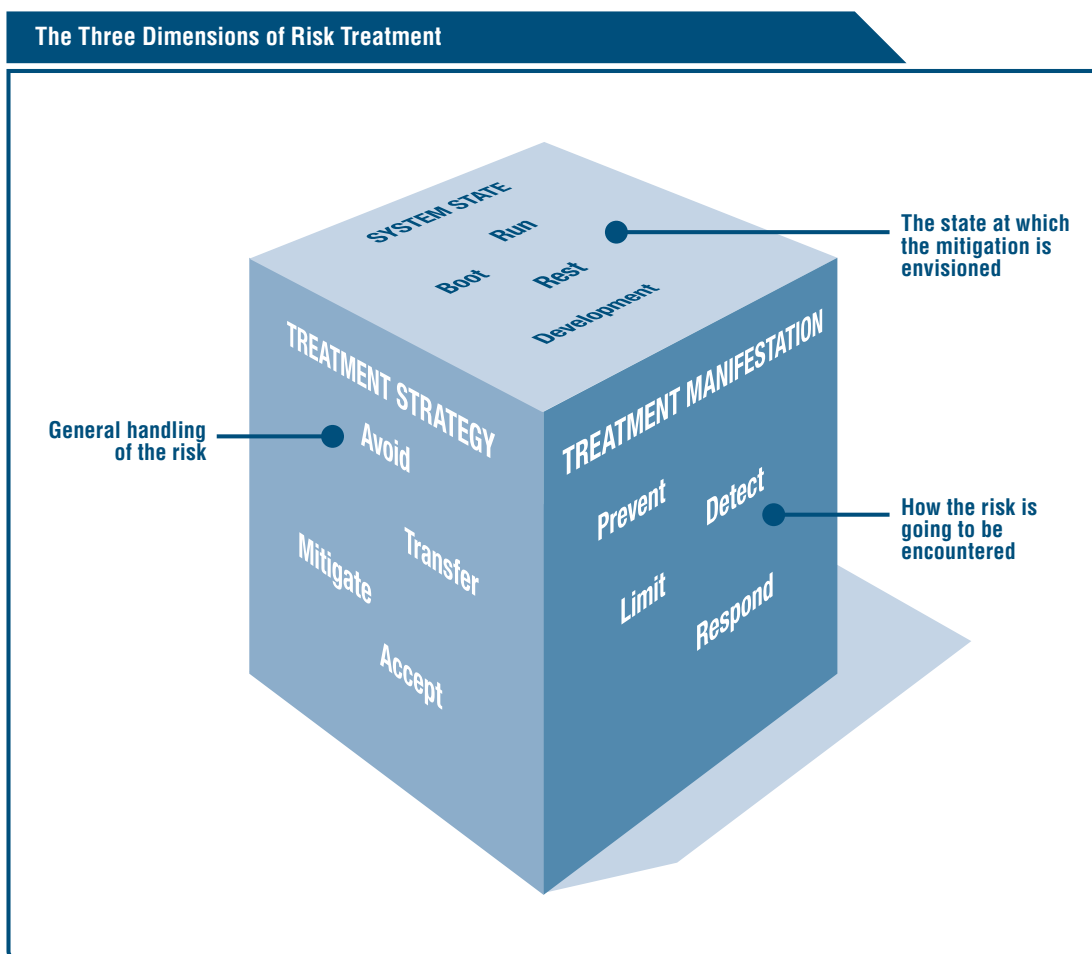


Figure 8: The Three Dimensions of Risk Treatment

2.1.5.2.2 DEFENSE-IN-DEPTH SECURITY ARCHITECTURE

This section discusses how security functions are layered to achieve the previously discussed security goals. For automated driving vehicles, defense-in-depth starts with low-level components and carries through to individual devices, groups of devices that form identifiable systems (e.g. perception), the vehicle itself and the infrastructure required to support the vehicle.

To achieve the automotive security goals, the traditional information security triad of confidentiality, integrity including authenticity and availability – known as CIA – of data, (sub) systems, functionalities or components is adopted to embedded systems for automated driving. At the component level (e.g. microcontrollers, ECUs, camera sensors, etc.), primitives are available to securely implement confidentiality, integrity, and authenticity (see Table 1). This generally means ensuring that microcontrollers implement or integrate with a hardware security module or similar purpose-built hardware where cryptographic functions utilized by higher level functions can be built. This publication also considers functions to establish component tamper-resistance, configurability (e.g. to remove or disable unneeded functions) and updatability at this point.

Moving up a layer, components are composed to construct devices (e.g. LIDARS, radars, camera units, etc.) where the security established by components is leveraged. At this point, this publication establishes the integrity and authenticity of firmware and software (secure boot), encrypts and authenticates messages, and authenticates entities authorized to update the device and the updates themselves. It also considers functions to mitigate denial-of-service attacks executed against the device and how to prevent unintended information disclosures from the device.

When the devices are composed to form systems, functions to secure group communications, attest to the state of a device to the rest of the group and resist denial-of-service against shared communication channels become relevant. Additionally, the methods used in this publication benefit from redundancy utilized for safety at this point. Utilizing sensor fusion and cross-referencing what is perceived across multiple modalities forces an attacker to coordinate their attack across multiple, diverse devices to fool the entire system. Independent safety systems within the vehicle further contribute to defense-in-depth, because the compromising of one system does not necessarily alter the operation of a separate system that can detect and respond to safety problems.

Automated vehicles do not operate by themselves. Vehicles are supported by public (e.g. DSRC, GNSS, etc.) and private (e.g. back-office functions) infrastructures. While automated driving vehicles take under consideration data received from an infrastructure, particularly data that can be strongly authenticated and validated, the vehicles ultimately maintain their own decision authority, not the infrastructure. The many humans involved in operating, managing, maintaining, etc. automated driving vehicles are considered at this point as well. Human access to vehicles is limited and compartmented based on function. Operations personnel, for example, may need access to vehicle position and status but do not need to know who is in the vehicle.

All the above improves the security, reliability, and trustworthiness of L3 and L4 vehicles. Table 1 lists some examples of security controls that can contribute to achieving the security goals set out in this publication.

Examples for Security Controls			
Security Goal	Environmental Level Security Controls	Vehicle Level Security Control	Component Level Security Controls
Integrity	<ul style="list-style-type: none"> Integrity management of access rights 	<ul style="list-style-type: none"> Secure communications, TLS, IPsec, etc. Functional separation and a trusted execution of the control flow 	<ul style="list-style-type: none"> Access control Control flow integrity (CFI) Trust anchor
Authenticity	<ul style="list-style-type: none"> Access control to development and production sites Secure communications 	<ul style="list-style-type: none"> Message authentication codes etc. 	<ul style="list-style-type: none"> Secure boot with a trust anchor, e.g. public keys in OTP
Availability	<ul style="list-style-type: none"> Intrusion detection mechanisms to react to potential attacks 	<ul style="list-style-type: none"> Congestion control on gateways/routers 	<ul style="list-style-type: none"> Rate limiting on networking interfaces Deterministic scheduling
Confidentiality	<ul style="list-style-type: none"> Access control to documentation 	<ul style="list-style-type: none"> Encryption of data in flight TLS, IPsec, etc. 	<ul style="list-style-type: none"> Encryption of data at rest Secure storage

Table 1: Examples for Security Controls

2.1.6 Capabilities of Automated Driving

2.1.6.1 INITIAL DERIVATION OF CAPABILITIES

In order to comply with the twelve principles outlined in Section 1.3.2, an automated driving system has to have a basic set of system properties that are specified here as capabilities. The following will discuss the capabilities which should be present in order to claim that the overall system is safe.

The capabilities are divided into fail-safe capabilities (FS) and fail-degraded capabilities (FD). Fail-safe capabilities provide and enable customer value. Fail-safe capabilities can be discontinued, because the safety relevance of their unavailability is low enough or is covered by the fail-degraded capabilities. Fail-degraded capabilities should be performed with a certain performance level, even in the case of a failure, to provide a safe system for a specific timeframe until a final Minimal Risk Condition (MRC), allowing deactivation, is reached (see Section 2.1.7).

The selection matrix in Table 2 demonstrates the state of completeness of the derived capabilities to evidence the traceability to the principles from Chapter 1.

Traceability of the Capabilities













ID	Safe Operation	Safety Layer	Operational Design Domain	Behavior in Traffic	User Responsibility	Vehicle-Initiated Handover	Veh.-Op.-Initiated Handover	Interdep. Veh. Op. & ADS	Data Recording	Security	Passive Safety	Safety Assessment
												
FS_1 Determine location			X	X						X		X
FS_2 Perceive relevant objects				X						X		X
FS_3 Predict the future behavior of relevant objects				X						X		X
FS_4 Create a collision-free and lawful driving plan				X						X		X
FS_5 Correctly execute the driving plan				X						X		X
FS_6 Communicate and interact with other (vulnerable) road users				X						X		X
FS_7 Determine if specified nominal performance is not achieved		X	X							X		X
FD_1 Ensure controllability for the vehicle operator	X				X	X	X	X		X		X
FD_2 Detect when degraded performance is not available	X									X		X
FD_3 Ensure safe mode transitions and awareness	X	X			X	X	X	X		X		X
FD_4 React to insufficient nominal performance and other failures	X	X								X		X
FD_5 Reduce system performance in the presence of failures	X	X								X		X
FD_6 Perform degraded mode within reduced system constraints	X	X	X			X				X		X

Table 2: Selection Matrix for the Traceability of the Capabilities

As seen above, the safety principle for *Safety Assessment* traces to all capabilities. This derives from the expectation that product development would be responsible for delivering a necessary level of evidence for the verification and validation of the capabilities, which may then be reviewed by an assessing group. While this is recognized here, it will be discussed in greater detail in later chapters where the full logic and rationale surrounding the methodology for validating an automated driving system is developed.

During the time in which the system is performing nominally, system operation may be understood using the classic Sense – Plan – Act design paradigm from robotics and automation literature. In this model, *Sensing & Perception* (including *Localization*), *Planning & Control*, and *Actuation & Stability* provide a general, implementation-independent view of the automated vehicle system. Figure 9 illustrates this general model:

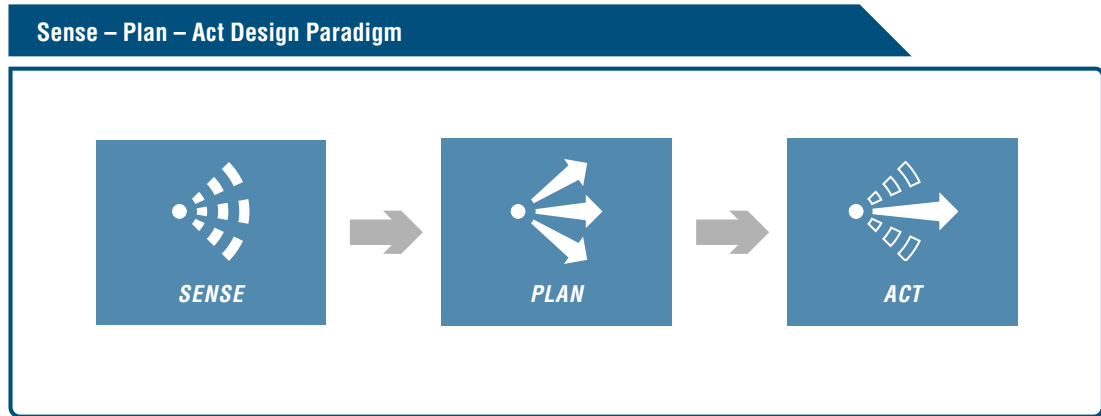


Figure 9: Sense – Plan – Act design paradigm

Based on the allocation of capabilities to the basic functions for Sense – Plan – Act, it is possible to allocate requirements for elements to ensure that the automated vehicle operates safely as depicted in Figure 10.

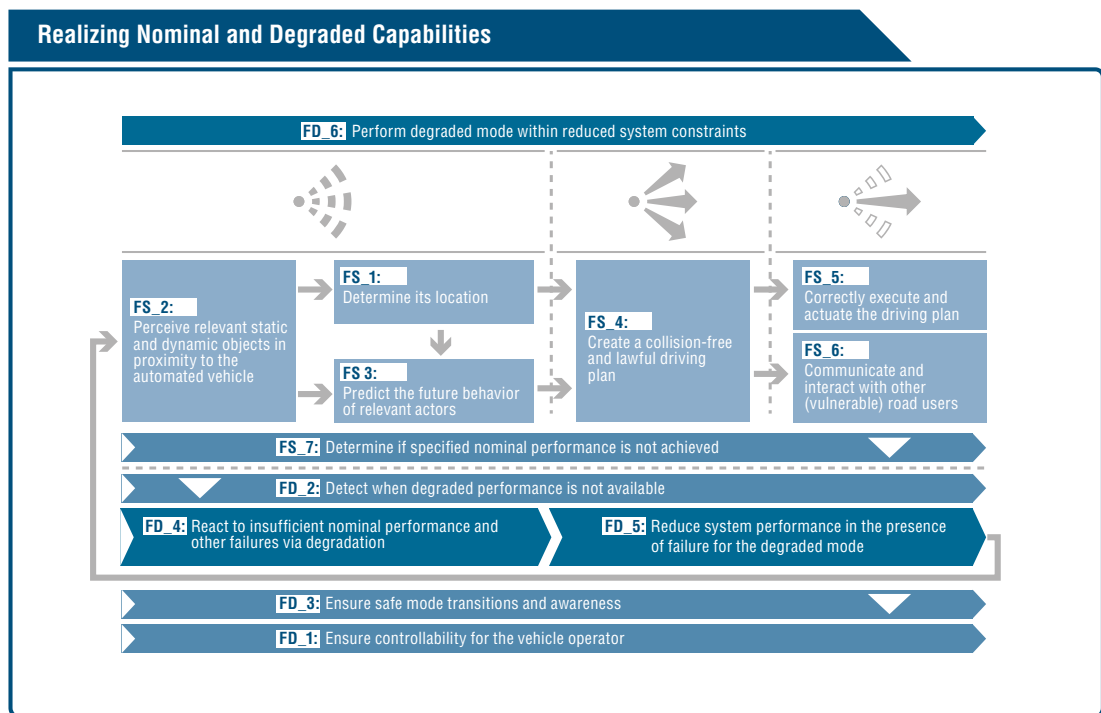


Figure 10: Realizing Nominal and Degraded Capabilities

2.1.6.2 OVERVIEW OF THE CAPABILITIES

FS_1: DETERMINE LOCATION

The system should be able to determine its location in relation to the ODD. The vehicle should be able to decide if it is inside or outside of a location-specific ODD. The location in the ODD may be required, depending on the item definition.

FS_3: PREDICT THE FUTURE BEHAVIOR OF RELEVANT OBJECTS

The relevant environment model needs to be extended by the predicted future state. The aim is to create a forecast of the environment. The intention of the relevant objects should be interpreted in order to form the basis for predicting future motion.

FS_2: PERCEIVE RELEVANT STATIC AND DYNAMIC OBJECTS IN PROXIMITY TO THE AUTOMATED VEHICLE

All entities that an automated driving system requires for its functional behavior should be perceived, optionally pre-processed, and provided correctly. The highest priority is placed on entities with an associated risk of collision. Sample entities include dynamic objects (e.g. (vulnerable) road users and characteristics of the respective movement), static instances (e.g. road boundaries, traffic guidance and communication signals) and obstacles.

FS_4: CREATE A COLLISION-FREE AND LAWFUL DRIVING PLAN

To ensure a collision-free and lawful driving policy, the following should be respected:

- Maintain a safe lateral and longitudinal distance to other objects.
- Comply with all applicable traffic rules within the ODD.
- Consider potential areas where objects may be occluded.
- In unclear situations the right of way is given, not taken.
- If a crash can be avoided without endangering third parties, traffic rules may be neglected if necessary.

**FS_5: CORRECTLY
EXECUTE AND ACTUATE
THE DRIVING PLAN**

The corresponding actuation signals for lateral and longitudinal control should be generated based on the driving plan.

**FS_6: COMMUNICATE AND
INTERACT WITH OTHER
(VULNERABLE) ROAD USERS**

Automated driving vehicles are required to communicate and interact with other (vulnerable) road users, depending on the ODD and the use cases.

**FS_7: DETERMINE IF SPECIFIED
NOMINAL PERFORMANCE IS
NOT ACHIEVED**

Any element of the automated driving system can, either on its own or in combination with others, result in adverse behavior. Therefore, mechanisms are required to detect the adverse nominal performance of the system. FD_4 covers the reaction to detected adverse behavior.

Typical aspects for influencing the nominal performance are:

- Unwanted human factors, including misuse and manipulations
- Deviation of the intended functionality
- Technological limitations
- Environmental conditions
- Systematic and random failure modes

Capabilities for recovering to nominal performance are possible but are not considered further in this publication, because they have no direct safety relevance. Fulfilling the capabilities is necessary but not sufficient for safe system operation. Additional capabilities will be required depending on the specified functionality and product.

FD_1: ENSURE CONTROLLABILITY FOR THE VEHICLE OPERATOR

The vehicle operator's level of control varies depending on the automation level as per SAE J3016 and the use case definition and should therefore be ensured.

FD_4: REACT TO INSUFFICIENT NOMINAL PERFORMANCE AND OTHER FAILURES VIA DEGRADATION

Due to possibly unavailable nominal performance capabilities and other failures (e.g. based on hardware faults), the system should degrade within a well-defined amount of time.

FD_2: DETECT WHEN DEGRADATION IS NOT AVAILABLE

It should be assured that a possible unavailability of the degraded mode is detected. If the degradation strategies depend on the degradation reason, the degradation reason should be identified.

FD_5: REDUCE SYSTEM PERFORMANCE IN THE PRESENCE OF FAILURE FOR THE DEGRADED MODE

The reaction in case of failures during degraded mode should be defined.

FD_3: ENSURE SAFE MODE TRANSITIONS AND AWARENESS

Ensure that mode transitions are performed correctly and controlled by the vehicle operator affected if necessary. The vehicle operator affected should also be aware of the current mode and their responsibility deriving from it. For example, actuating an automated mode is permitted only when inside the ODD, and it will be deactivated prior to leaving the ODD or as a result of the vehicle operator taking control again.

FD_6: PERFORM DEGRADED MODE WITHIN REDUCED SYSTEM CONSTRAINTS

Automated driving system operation in degraded mode is actuated as nominal capabilities with new limits. Multiple degraded modes are possible. The limitations should be defined such that the degraded mode can be stated as safe. Therefore, it may be necessary to avoid a permanent operation. A well-defined timeframe for an additional reaction is required.

2.1.7 Minimal Risk Conditions and Minimal Risk Maneuvers

A minimal risk maneuver (MRM) is the system's capability of transitioning the vehicle between minimal risk conditions (MRC). The concept of MRCs and MRMs derives from the principles of ISO 26262 and is defined as an operating mode (in the case of a failure) of an item with a tolerable level of risk. In terms of ISO 26262 [ISO 26262, 2018, p. 9], an MRM is an emergency operation to reach an MRC – referred to as a safe state. Contrary to the commonly used definition of an MRC, which describes only a standstill, this publication expands the definition to also include degraded operation and takeovers by the vehicle operator. Final MRCs refer to MRCs that allow complete deactivation of the automated driving system, e.g. standstill or takeover by the vehicle operator. Figure 11 and Figure 12 visualize this general principle:

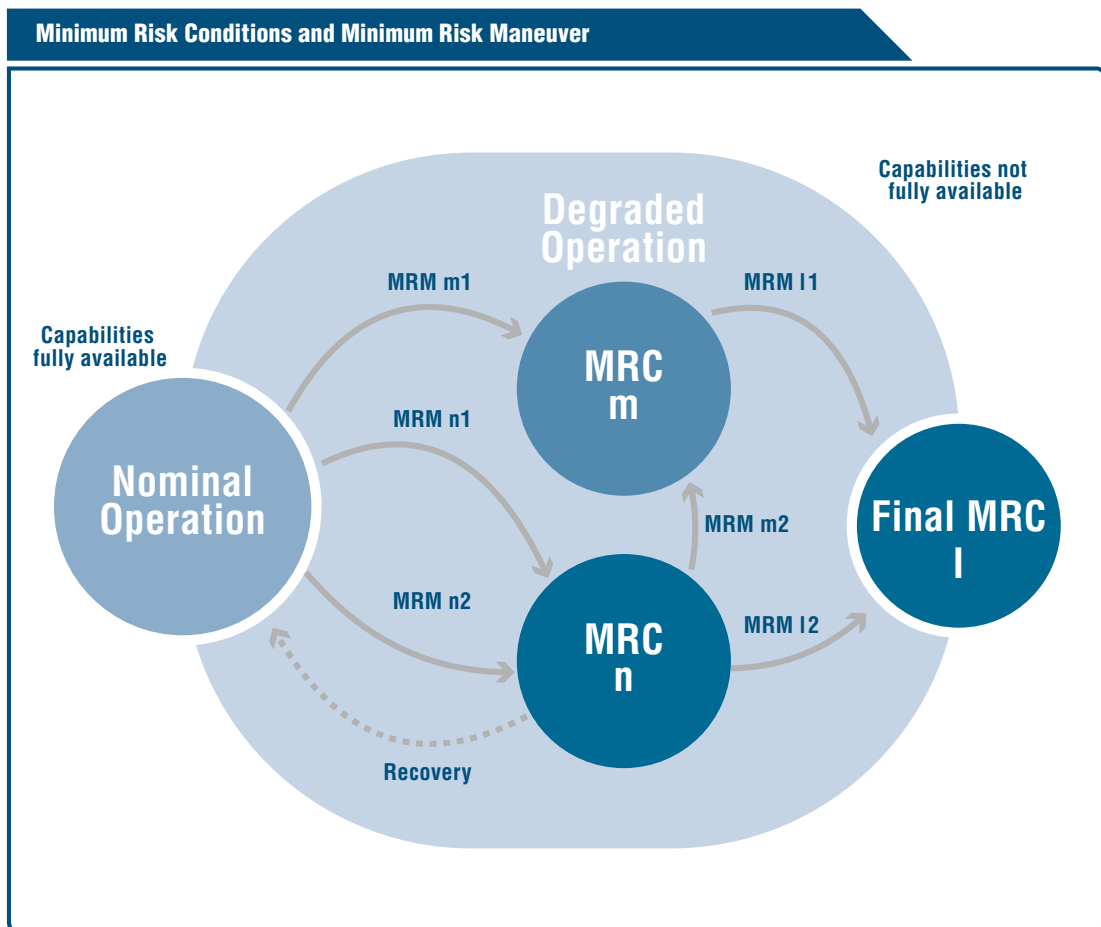


Figure 11: Minimal Risk Conditions and Minimal Risk Maneuvers

The purpose of the MRM is to bring the vehicle to level of tolerable risk given the level of functionality that is possible for it to support. Due to the complexity of automated systems and risk-influencing conditions, several MRCs and MRMs could be conducted consecutively. If not all fail-safe capabilities are available, the system will be in degraded mode and the remaining fail-degraded capabilities will reach and maintain a minimal risk condition by executing an appropriate minimal risk maneuver. The degraded mode is a time-limited operational domain in which the frequency of its occurrence should be reduced whenever possible. The acceptable time for the degraded mode depends on the remaining capabilities in the current system. In general, this fundamental argument is derived from ISO 26262. One principle concept of ISO 26262 is

that every order of magnitude in reducing the frequency of possible harm lowers the safety integrity level. Table 3 defines the conditions between which the MRM will allow for safe transition. Specific examples are outlined in Appendix A.

Minimal Risk Conditions			
ID	MRC	Definition	Possible Reasons
MRC_1	Takeover by the Vehicle Operator	The vehicle operator has completely taken over the driving task.	<ul style="list-style-type: none"> Known future limitation in ODD Limitations or the vehicle operator (if present) has initiated takeover It is detected that degraded performance is not available (FD_2)
MRC_2	Limited Operation	Vehicle is still operational within reduced capabilities. There could be several limited operation conditions depending on the functional definition and remaining capabilities.	<ul style="list-style-type: none"> Derivations of nominal state, reduced capabilities
MRC_3	End of Operation	This condition describes a vehicle state that allows safe deactivation of the function.	<ul style="list-style-type: none"> Severe system failures, loss of capabilities, missing driver takeover

Table 3: Minimal Risk Conditions

The following MRMs are proposed:

Proposed MRMs			
ID	MRM	Definition	Target Condition
MRM_1	Transition Demand	Request takeover by the vehicle operator	MRC_1 Takeover by the Vehicle Operator
MRM_2	Limit Function State	Transition to limited operation. Depending on the MRC and the actual state, several MRM variants are possible.	MRC_2 Limited Operation
MRM_3	Comfort Stop	Comfortable transition to end of operation	MRC_3 End of Operation
MRM_4	Safe Stop	Due to severe failures, a fast but safe transition to end of operation is necessary	MRC_3 End of Operation
MRM_5	Emergency Stop	In case of sufficient rare severe system failures, an emergency stop is initiated to minimize risk, and so that the End of Operation condition can be reached.	MRC_3 End of Operation
RECOVERY	Recovery	Limitations of capabilities are resolved and therefore nominal state is reached again	Nominal State

Table 4: Proposed MRMs

Using the list of MRMs in Table 4, the potential failure modes should be reflected within the overall system. Several analysis methods should be applied, which may also include failure analysis techniques such as FMEA or DFMEA. Additional analysis may allow for introspection into the intended use of the system but also its misuse. The outcome of such analysis measures is to define all desired safe states for each component and to characterize how these safe states enable the MRCs and MRMs of the integrated system.

2.2 Elements for Implementing the Capabilities

In addition to the different SAE levels and ODD definitions, there are many possibilities when implementing automated driving. The development examples introduced in Appendix A can be considered possible real-world implementations that fulfill the capabilities introduced in Section 2.1.3. This chapter discusses how to fulfill the fail-safe and fail-degraded capabilities with real-world elements in a generic manner. Section 2.2.1 discusses possible implementations of each capability by introducing elements. Section 2.2.2 then discusses each element in detail.

2.2.1 Implementing the Capabilities

The following describes the capabilities in relation to their associated elements. Elements are designated in italics.

2.2.1.1 FS_1: DETERMINE LOCATION

It must be ensured that the automated driving system operates only within specified system limits if the intended use of the automated driving system is restricted. Therefore, the automated driving vehicle should be located adequately, using fused environmental sensor information from *Sensor Fusion* algorithms (see Section 2.2.1.4). To achieve appropriate *Localization*, it may be necessary to link additional a-priori information from outside of the onboard perception's performance (e.g. via map information, referencing detected events to a unique coordinate system), either in range or in interpretation, with the automated driving vehicle. *Localization* may consider information from *Egomotion* to predict whether the automated driving vehicle is about to exceed an ODD limit. A precondition for activating the automatic driving system, this prevents operator misuse of using the automated driving system outside the ODD.

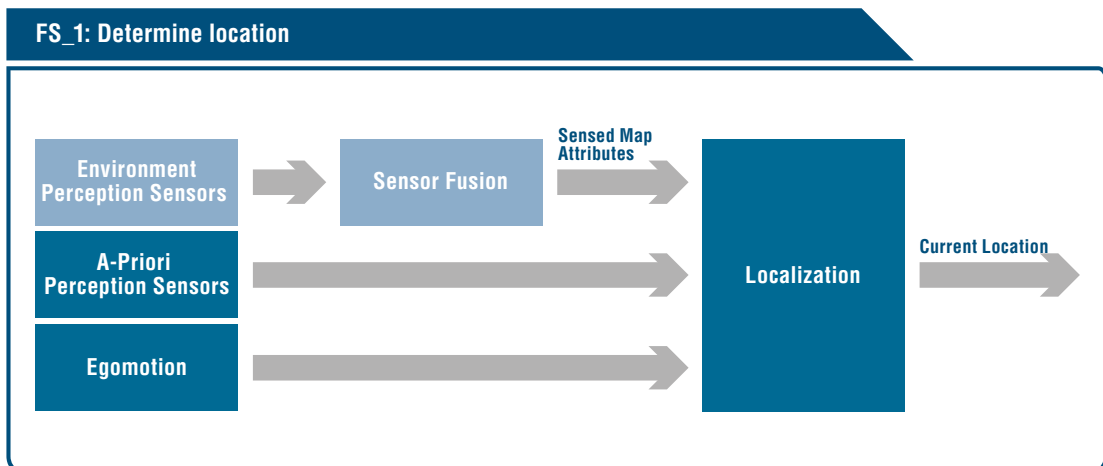


Figure 12: FS_1: Determine location

2.2.1.2 FS_2: PERCEIVE RELEVANT STATIC AND DYNAMIC OBJECTS IN PROXIMITY TO THE AUTOMATED VEHICLE

All entities that an automated driving system requires to account for its functional behavior should be perceived, pre-processed and provided safely. The highest priority is placed on entities with the highest associated risk of collision. Sample entities include dynamic instances (e.g. (vulnerable) road users and characteristics of the respective movement), static instances (e.g. road boundaries, traffic guidance signals), and obstacles exceeding a critical size. The main element for this is the *Sensor Fusion* element, where the different inputs from onboard sensors, *Localization*, *Egomotion* and optional *V2X* information are used to generate the present world model. *Traffic Rules* might be used to optimize the content of the world model. The semantic knowledge of the perceived environment is important for later interpretation, prediction and planning. Consequently, the automated driving system knows what has been detected and where. The definition of relevant objects depends on the item definition.

The detection of static instances can be supported by the *Localization* element, which provides a map and a location of the automated driving vehicle on the map. With entities marked on the map, this supports the *Sensor Fusion* algorithm with another independent sensor.

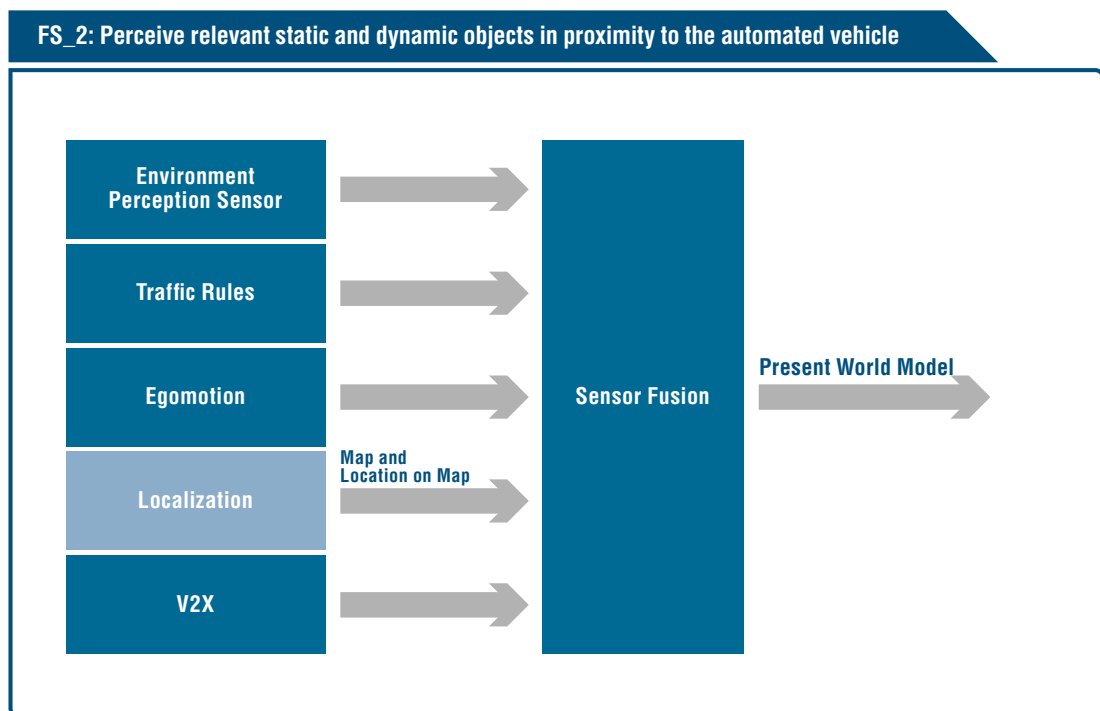


Figure 13: FS_2: Perceive relevant static and dynamic objects in proximity to the automated vehicle

2.2.1.3 FS_3: PREDICT THE FUTURE BEHAVIOR OF RELEVANT OBJECTS

The present world model recorded as the output of FS_2 may not suffice as an input for the safe and lawful creation of a driving plan (FS_4). It should therefore be extended to reflect not only the current but also the predicted future state of the world model in order to generate a complete description of the dynamic driving situation or “scene”. It should also consider the intention of other dynamic objects and objects possibly not visible due to occlusions as the basis for predicting future motions. Finally, current environment conditions such as low road friction and reduced sensor performance (fog, mist, heavy rain) must also be taken into consideration.

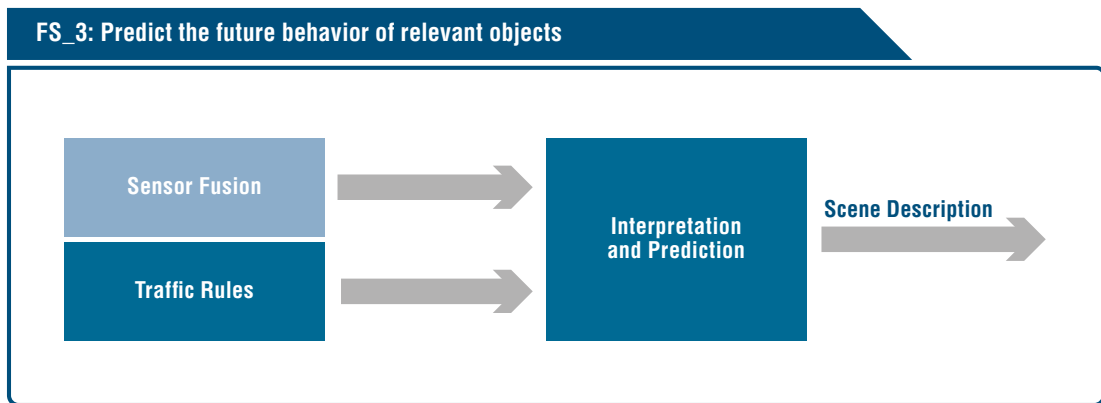


Figure 14: FS_3: Predict the future behavior of relevant objects

2.2.1.4 FS_4: CREATE A COLLISION-FREE AND LAWFUL DRIVING PLAN

Creating a collision-free and lawful driving plan may consider several elements. For example, the vehicle should first have accurately sensed its environment and performed *Localization* before any driving plan can be created. With *Localization* performed and an accurate world model provided, the vehicle should consider the safety-relevant (vulnerable) road users in the world model and what their tracked motion suggests from an *Interpretation and Prediction* standpoint. This provides a baseline for possible reasonable assumptions that can be considered regarding detected (vulnerable) road users.

Machine-interpretable traffic rules are also necessary, as the automated vehicle should obey traffic rules in order for the *Drive Planning* element to produce a lawful driving plan, unless exceptions are necessary to prevent collisions.

The automated vehicle's *Egomotion* may also be considered, as physical properties of both the Earth and the automated vehicle limit the set of possible maneuvers. Finally, the *ADS Mode Manager* should be considered so that *Drive Planning* is aware of whether it is in normal operation mode or a degraded/ minimal risk condition mode.

Figure 15 illustrates the elements required:

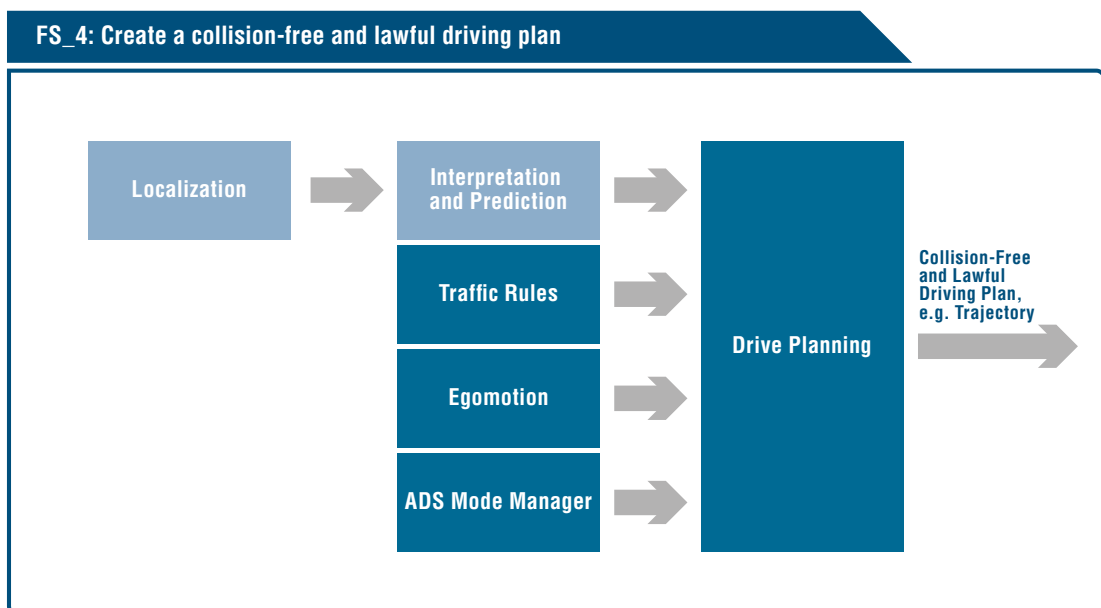


Figure 15: FS_4: Create a collision-free and lawful driving plan

2.2.1.5 FS_5: CORRECTLY EXECUTE AND ACTUATE THE DRIVING PLAN

Once *Drive Planning* has created a collision-free and lawful driving plan, *Motion Control* and *Motion Actuators* may also consider the current *Egomotion* to translate the *Drive Planning* element's requested trajectory into the physical motion for the vehicle's motion actuators (e.g. steering, braking or powertrain).

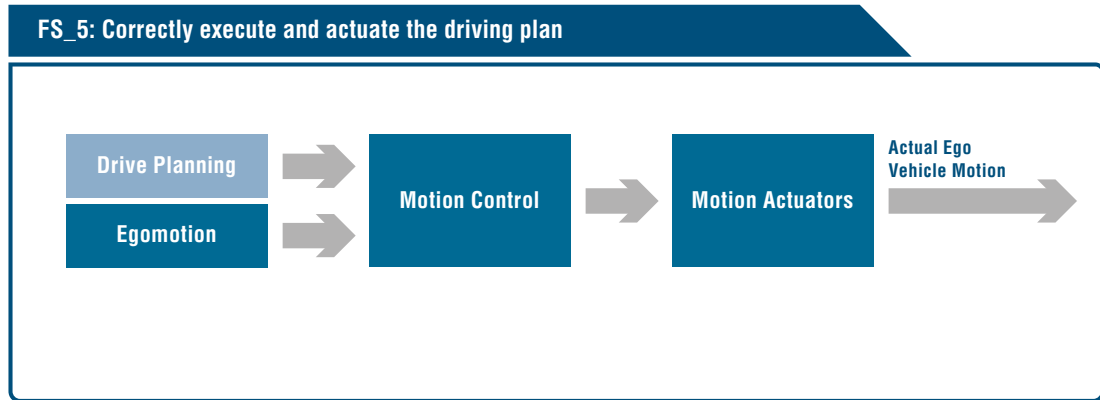


Figure 16: FS_5: Correctly execute and actuate the driving plan

2.2.1.6 FS_6: COMMUNICATE AND INTERACT WITH OTHER (VULNERABLE) ROAD USERS

Other (vulnerable) road users need to be aware of future actions that the automated driving vehicle is going to take. As with manual driving, means of communication include visual and acoustic indicators to fulfill traffic rules. V2X or other types of interaction could be a means of communication as well.

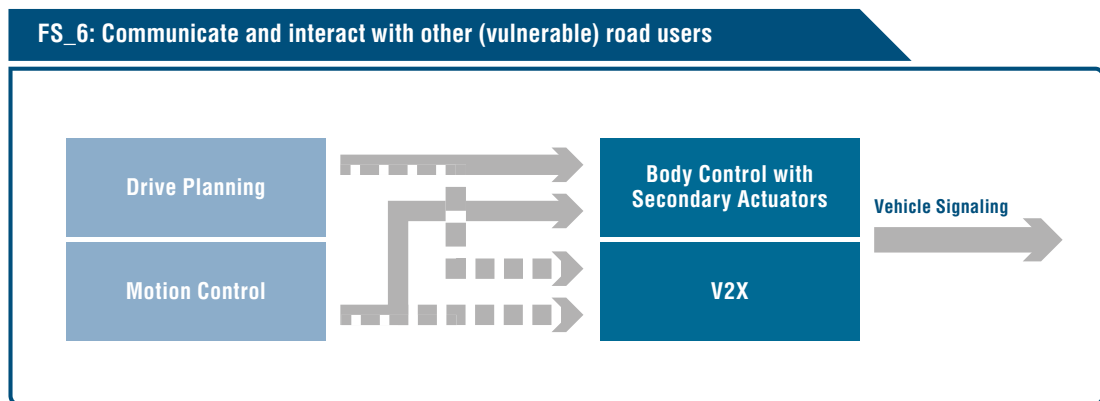


Figure 17: FS_6: Communicate and interact with other (vulnerable) road users

2.2.1.7 FS_7: DETERMINE IF SPECIFIED NOMINAL PERFORMANCE IS NOT ACHIEVED

Monitors should be in place to detect defined nominal performance boundaries at the element or system level with sufficient time to ensure a safe reaction. The boundaries are derived following the iterative approach described at the beginning of Chapter 2.

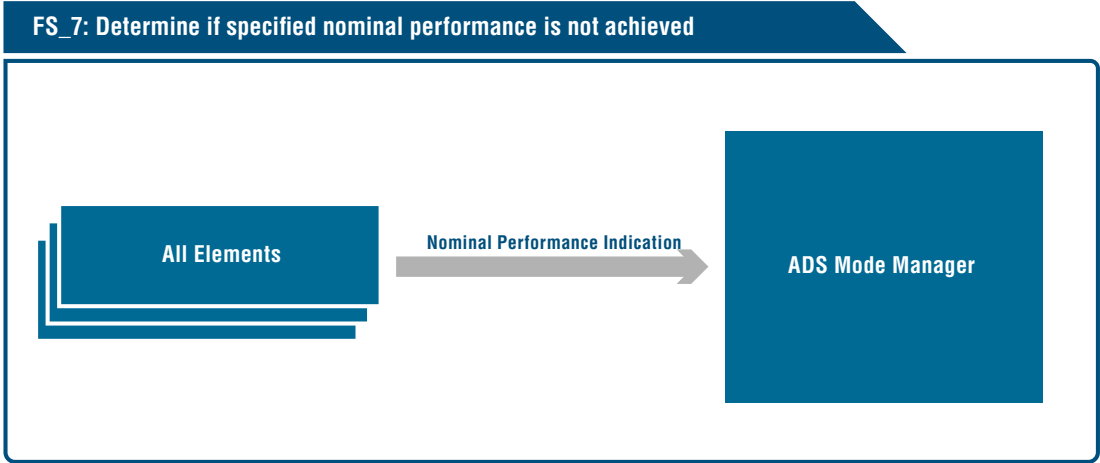


Figure 18: FS_7: Determine if specified nominal performance is not achieved

2.2.1.8 FD_1: ENSURE CONTROLLABILITY FOR THE VEHICLE OPERATOR

The role of the vehicle operator depends on the intended SAE level for the automated driving system. In an SAE L3 automated driving system, the vehicle operator can turn their attention away from the driving task. In this case, the system is responsible for maintaining vehicle control to allow the vehicle operator to re-adjust to concentrating on the driving task and regain situational awareness when a system takeover request is imminent. Therefore, the automated driving system should continuously monitor the vehicle operator for possible distraction or mode confusion. This is also valid for a vehicle equipped with both an SAE L4 automated driving system and a manual driving mode.

The controllability for the vehicle operator in an SAE L4 automated driving system without a manual driving mode may be limited to an ability to access an emergency stop actuator when the user recognizes a hazard or upon realization that the ODD is being exited. The vehicle operator may be an entity outside of the vehicle in question in cases where the system foresees remote control. To ensure controllability for a local or remote vehicle operator, the *ADS Mode Manager* should sense an the vehicle operator's (driver or a teleoperator) wish to control the vehicle and react to this.

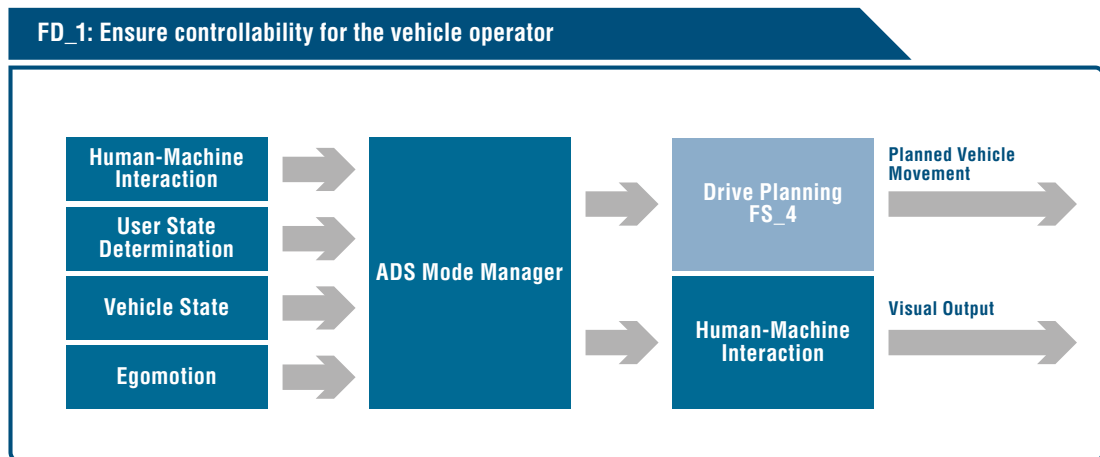


Figure 19: FD_1: Ensure controllability for the vehicle operator

2.2.1.9 FD_2: DETECT WHEN DEGRADED PERFORMANCE IS NOT AVAILABLE

A *Monitor* should check whether degraded performance is available even if it is not actively being used in the nominal drive mode. Thus, the *Monitor* has to continuously check the availability of the degraded mode that can be implemented in several ways.

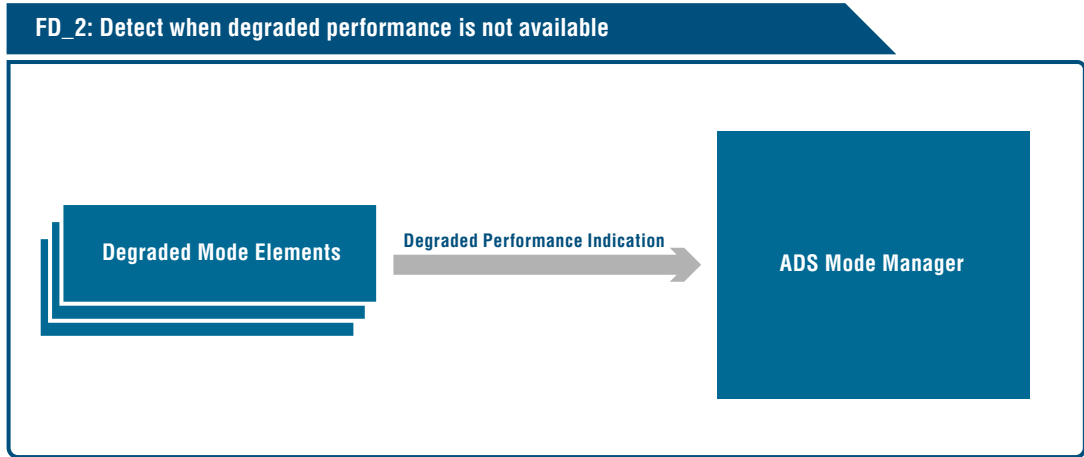


Figure 20: FD_2: Detect when degraded performance is not available

2.2.1.10 FD_3: ENSURE SAFE MODE TRANSITIONS AND AWARENESS

A safe mode transition is performed by the *ADS Mode Manager* that collects all necessary information needed to decide whether to change a mode. This includes information from the *Monitor* about electric failures, performance issues, or the vehicle and vehicle operator states. The second step after collecting all necessary information is to safely switch between modes. Deactivation is possible only if the vehicle operator is back in the loop for controlling all functions of the vehicle or if the vehicle is in a safe state.

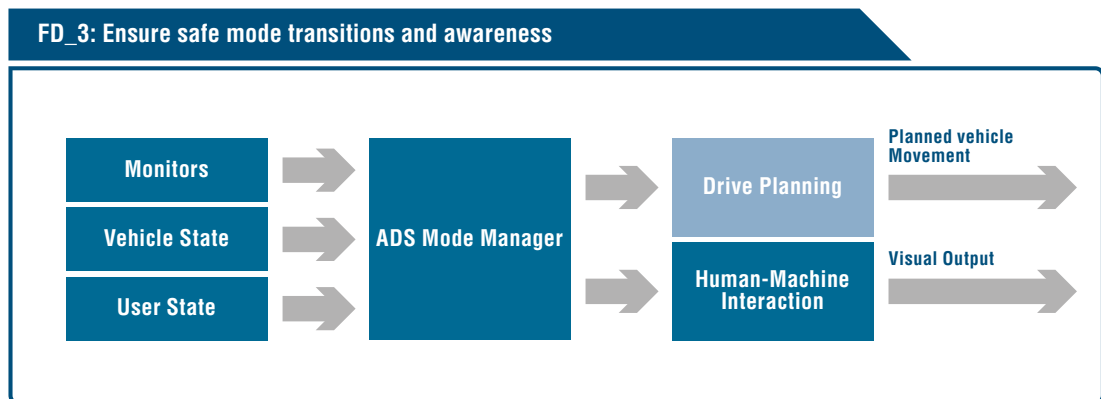


Figure 21: FD_3: Ensure safe mode transitions and awareness

2.2.1.11 FD_4: REACT TO INSUFFICIENT NOMINAL PERFORMANCE AND OTHER FAILURES VIA DEGRADATION

The systems reaction to insufficient nominal performance (see FS_7) should be defined, and the system should react properly even in case of failures. This task is performed by the *ADS Mode Manager*. It is triggered by the *Monitor* and initiates the defined reaction for the corresponding trigger. This can affect just a few elements or almost the complete automated driving system, depending on the severity of the failure.

In summary, the demanding task is to decide which degradation to choose with all the different combinations of failures that could occur.

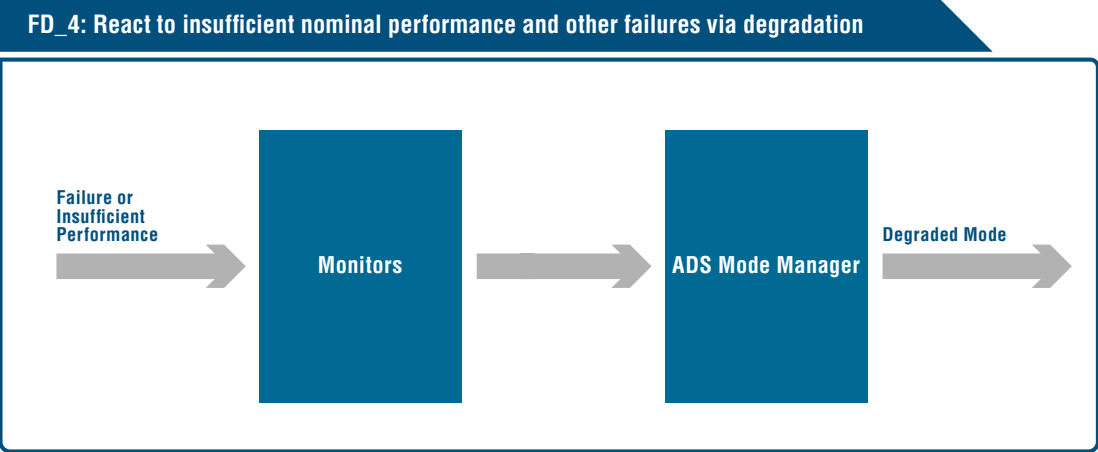


Figure 22: FD_4: React to insufficient nominal performance and other failures via degradation

2.2.1.12 FD_5: REDUCE SYSTEM PERFORMANCE IN THE PRESENCE OF FAILURE FOR THE DEGRADED MODE

Given by the targeting MRC, which is decided by the *ADS Mode Manager* and other inputs that may contain degraded constraints (e.g. reduced perception range), *Drive Planning* should be able to generate collision-free and lawful vehicle movement to achieve the corresponding MRC with reduced system performance. Reduced system performance capability spans from the loss of some functionality of the automated driving system to the request to discontinue automated driving safely. It also includes operator information about the mode change (e.g. takeover request) performed via human-machine interaction. Failure that results in reduced comfort is not included in the scope of this document.

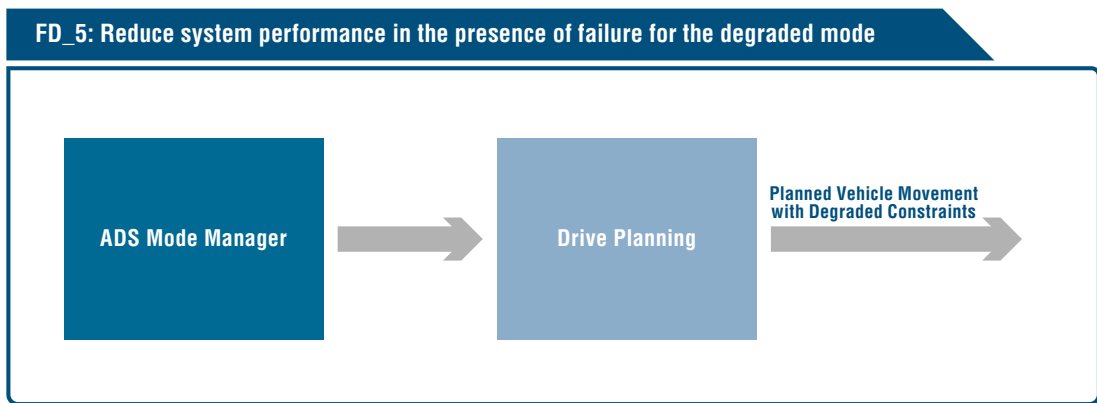


Figure 23: FS_5: Reduce system performance in the presence of failure for the degraded mode

2.2.1.13 FD_6: PERFORM DEGRADED MODE WITHIN REDUCED SYSTEM CONSTRAINTS

Automated driving system operation in degraded mode is actuated as nominal capabilities with new limits are defined. An MRM should be carefully defined to achieve the MRC. In this case, the automated driving system should be able to perform the DDT within a well-defined timeframe.

The elements for degraded mode depend on the item definition. Depending on the degraded mode of the automated driving system, adequate HMI operations should be implemented.

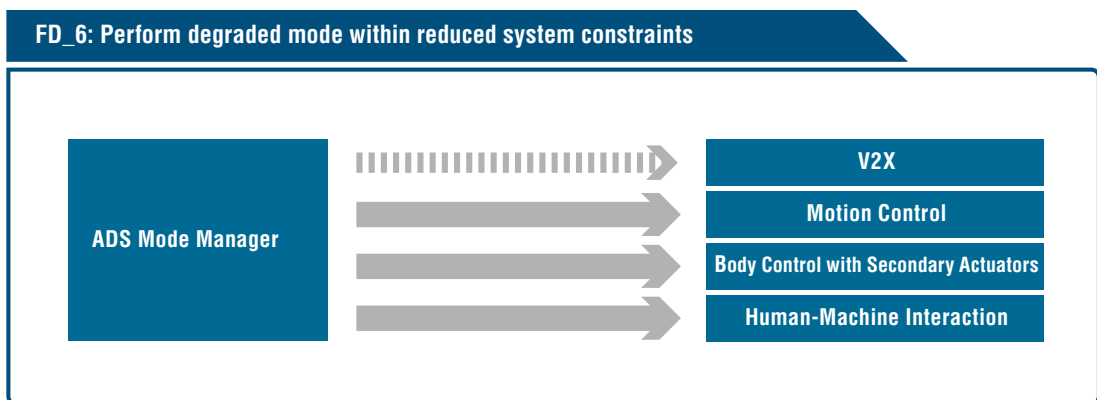


Figure 24: FD_6: Perform degraded mode within reduced system constraints

2.2.2 Elements

Fail-safe and fail-degraded capabilities are implemented by the elements described in detail below. The general capability regarding cybersecurity as described in Section 2.1.4 has to be considered for each element. However, the specific cybersecurity measures are not described in each element, as this depends on the overall cybersecurity architecture.

2.2.2.1 ENVIRONMENT PERCEPTION SENSORS

The environment perception sensors cluster should capture all relevant external information to create a world model. Entities to detect are, but are not limited to, infrastructure defining the allowed area of driving, (vulnerable) road users, obstacles, traffic signs and acoustic signals.

Sensor types: As of today, a single sensor is not capable of simultaneously providing reliable and precise detection, classifications, measurements, and robustness to adverse conditions. Therefore, a multimodal approach is required to cover the detectability of relevant entities. In more detail, a combination of the following technologies shall provide suitable coverage for the given specific product:

CAMERA

Sensor with the highest extractable information content as sensor captures visible cues similar to human perception. Main sensor for object/feature type classification. Limited precision in range determination, high sensitivity to weather conditions.

LIDAR

High-precision measurement of structured and unstructured elements. Medium sensitivity to environment conditions.

RADAR

High-precision detection and measurement of moving objects with appropriate reflectivity in radar operation range, high robustness against weather conditions.

ULTRASONIC

Well-established near-field sensor capable of detecting closest distances to reflecting entities.

MICROPHONES

Public traffic uses acoustic signals to prevent crashes and regulate traffic, e.g. on railway intersections. Thus, devices capturing acoustic signals are required for automation levels where the systems need to react to these.

Sensor sets need to be capable of detecting sensing degradation, such as sensor blindness, de-calibration or misalignments. Possible methods for this could be based on sensor-specific measures or cross-sensor comparisons and calibration methods.

SENSOR ARRANGEMENT

The design of the sensor cluster needs to cover the ODD of the respective functionality. For example, a sensor cluster designed for a system on highways needs to cover ranges and precision levels that are different to those of urban scenarios. The detectability of external entities strongly depends on the material these are made of. This publication considers a combination of at least two, if not three, different measurement technologies to implement susceptibility of the sensor cluster to all relevant elements in the real world. This approach further enables the simultaneous capturing of the majority of elements with at least two different measurement technologies. Subsequent processing steps are thus enabled to provide detection rates superior to individual sensor detection rates.

However, errors and perception failure may still occur even when an iterative design approach is followed and ISO 26262 recommendations are complied with. In the unlikely event of severe sensor degradation or E/E faults, the sensor arrangement needs to be laid out such that it enables the safe capturing of relevant elements in degraded mode until the safe state is reached.

2.2.2.2 A-PRIORI PERCEPTION SENSORS

2.2.2.2.1 HD MAP

HD MAP AS A RELIABLE SENSOR

An in-vehicle map has never played a safety-related role as it could do in automated driving. For a relatively long period of time, the capabilities of onboard sensors alone will be insufficient to meet the high reliability, availability and safety requirements of the automated vehicle system in certain situations. A HD map is therefore necessary as a reliable off-board sensor containing carefully processed a-priori information to “detect” features that are not easily detectable by on-board sensors or to provide a redundant source of information for on-board sensors, including location-based ODD determination, environment modeling in adverse conditions and precise semantic understandings in complex driving situations. In situations where on-board sensors cannot reliably detect features, the HD map can be utilized as a more reliable redundant source of information.

RELIABLE MAP ATTRIBUTES AND HOW TO IDENTIFY RMA SETS

Multiple map attributes are utilized in location-based ODD determination, such as lane markings, road markings, traffic signs, light poles, guardrails or artificial markers. However, some attributes are not always “reliable” to detect due to reasons including occlusion, abrasion or frequent changes. Therefore, reliable map attributes (RMAs) should be detected correctly in safety-relevant use cases, so that collectively they can meet the low location-based ODD determination false positive rate requirement.

RMAs should have the following properties:

- Fused with on-board sensor inputs, a combination of RMAs should be a sufficient condition to infer that the automated vehicle is reaching the boundary of the ODD.
- RMAs should be reliably detectable by onboard sensors within the ODD.
- RMAs should be observed with a relatively low real-world rate of change, so that the RMA failure rate can be controlled to an acceptable risk level.
- The quality and freshness of RMAs should be verifiable within an acceptable time delay.

As stated above, only a subset of all map attributes is related to safety. The method for abstracting the complete list of RMAs is project-specific (these include but are not limited to the development examples outlined in Section 1.2).

RMA FAILURE MODES AND CORRESPONDING MEASURES

Due to its nature of being offline but not processed in real time, a HD map has the advantage of being less probabilistic compared to onboard sensors. However, this also results in the limitations of a HD map when employed in safety-related use cases. RMA failure occurs due to deviations between the map and reality, possibly arising from:

- Errors introduced during source data collection, map creation and distribution processes
- Errors introduced due to real-world changes, which can further be classified as:
 - **INTENDED CHANGES:** Typically by a local road authority (e.g. planned road construction)
 - **UNINTENDED CHANGES:** Typically due to external forces or normal wear (e.g. a piece of guardrail is damaged in a collision and not recovered before the next road maintenance)
 - **MALICIOUS CHANGES:** Typically due to an unauthorized/malicious action (e.g. unauthorized removal of a speed limit sign)

The above errors should be addressed appropriately to ensure that the automated driving system is able to reach an accepted risk level. Failures relating to procedural deficiency can be avoided by a quality assurance system including but not limited to those articulated in established map quality standards (e.g. ISO 19157, ISO/TS 19158, TS 16949). Failures relating to planned road changes can be avoided by incorporating road change plans from a road authority into the map updating process. Meanwhile, as indispensable public information, road construction and maintenance plans should be fully transparent and easily accessible by all map providers. Errors as a result of real-world changes are difficult to monitor and control, thus they should be carefully analyzed. Changes of RMAs can be divided into two categories based on the impact that they have for the use case of an automated vehicle system:

MINOR CHANGES

do not impede or exceed the specification tolerance for the given RMA-associated functionalities in safety-relevant use cases (e.g. dents in guardrails or lane markings with small parts missing)

MAJOR CHANGES

significantly reduce the detection rate and exceed the specification tolerance for the given RMA and may lead to localization errors (e.g. missing guardrails for a certain distance due to a severe crash)

Therefore, failures of RMAs due to major changes should primarily have an impact on the failure rate of location-based ODD determination. Several measures can be implemented to mitigate random failures. First, RMAs should be carefully chosen so that the possibilities of unplanned major changes are limited and can be statistically proven. Second, an effective mechanism for map updating or maintenance is critically important. A map updating or maintenance platform that comprises sensor data collected from multiple inputs, including but not limited to survey car fleets, massively deployed intelligent vehicles (e.g. vehicles with the ability to collect sensor data), high resolution satellite images and/or road infrastructures with surveillance sensors, can effectively detect the random road changes and lower the risk of random RMA failures.

OTHER SAFETY CONSIDERATIONS:

Map modification after initial creation is a mandatory map processing step in certain regions of the world. (NPC, 2017). Safety-relevant content should not lose reliability as a result of these measures. A sound safety analysis and eventual measures are required to continue to enable the use of maps in the vehicle.

Furthermore, malicious changes to map content need to be prevented. As a HD map is an off-board sensor, cybersecurity should be considered from creation through to storage and distribution. This is discussed in greater detail in Section 2.1.5.

2.2.2.2.2 GNSS

Absolute GNSS position contributes to the automated vehicle system safety. Consequently, not only accurate but also trustful absolute GNSS positions are required for location-based ODD determination. A time window of GNSS position validity with integrity should be defined, as various levels of accuracy, integrity and availability will be in place while the automated vehicle is in operation. Continuity metric is no longer the main parameter of GNSS-based positioning with integrity.

A higher availability of GNSS-based positioning can be achieved by implementing multi-frequency and multi-constellation GNSS antennas and receivers, which is a prerequisite for interoperability and compatibility between GNSS constellations and radio frequency signals.

GNSS sensor functionality relies on the direct visibility of satellites. Consequently, GNSS-based positioning cannot have high continuity and availability due to environmental obstructions such as bridges or tunnels. In good GNSS conditions, position accuracy with high integrity, detection of loss of lock and fast convergence times after GNSS outages are therefore substantial for an automated driving system. Reaching accuracies and integrity performance metrics simultaneously is enabled by GNSS receivers that can utilize data received from an adequate number of satellites (e.g. 10 or more satellites) and additional data from correction services. These services need to implement fast processing, frequent updates and dedicated correction sets to support a best possible GNSS positioning algorithm.

A further aspect to cover is the assessment of new signals with respect to interferences in ARNS/RNSS bands or other interferers or jammers that could harm GNSS positioning performance. Integrity can be given only if spoofing is addressed at the GNSS component level.

2.2.2.3 V2X

V2X may provide valuable information to the automated driving system. However, safety and security aspects should be considered to ensure a proper integrity. In addition, the automated driving system should operate safely in conditions where V2X is not available.

An example for this could be providing redundancy for the detection of traffic signal state that else could be detected only by camera. There is currently no redundant method for detecting traffic signal states without additional communication from the infrastructure.

2.2.2.4 SENSOR FUSION

There is a variety of sensor fusion algorithms, each of which requires individual analysis with respect to hardware or software error robustness or input data error sensitivity, for instance. Thus, a carefully selected approach incorporating inductive, deductive and data-driven iterative design procedures, for example, should be followed.

Generally, input checks that determine the plausibility of individual sensor data, fusing multiple weighted input sources, and accumulating sensor data are possible strategies. Hardware and software diversity for the implementation of functionalities with the highest required error robustness should be considered.

While individual sensors can provide information about their current detection capabilities and range, sensor fusion can add substantial value in determining the current horizon of full sensor cluster perception, which may help to monitor the actual sensor performance. Regarded as a cross-referencing mechanism, sensor fusion can enable the detection of individual sensor limitations that are not detectable by the individual sensor itself.

2.2.2.5 INTERPRETATION AND PREDICTION

Prediction is an essential element for the realization of an automated driving system. The automated vehicle should behave almost like a manually driven vehicle, so that its behavior is predictable to other participants and does not disturb the traffic flow. Actual traffic is based on knowledge, rules and experience and how (vulnerable) road users will usually act next. To adapt this behavior for the automated vehicle, the vehicle needs a prediction based on a reliable interpretation of the situation.

Interpreting the current environment enables the prediction of other (vulnerable) road users. It is not possible to base safety on probabilistic calculations without measurable or common properties. Human road users in particular can make irrational decisions. On the other hand, if a function is provided that is always planning for the worst-case scenario, it may include actuations which provide risks to the overall system in other ways that are unacceptable for the goal of attaining the capabilities.

A solution may consider a combination of the following properties:

- Predict only a short time into the future. The likelihood of an accurate prediction is indirectly related to the time between the current state and the point in time it refers to (i.e. the further the predicted state is in the future, the less likely it is that the prediction is correct).
- Rely on physics where possible, using dynamic models of (vulnerable) road users that form the basis of motion prediction. For example, a vehicle driving in front of the automated vehicle will not stop in zero time on its own. Thus, a classification of relevant objects is a necessary input to be able to discriminate between various models.
- Predictable drive planning should consider the compliance of other (vulnerable) road users with traffic rules to a valid extent. For example, the automated vehicle should cross intersections with green traffic lights without stopping, relying on other (vulnerable) road users to follow the rule of stopping at red lights. In addition to this, foreseeable non-compliant behavior to traffic rules (e.g. pedestrians crossing red lights in urban areas) needs to be taken into account, supported by defensive drive planning.
- Situation prediction to further increase the likelihood of (vulnerable) road user prediction being correct. For example, the future behavior of other (vulnerable) road users when driving in a traffic jam differs greatly to their behavior in flowing traffic.

The Interpretation and Prediction system should understand not only the worst-case behavior of other (vulnerable) road users, but their worst-case reasonable behavior. This allows the Interpretation and Prediction system to make reasonable and physically possible assumptions about other (vulnerable) road users. The automated driving system should make a naturalistic assumption, just as humans do, about the reasonable behavior of others. These assumptions need to be adaptable to local requirements so that they meet locally different „driving cultures“.

2.2.2.6 LOCALIZATION

An automated driving system must reliably know its location as precisely as required depending on the system design. Different approaches can be applied when determining an automated driving vehicle's position on a HD map, including:

GNSS-BASED LOCALIZATION

This approach consists of GNSS, odometry and correction services to achieve precise global coordinates, and matching GNSS measurements to an HD map to obtain a relative position on the map.

ENVIRONMENT-PERCEPTION-SENSOR-BASED LOCALIZATION

Based on a rough global coordinate obtained by GNSS and odometry, this approach matches real-world features (such as natural or artificial landmarks) or point clouds detected by Environment Perception Sensors with respective features or point clouds on an HD map to localize the automated driving system on the map.

Both localization approaches are subject to errors caused by performance limitations of the sensors involved, or sensor processing chains (or by real failures in either of these), or by multiple elements involved in the procession.

Localization needs to be implemented such that is robust against at least single, simple and timebound sensor performance issues. This is necessary due to the nature of the sensors (e.g. limitations of GNSS in tunnels, light conditions affecting vision sensors, etc.). Therefore, a sound safety analysis of involved inputs with relevant failure modes, performance limitations, availabilities, and respective effects on the position estimation needs to be carried out. As a single localization approach may not be sufficient for all relevant situations of an automated driving system, a redundant system incorporating both of the above localization approaches to provide seamless localization with the required integrity can increase localization performance.

2.2.2.7 ADS MODE MANAGER

The ADS Mode Manager has to fulfill the task of safely changing between manual and different automated driving modes. For the activation of an automated driving mode, this means obtaining all information to check whether all prerequisites such as the ODD are fulfilled (e.g. whether the automated vehicle is on the correct road type, check the weather conditions, etc.). Required information can be transferred from a backend to the vehicle, directly measured, calculated or derived from statistics.

There are many reasons for requesting that the automated driving system be deactivated. These include requests from the vehicle operator or from a monitor, or as a result of leaving the ODD or a monitor being unavailable. If such a request or reason is perceived, the relevant MRC should be targeted (see Section 2.1.7).

Reasons for changing modes may be triggered based on the vehicle state, user state determination or monitors. For example, a deactivation request arising from the vehicle state may be a fuel gauge, tire pressure or other vehicle systems. Examples arising from the user state include the belt status or vehicle operator attention. Based on the information from one or more monitors, the ADS Mode Manager has to decide whether to change to a degraded mode or issue an MRM to reach an MRC. However, these examples are strongly linked to the specific automated driving system.

Checking whether the automated vehicle is inside or outside of the ODD is a complicated task, because an ODD definition covers a widespread set of requirements. Being able to sense all of them is crucial for activation and deactivation. Table 5 lists all combinations of errors that may occur in the event of erroneous detection:

		Reality	
		Vehicle within ODD	Vehicle outside ODD
System Output	Vehicle within ODD	True Positive (TP)	False Positive (FP)
	Vehicle outside ODD	False Negative (FN)	True Negative (TN)

Table 5: Determining the Vehicle's Location

Only the false positive combination is safety-related. The system erroneously detects being inside the ODD when in reality the vehicle is outside of the ODD. The behavior and consequences of automated driving operation outside of the ODD are by definition not safe enough. Therefore, appropriate safety measures are required to ensure the safe detection of ODD areas and limits. Being inside the ODD but detecting being outside will result only in deactivation, which is carried out in a safe manner.

2.2.2.8 EGOMOTION

Egomotion describes the actual state of the car in terms of yaw rate, longitudinal acceleration, lateral acceleration and more. Further values describing the vehicle state may include vehicle speed or slip angle. Some of the data can directly be measured using an inertial measurement unit, wheel ticks or derived from other sensors such as cameras. Other Egomotion data cannot be read directly and so can be estimated with the aid of other sensors using mathematical models, for example. Because Egomotion is an input for several other elements, it should be fail-degraded to fulfill the capabilities. There are numerous ways to achieve this, so implementations will vary considerably.

2.2.2.9 DRIVE PLANNING

Creating a driving policy that can drive in a collision-free manner without compromising comfort or traffic flow is a challenge in automated driving. A promising solution lies in defining formal rules, such as the examples of a theoretical approach (Shalev-Schwartz, Shammah, & Shashua, 2018) and hierarchical sets of rules (Censi, et al., 2019). These theoretical rules must still be applied to the complexities of real-world mixed traffic, and the resulting evaluation of the effect on traffic must still take place.

These formal rules may include, but are not limited to, the following examples. They should be followed during implementation for all drive modes.

EXPLICIT TRAFFIC RULES

- Conform to all applicable traffic rules within the ODD that the automated vehicle is operating in, taking regional differences in traffic rules into special consideration. Roads, signaling elements and other examples of infrastructure are the physical embodiment of the explicit traffic rules, e.g. a STOP sign or double solid lane marking.

IMPLICIT TRAFFIC RULES

- Maintain a safe longitudinal and lateral distance from other objects to avoid collision.
- Right of way is given, not taken. Following the safety-first principle, non-compliance of other (vulnerable) road users with traffic rules should be expected and dealt with defensively.
- Be cautious in areas where other (vulnerable) road users may be occluded. If information from Interpretation and Prediction, the ODD, or other sources indicates that there is a potential for occluded objects, the automated vehicle should be prepared for the possible sudden appearance of other vulnerable road users such as pedestrians.
- If it is possible to perform a legal and safety-assured maneuver to evade a potentially unsafe situation, then the automated vehicle should do so.
- If it is not possible to evade an unsafe situation without violating traffic rules, then it may be possible for the automated vehicle to violate traffic rules while making a safety-assured maneuver.

Formal models can facilitate traceability between driving decisions (down to the level of specific software or hardware pieces) and these rules. The process of formalizing the specific parameters to be used within these rules and their associated hierarchies is a delicate balance. Uncertainty could be reduced if a set of rules, their parameters and their hierarchy are agreed on in advance. It should be noted that safe driving is inherently based on assumptions about other (vulnerable) road users (e.g. maximum deceleration). This is particularly important for occlusion scenarios. On the other hand, driving too defensively may confuse other (vulnerable) road users and could lead to a safety incident.

Such rules can be further described within the context of a set of specifically defined constructs:

- A **DANGEROUS SITUATION** is a state of the automated vehicle such that there exists the possibility of a collision, e.g. the safe longitudinal or safe lateral distance has been violated. This could be caused by the automated vehicle itself, another (vulnerable) road user or due to a change in the environment.
- The **DANGEROUS TIME** is all the time(s) in which the automated vehicle is in a dangerous situation.
- The **DANGER THRESHOLD** is the moment in time immediately before the automated vehicle enters a dangerous situation.
- A **PROPER RESPONSE** is the reaction the automated vehicle should perform to escape a dangerous situation and return to a safe state.

The idea is that if the automated vehicle implements a proper response to a dangerous situation at the danger threshold, then the automated vehicle should not cause collisions on the basis of its own actions and should often be able to avoid collisions caused by others who were not driving safely.

2.2.2.10 TRAFFIC RULES

Traffic rules are an important part of the behavior of any vehicle on the road. All automated vehicles should comply with the traffic rules in the ODDs that they operate in. However, not all traffic rules are created equal. Some traffic rules are explicit, such as the meaning or purpose of a STOP sign or speed limit. Other traffic rules, however, are open to interpretation. For instance, California's Basic Speed Law states that the vehicle should not drive faster than is safer for current conditions (CA Vehicle Code, 1959). However, "safer for current conditions" is not explicitly defined and so could be subject to interpretation. For these subjective traffic rules, a uniform machine-interpretable definition of the expected behavior (e.g. by providing updated parameters to use in a formal safety model such as the one in the Drive Planning element would reduce interpretation uncertainty).

2.2.2.11 MOTION CONTROL

To implement the desired vehicle motion, precise actuator commands must be derived from the given trajectory that is the output of the Drive Planning element (see Section 2.2.2.9). Therefore, a motion controller is necessary for generating lateral and longitudinal commands. The respective closed control loops must be stable with sufficient reserve to compensate for dynamic changes in road conditions, the vehicle dynamics and while performing mode transitions. The generated actuator commands are then allocated to steering, braking and the powertrain.

2.2.2.12 MOTION ACTUATORS

The motion actuators for steering and braking systems and the powertrain form the primary ability to control the motion of the ego vehicle. For this reason, they are often referred to as primary actuators. With regards to the aforementioned fail-safe and fail-degraded capabilities, there are various approaches to achieving fail-operational performance goals. Depending on the item definition, different sets of maneuvers can be derived that still have to be actuated and accordingly require different fail-operational capabilities of the different primary actuator systems.

Lateral and longitudinal guidance as performed by the motion actuators have to fulfill the capabilities according to the item definition.

2.2.2.12.1 STEERING SYSTEM

The aim of a steering system is to control the lateral movement of a vehicle. The steering system has to deal with a lot of interference, such as road undulation, crosswind or friction coefficient, which directly affects the intended lateral movement.

To fulfill the capabilities, particularly being able to fail degraded, there are now fail-operational EPS systems that have an additional independent electronic system. This can fail degraded while retaining enough performance to control lateral movement. These EPS systems are generally suitable to act as an element covering the requirements based on the capabilities. In addition to this, there are further solutions for covering the capabilities, e.g. rear-wheel steering or yawing by braking.

2.2.2.12.2 BRAKING SYSTEM

The aim of a braking system is to control the longitudinal movement of a vehicle in terms of deceleration requested by motion control. As with manual driving, stability functions such as ABS and ESC are crucial prerequisites for ADS-controlled deceleration requests. However, the impact of the automated driving functionality to the brake functions should be considered.

2.2.2.12.3 POWERTRAIN

The aim of the powertrain is to control the longitudinal movement of a vehicle in terms of acceleration. Compared to the other two steering system and braking system elements, this element may not need to be fail-degraded.

2.2.2.13 BODY CONTROL WITH SECONDARY ACTUATORS

The role of body control for automated driving is mainly to communicate planned driving maneuvers and to enable safe and lawful driving conditions (e.g. ensuring a clear view through the windshield or adequate control of the headlights). Therefore, components such as indicator lights, headlights and the windscreen wiper motor are often referred to as secondary actuators, as they do not directly influence the ego motion of the vehicle.

The following describes examples of potential impacts that should be controlled by the automated driving system:

- External lights should illuminate with the correct intensity to ensure adequate visibility to surrounding (vulnerable) road users and to provide a bright illumination for optical sensors (e.g. a camera sensor). The automated driving system has to ensure this operation when in automated driving mode, as the driver may be performing other tasks.
- Warning or indicator lights should work correctly, as they may confuse (vulnerable) road users (e.g. by unintended activation or indication of wrong direction). Additional communication systems may be needed depending on the item definition.
- Windshields (and rear mirrors) should be kept clean during automated driving mode, as a safe takeover by the vehicle operator needs to be ensured by providing a clear view to the front and the rear. Thus, cleaning, air conditioning and heating systems should provide adequate operation during the automated driving mode.

Passive safety components (e.g. seat adjustment, seat belt pre-tensioners, airbags) are not considered in this publication. Nonetheless, the impact automated driving has on these components should be considered.

2.2.2.14 HUMAN-MACHINE INTERACTION

Human-machine interaction (HMI) is considered a crucial element for the safe operation of SAE L3, L4 or L5 vehicles. HMI provides the means of interaction between human and machine to exchange information and operations and is designed in a way that makes using the automated driving system clear and intuitive for users. Therefore, HMI can use visual cues, tactile feedback and acoustic cues to support the user with relevant information, and it can offer different types of interfaces to receive input from the user. HMI should be carefully designed to consider the psychological and cognitive traits and states of human beings with the goal of optimizing the human's understanding of the task and situation and of reducing accidental misuse or incorrect operations.

In vehicles with different levels of automation (SAE L1, L2, L3 or higher), the most important and most challenging goal for HMI is the user's correct interpretation of the actual driving mode and their affiliated responsibilities and (driving) tasks:

- In the moment of a mode transition
- While driving with the same automation mode for a certain period of time

Regarding the different levels of automation (SAE L0-5), the user's driving tasks and responsibilities change with increasing automation, while each level places different demands on the user (compare the roles of human driver and automated driving system by level of driving automation in SAE J3016, as depicted by Figure 25):

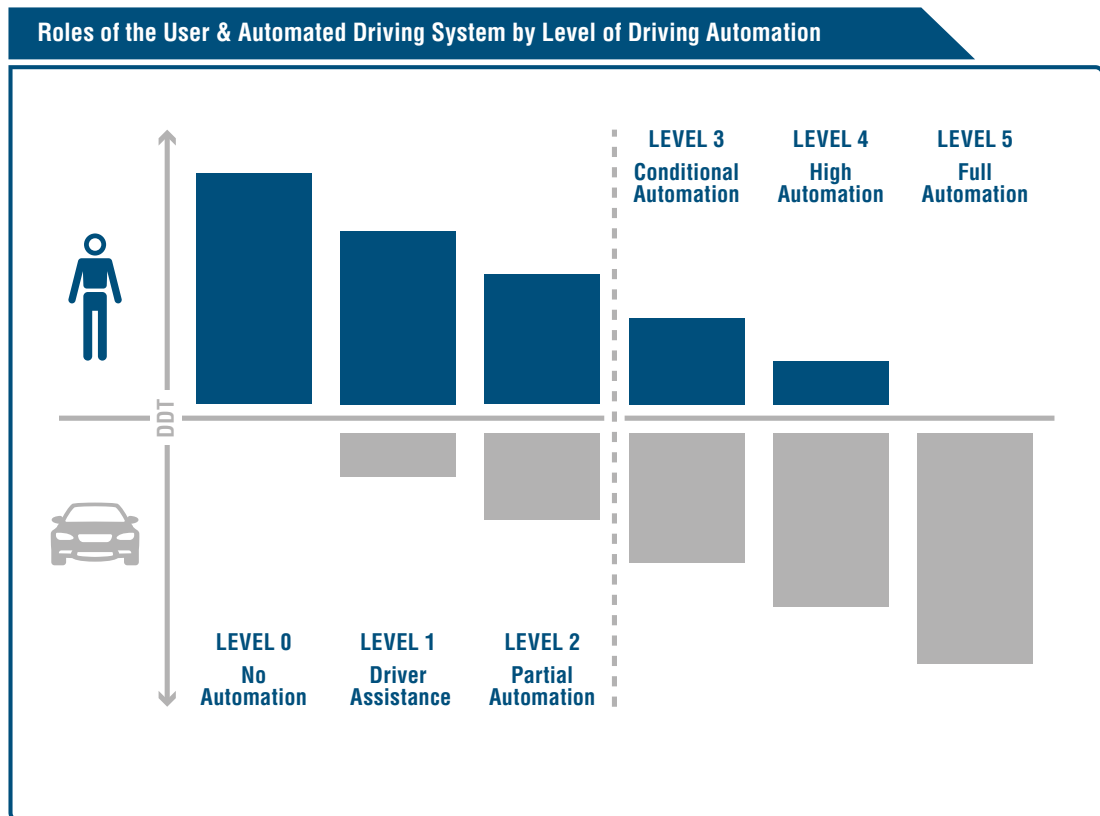


Figure 25: Roles of the User & Automated Driving System by Level of Driving Automation

Within these levels of automation, there is a paradigm shift at the introduction of an L3 automated driving system, as this is the first time the vehicle operator is allowed to cede full control to the vehicle during the nominal driving task within the specified ODD. Therefore, in vehicles equipped with different levels of automation, the vehicle operator experiences several systems with completely different tasks and responsibilities. Mode confusion and a diffusion of responsibility might be possible consequences. Above all, L3 and L2 have a high potential of being mixed up by the driver, as both take over longitudinal and lateral control, while one demands continuous monitoring and the other does not (see Table 7 and Figure 25).

Thus, the HMI design in vehicles with different levels of automation is safety relevant and should be carefully designed to allow the driver to discriminate between different modes. It therefore should:

- Reliably detect intended driver behavior during activation and, above all, deactivation of a certain driving mode and during (driver-initiated) transitions from L5, L4 or L3 to lower levels of automation (minimize false positive and false negative). This requirement refers to all types of HMI operations, including remote control
- Point out the actual driving mode and the driver's responsibility in an unambiguous and easily understandable way
- Promote an appropriate trust in automation for the actual driving mode
- Issue prominent and easily understandable takeover requests (e.g. combining acoustic and visual signals) that give the vehicle operator enough time to take over manual control and regain situational awareness

A first approach for defining evaluation criteria was also released by the National Highway Safety Administration (NHTSA, 2017). Following NHTSA's guidance, HMI of an automated driving system should at minimum "be capable of informing a human operator or occupant through various indicators that the ADS is:

- Functioning properly
- Currently engaged in ADS mode
- Currently 'unavailable' for use
- Experiencing a malfunction and/or
- Requesting control transition from the ADS to the operator" (NHTSA, 2017, p. 10).

To make sure that HMI meets all the requirements on usability and safety, studies with subjects unfamiliar with automated driving should be implemented to test, assess and validate each element. This means that the subjects have no more experience or prior knowledge of the system. Each HMI requirement should be operationalized through suitable use cases that demonstrate how users handle the driver interface and displays within the driving environment. The Code of Practice for the Design and Evaluation of ADAS (as elaborated by Response 3 of the Preventive and Active Safety Applications Integrated Project [PReVENT]) proposes, that "a number of 20 valid datasets per scenario can supply a basic indication of validity" (Knapp, Neumann, Brockmann, Walz, & Winkle, 2009, p.15). This means that 20 out of 20 subjects have to "react [...] as previously anticipated or in an adequate way to control the situation" (Knapp, Neumann, Brockmann, Walz, & Winkle, 2009, p.15).

Even though a L4 automated driving system has a further reduced responsibility of the driving task, such that the minimal risk maneuver and minimal risk condition replace the need for a driver takeover due to ODD restrictions or system malfunction, this does not mean that HMI is less important at this level of automation. If equipped with different levels of automation, the vehicle faces the same challenges of mode awareness and responsibility diffusion as seen in L3. Because L4 may have a different ODD, including urban areas, HMI may also include communication to the relevant (vulnerable) road users in the surrounding environment concerning the status of the vehicle motion, the state of the vehicle and the vehicle's intent, but not necessarily the level of automation due to reasons of misuse by actors "testing" the system. Another new use case and challenge for HMI in L4 is the proper indication and explanation of the entire sequence of passenger pick-up and drop-off, e.g. for ride hailing services. Thus, developments relating to HMI and the human factors remain an important part of automated driving development.

2.2.2.15 USER STATE DETERMINATION

In addition to well-designed HMI for the user, other systems can measure and portray information useful for the user and the relevant (vulnerable) road users in the environment by using new state-of-the-art technologies (e.g. a driver-monitoring camera) and by taking advantage of established technologies (see Table 6).

State-of-the-Art Technology								
HMI Technology Map (Example)	Driver Monitor Camera	Seat Position	Lighting Element	Hands-on Steering Wheel Sensor	Occupant Sensor	Brake Pedal/Acc. Pedal	Indicator Stalk Switch	Steering Column Torque
User Intent	+		+	++		++	+	++
User Readiness for Takeover	+	+		+		+		+
User Distraction	++				+			
Mode Confusion	++	+		++		++		++
Driver Absence	++				++			
Signalling Vehicle Intent to Pedestrians			++					

▪ + = A higher number of + indicates a greater potential of sensor data being an indicator for the occupant attribute.

Table 6: Example of Available Technology and Potential Use of Sensor Data to Measure or Detect Occupant Attributes and to Inform

Like HMI, the sensor set to measure occupant attributes such as driver distraction or mode confusion should be tested on reliability and validity using a heterogeneous sample of potential clients, because certain eye shapes, glasses or heights may challenge the corresponding sensor systems.

Role of the Automated Driving System

Level of Driving Automation	Role of the User	Role of the Automated Driving System
DRIVER PERFORMS PART OR ALL OF THE DDT		
Level 0 No Driving Automation	Driver (at all times): <ul style="list-style-type: none"> ▪ Performs the entire DDT 	Automated driving system (if any): <ul style="list-style-type: none"> ▪ Does not perform any part of the DDT on a sustained basis (although other vehicle systems may provide warnings or support, such as momentary emergency intervention)
Level 1 Driver Assistance	Driver (at all times): <ul style="list-style-type: none"> ▪ Performs the remainder of the DDT not performed by the driving automation system ▪ Supervises the driving automation system and intervenes as necessary to maintain safe operation of the vehicle ▪ Determines whether/when engagement or disengagement of the driving automation system is appropriate ▪ Immediately performs the entire DDT whenever required or desired 	Automated driving system (while engaged): <ul style="list-style-type: none"> ▪ Performs part of the DDT by executing either the longitudinal or the lateral vehicle motion control subtasks ▪ Disengages immediately upon driver request
Level 2 Partial Driving Automation	Driver (at all times): <ul style="list-style-type: none"> ▪ Performs the remainder of the DDT not performed by the driving automation system ▪ Supervises the driving automation system and intervenes as necessary to maintain safe operation of the vehicle ▪ Determines whether/when engagement and disengagement of the driving automation system is appropriate ▪ Immediately performs the entire DDT whenever required or desired 	Automated driving system (while engaged): <ul style="list-style-type: none"> ▪ Performs part of the DDT by executing both the lateral and the longitudinal vehicle motion control subtasks ▪ Disengages immediately upon driver request
<ul style="list-style-type: none"> ▪ Key: DDT = DYNAMIC DRIVING TASK 		

Roles of the User and Driving Automation System

Level of Driving Automation	Role of the User	Role of the Automated Driving System
ADS PERFORMS THE ENTIRE DDT WHILE ENGAGED		
Level 3 Conditional Driving Automation	<p>Driver (while the ADS is not engaged):</p> <ul style="list-style-type: none"> ▪ Verifies operational readiness of the ADS-equipped vehicle ▪ Determines when engagement of ADS is appropriate ▪ Becomes the DDT fallback-ready user when the ADS is engaged <p>DDT fallback-ready user (while the ADS is engaged):</p> <ul style="list-style-type: none"> ▪ Is receptive to a request to intervene and responds by performing DDT fallback in a timely manner ▪ Is receptive to DDT performance-relevant system failures in vehicle systems and, upon occurrence, performs DDT fallback in a timely manner ▪ Determines whether and how to achieve a minimal risk condition ▪ Becomes the driver upon requesting disengagement of the ADS 	<p>ADS (while not engaged):</p> <ul style="list-style-type: none"> ▪ Permits engagement only within its ODD <p>ADS (while engaged):</p> <ul style="list-style-type: none"> ▪ Performs the entire DDT ▪ Determines whether ODD limits are about to be exceeded and, if so, issues a timely request to intervene to the DDT fallback-ready user ▪ Determines whether there is a DDT performance-relevant system failure of the ADS and, if so, issues a timely request to intervene to the DDT fallback-ready user ▪ Disengages at an appropriate time after issuing a request to intervene ▪ Disengages immediately upon driver request
Level 4 High Driving Automation	<p>Driver/dispatcher (while the ADS is not engaged):</p> <ul style="list-style-type: none"> ▪ Verifies operational readiness of the ADS-equipped vehicle ▪ Determines whether to engage the ADS ▪ Becomes a passenger when the ADS is engaged only if physically present in the vehicle <p>Passenger/dispatcher (while the ADS is engaged):</p> <ul style="list-style-type: none"> ▪ Needs not perform the DDT or DDT fallback ▪ Needs not determine whether and how to achieve a minimal risk condition ▪ May perform the DDT fallback following a request to intervene ▪ May request that the ADS disengage and may achieve a minimal risk condition after it is disengaged ▪ May become the driver after a requested disengagement 	<p>ADS (while not engaged):</p> <ul style="list-style-type: none"> ▪ Permits engagement only within its ODD <p>ADS (while engaged):</p> <ul style="list-style-type: none"> ▪ Performs the entire DDT ▪ May issue a timely request to intervene ▪ Performs DDT fallback and transitions automatically to a minimal risk condition when: <ul style="list-style-type: none"> - A DDT performance-relevant system failure occurs OR - A user does not respond to a request to intervene OR - A user requests that it achieve a minimal risk condition ▪ Disengages, if appropriate, only after: <ul style="list-style-type: none"> - It achieves a minimal risk condition or - A driver is performing the DDT ▪ May delay user-requested disengagement
Level 5 Full Driving Automation	<p>Driver / dispatcher (while the ADS is not engaged):</p> <ul style="list-style-type: none"> ▪ Verifies operational readiness of the ADS-equipped vehicle ▪ Determines whether to engage the ADS ▪ Becomes a passenger when the ADS is engaged only if physically present in the vehicle <p>Passenger/dispatcher (while the ADS is engaged):</p> <ul style="list-style-type: none"> ▪ Needs not perform the DDT or DDT fallback ▪ Needs not determine whether and how to achieve a minimal risk condition ▪ May perform the DDT fallback following a request to intervene ▪ May request that the ADS disengage and may achieve a minimal risk condition after it is disengaged ▪ May become the driver after a requested disengagement 	<p>ADS (while not engaged):</p> <ul style="list-style-type: none"> ▪ Permits engagement of the ADS under all driver-manageable on-road conditions <p>ADS (while engaged):</p> <ul style="list-style-type: none"> ▪ Performs the entire DDT ▪ Performs DDT fallback and transitions automatically to a minimal risk condition when: <ul style="list-style-type: none"> - A DDT performance-relevant system failure occurs OR - A user does not respond to a request to intervene OR - A user requests that it achieve a minimal risk condition ▪ Disengages, if appropriate, only after: <ul style="list-style-type: none"> - It achieves a minimal risk condition or - A driver is performing the DDT ▪ May delay a user-requested disengagement

Table 7: Roles of the User and Driving Automation System

2.2.2.16 VEHICLE STATE

Beneath the obvious driving task that will shift from the user's responsibility to that of the automated driving system, monitoring and maintenance duties also should be in charge of the automated driving system while it is in automated driving mode. The Vehicle State has to inform the system of any conditions that would block activation of the automated driving system or to disable the automated driving system in time in the appropriate situations. The vehicle operator should then be able to carry out their mission.

The monitoring of the Vehicle State includes:

- The status of the energy storage system such as the fuel or the electric battery's state of charge
- The tire pressure
- Oil temperature and level
- Door status

2.2.2.17 MONITORS (NOMINAL AND DEGRADED MODES)

Monitors are essential for ensuring safe operation of the system by monitoring the state and behavior of system elements in terms of performance, security events and failures. Monitors could be included as sub-elements or as a separate element monitoring an event chain. The tasks of each monitor may include, but are not limited to:

- Monitoring fail-safe performance (FS_7)
- Monitoring the availability of fail-degraded performance (FD_2)
- Monitoring cybersecurity events

If a monitor detects a lack of performance or one or more failures, this information is sent to the ADS Mode Manager, where appropriate measures are taken.

2.2.2.18 PROCESSING UNIT

The processing unit needs to be developed in accordance with ISO 26262 and under consideration of the safety goals of the automated driving system. It should offer enough integrity, performance calculating power, availability, cybersecurity support, real-time support, automotive bus interfaces, high speed data interfaces, digital and analog pins and low-power mode.

2.2.2.19 POWER SUPPLY

The power supply should provide the required availability and integrity depending on the item definition, thus supporting the automated driving capabilities. If one power supply cannot implement the availability or integrity required, two independent power supplies are a possible solution. In this case, no single point of failure or dependent failure should affect both power supplies at once. The power supply should assure the availability of the automated driving system component both for nominal and degraded performance.

2.2.2.20 COMMUNICATION NETWORK

The communication network should provide the required availability and integrity depending on the item definition, thus supporting the automated driving capabilities. One method to support this is using a high diversity communication network for both nominal and degraded elements.

2.3 Generic Logical Architecture

Section 2.1.6.1 introduced the generic Sense – Plan – Act design paradigm and derived extensions via functional safety and safety of the intended functionality, leading to the deriving of the capabilities of an automated driving system. Section 2.2.2 introduced the logical building blocks (elements) of a system for implementing automated driving functionalities.

This chapter combines these inputs into a more specific design outline, albeit one that is still free of implementation-specific system design aspects. However, it can be regarded as a system blueprint. By the end of this chapter, it will be evident how the elements relate in an automated driving system. In addition to logical elements, the implementation of an automated driving system comprises calculation resources, a communication and energy network and storing capacities. These implementation elements are implementation-specific, and so this publication omits their implementation. Appendix A uses four exemplary functionalities to explain the possible specific properties of relevant elements. It focuses on highlighting the potential differences between the function-specific implementations instead of providing full example descriptions.

A generic architecture is derived by compiling signal chains of derived capabilities from Section 2.2. The resulting logical architecture provides a complete view of the connection and signal flow between the different elements. The functional architecture of the intended functionality of an automated driving system is shown in Figure 26 and Figure 27.

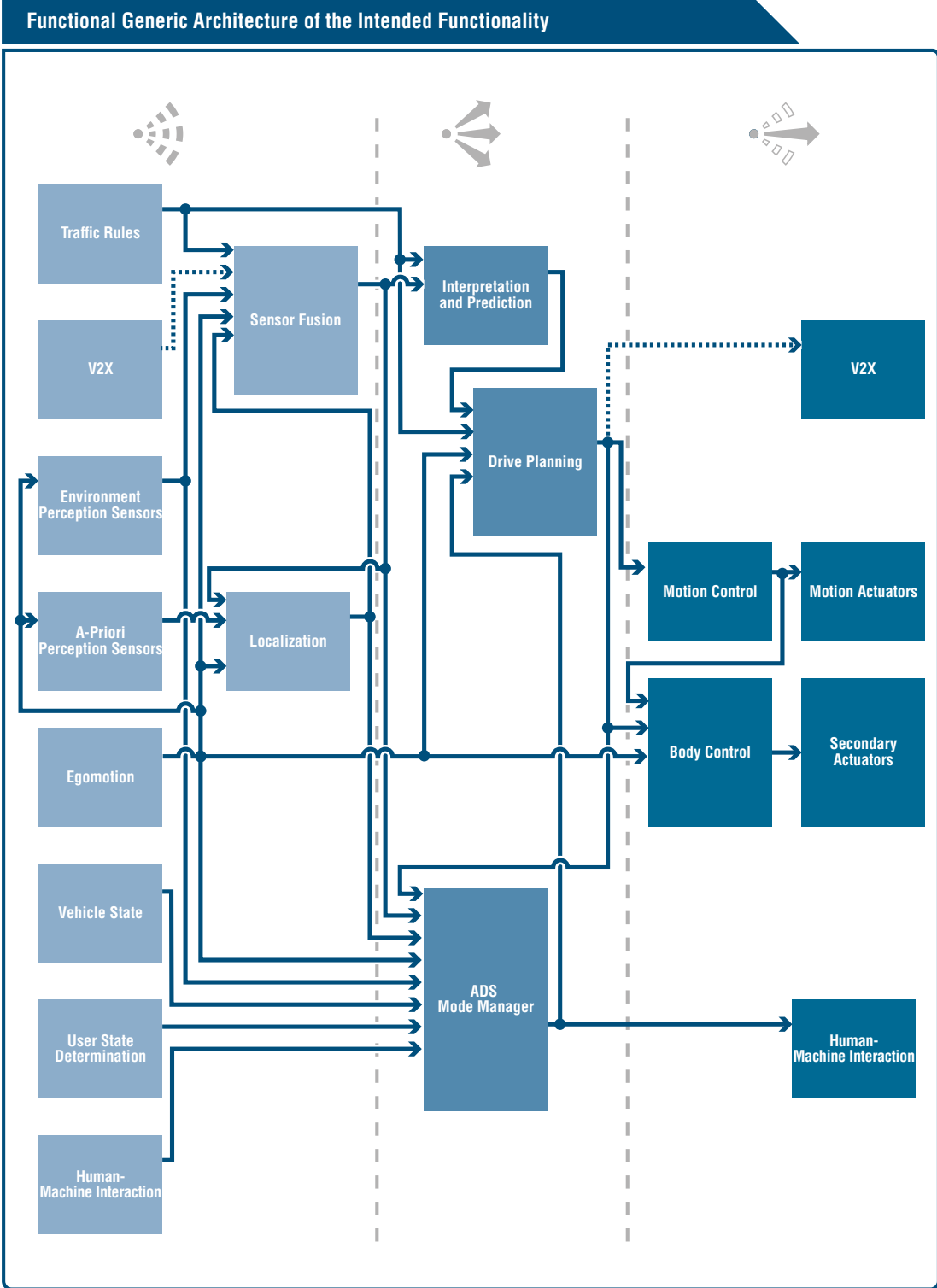


Figure 26: Functional Generic Architecture of the Intended Functionality

In addition to the elements described in Section 2.2, a set of additional requirements should also be fulfilled in order to represent the current state of the art. Each element should have a fail-safe and/or fail-silent mode, depending on the individual system design aspects. In any case, the current performance and failures of individual elements or a combination of elements should be observed and reported to the system monitor.

Redundant elements should avoid dependent failure, so they should be separated by design. Furthermore, redundant elements should avoid common cause failure, so diversity could be considered during the design phase. Lastly, the system should maintain at least degraded capability (even when a single element is not available). Integrating these aspects into the functional architecture leads to an architecture with enhanced elements.

As a result of the above requirements, a generic architecture could be derived by instantiating elements multiple times. These instances might have different individual properties, and multiple instantiations of elements may be included in elements.

Functional Generic Architecture of the Intended Functionality, including Monitors

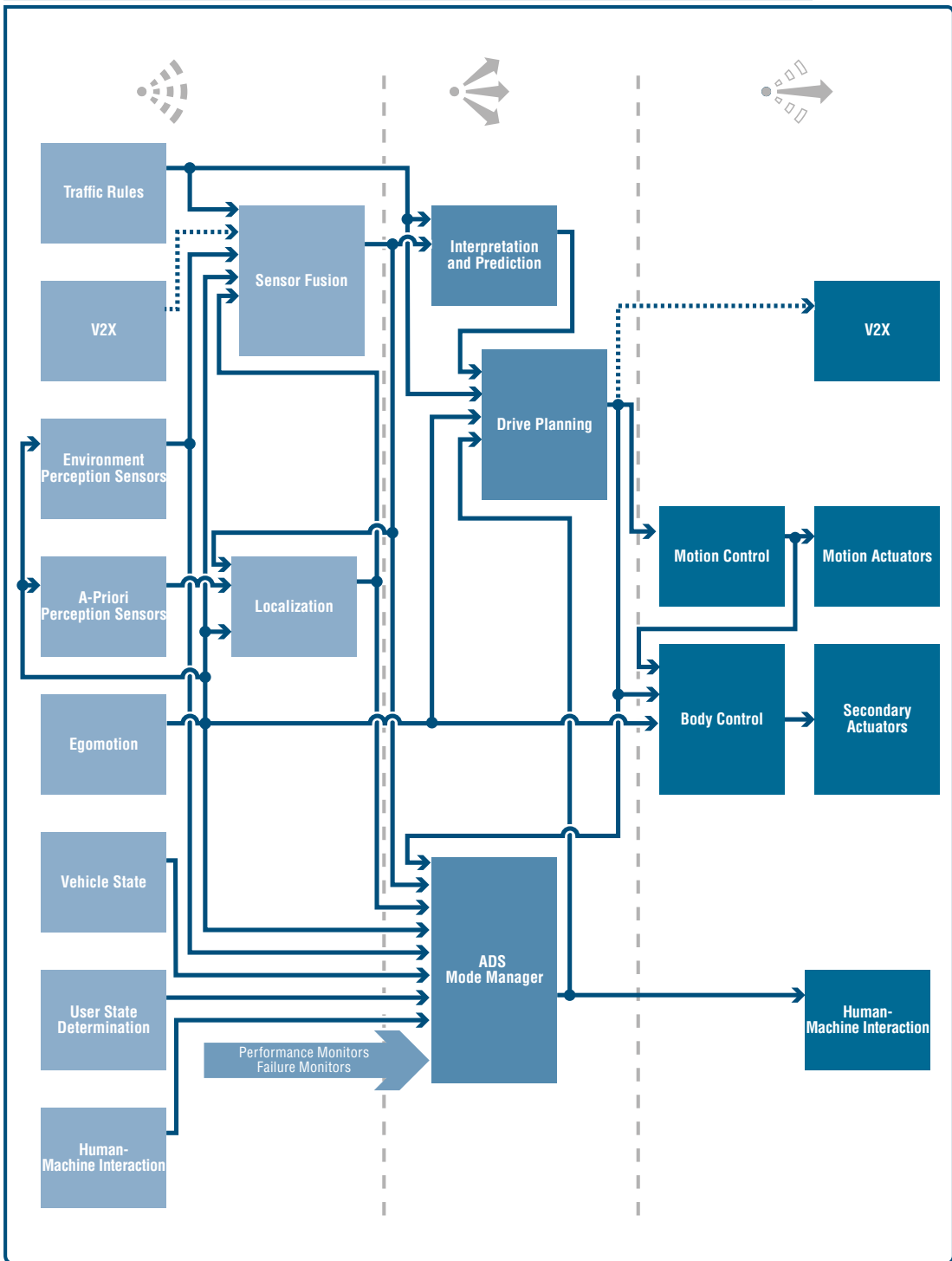


Figure 27: Functional Generic Architecture of the Intended Functionality, including Monitors

The final step assigns capabilities to the elements. Figure 28 demonstrates that all capabilities are implemented, that all elements are assigned to at least one capability and that all elements are connected to each other. The connection of capabilities, as shown in Figure 28, is also evident. It is important to note that while the real world is not depicted in the automated driving system's closed-loop system of sensing and reacting to the real world, this closed loop system does indeed begin with and in turn affects the real world.

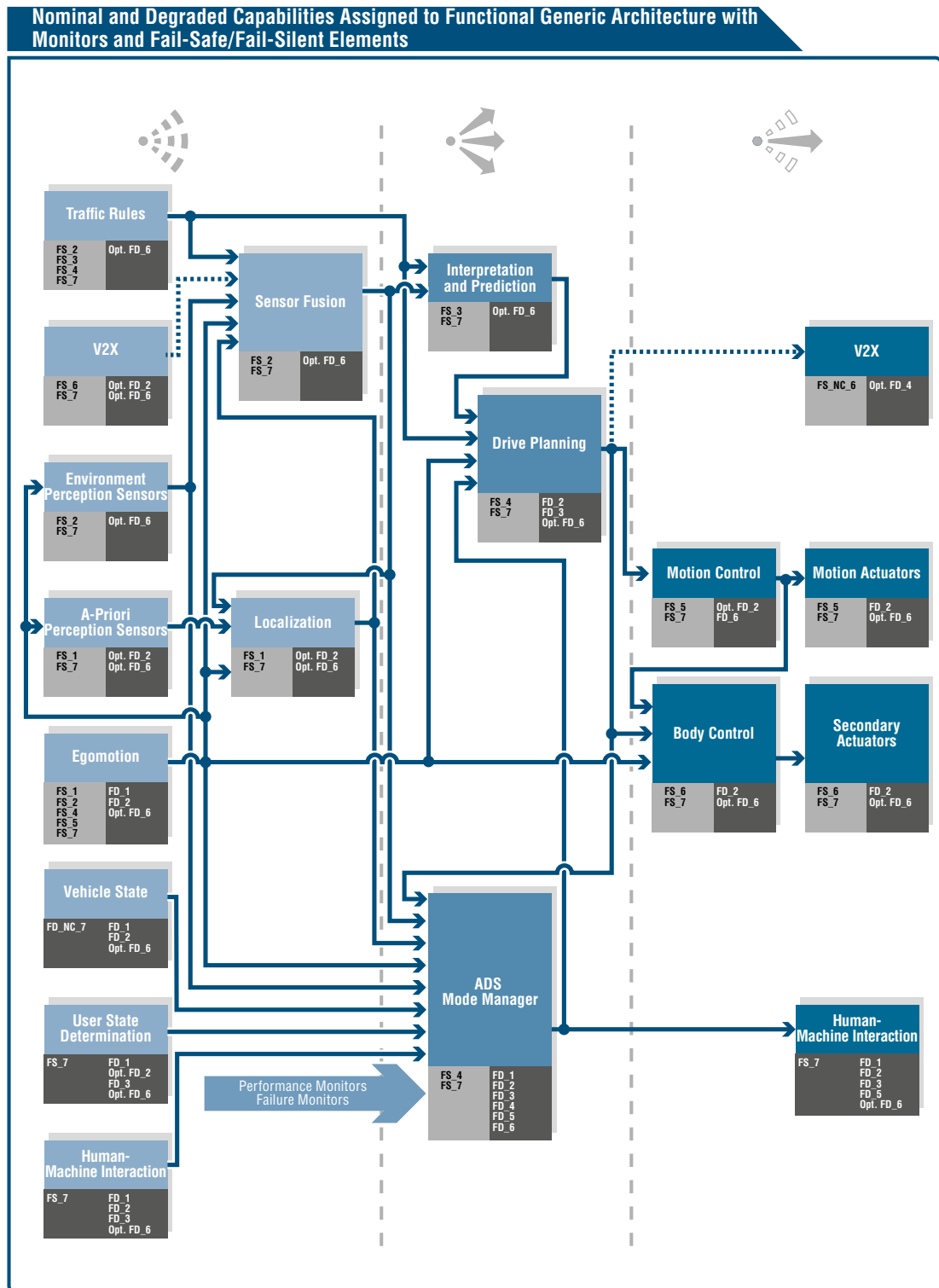


Figure 28: Nominal and Degraded Capabilities Assigned to Functional Generic Architecture with Monitors and Fail-Safe/Fail-Silent Elements

Chapter

03

**VERIFICATION
AND VALIDATION**

3 Verification and Validation

This chapter addresses the V&V of automated driving systems, including field monitoring and updates. Section 3.1 introduces the main steps and general approach and defines the scope of this chapter. Section 3.2 lists five key challenges that are unique to the V&V of L3 and higher automated driving systems. Section 3.3 proposes solutions for each of the challenges and includes a discussion of the various test platforms involved. Section 3.4 discusses the quantity and quality of real-world driving required, while Section 3.5 reviews the use of simulation for V&V. Finally, Section 3.6 focuses on specific V&V considerations for individual elements of an automated driving system. Although this publication recognizes the possibility that validation testing may trigger functional design changes, most of this chapter focuses on validating a stable system in a fixed ODD. However, Section 3.7 discusses post-deployment field operations, including the monitoring and management of configuration and ODD changes and updates.

3.1 The Scope and Main Steps of V&V for Automated Driving Systems

This chapter focuses on V&V as it relates to the safety validation of SAE L3 and L4 automated driving systems. Thus, its scope excludes the V&V of product requirements not related to safety, such as comfort and efficiency. It also excludes standard V&V processes already in use for SAE L0–L2 components, subsystems or systems (e.g. system functions such as start-up and flashing or smoke tests and stress tests described in ISO and ISTQB standards – see ISO 26262, for example). This publication assumes that SAE L3 and L4 systems will adhere to these same standards, which cover many of the verification testing procedures. For example, the V&V methods and processes for security are the same as for L0–L2, L3 and L4 systems and are stated in the upcoming ISO/SAE 21434 standard. Furthermore, it also excludes V&V activities in the production line (e.g. sensor adjustment, security in the production line), which can also follow standard SAE L0–L2 procedures.

Section 2.1 outlines the general concept of the systematic safety system development approach for an automated driving system. This process highlights the main V&V steps that are essential for the safe deployment and continued operation of SAE L3 or higher automated driving systems (see Figure 29).

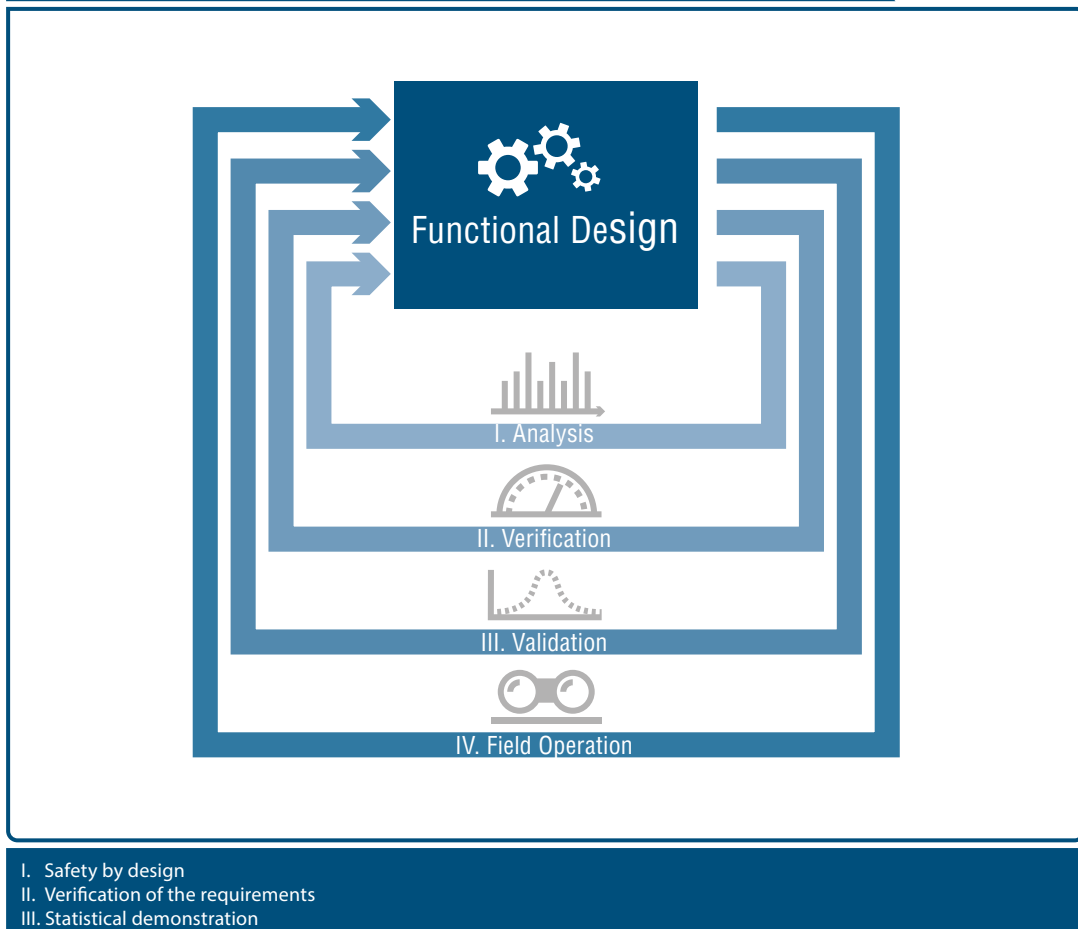


Figure 29: Process of Functional Design in acc. with ISO/PAS 21448 SOTIF

Full system safety validation requires not only testing, but also efforts such as quality audits of the development process, or the implementation of a robust system design through analysis techniques. These efforts combine to form a safety argumentation for the automated driving system. Given that the V&V on the analysis side is the same as for L0–L2 systems (see ISO 26262, for example), but with increased complexity, this chapter focuses on the testing side of V&V.

The first main step is to verify that all the requirements derived through the safety by design strategy are met. This step ensures that known scenarios are covered and that the system behaves as expected. Thus, verification focuses on readily testable requirements and can rely on well-established safety by design processes for systems that have already been integrated into production vehicles for decades. For example, a throttle-by-wire system prevents unrequested positive torque at the wheels through a set of verifiable measures such as redundant accelerator pedal position sensors and redundant microprocessors running redundant software. In line with safety by design principles and verifiable requirements, modern automated driving systems require a design and the testing of measures that ensure safe system output. As shown in Figure 29, verification may lead to improvements to the functional design that result in new verification needs. This iterative process increases the confidence in safety by addressing known unsafe scenarios and thus shrinking its area.

While the principle of safety by design is fundamental to the safety approach, it remains insufficient for automated driving systems, because of the existence of unknown unsafe scenarios that cannot be directly designed for or verified (see Chapter 2). For example, it is impossible to foresee every possible combination of sun angle and clothing worn by pedestrians, or objects that may occlude them. Therefore, to meet the overall safety vision outlined in Chapter 1, validation aims to build the statistical argument to confirm the safety across both known and unknown scenarios with enough confidence. This represents the second step in the V&V process. 100 % reliability of the system and 100 % confidence in a given level of reliability are not possible due to the complexity and time variance of the system and the corresponding uncertainties. Thus, there will remain some small risk of crashes. The concept of residual risk has already been accepted for a long time now (see the rollout of airbags or new medicines). Validation puts the verified system to the test in scenarios or situations that the system would likely encounter in everyday driving after its release. These scenarios can either be controlled directly in a physical (closed-course proving ground) or virtual (simulation of pre-defined scenarios) environment, or they can arise spontaneously during operation in the real world (open-road testing or simulation of randomly generated scenarios). Like verification, validation may trigger changes in the function design. However, a valid statistical claim requires that the system under test be stable, and thus any iteration on the function design will require repeating any validation tests that may have been affected by the change. Given the considerable extent of testing likely required for validation, this publication recommends minimizing the degree of iteration on the functional design once validation has started.

The third step (loosely included in the broader V&V process) consists of post-deployment observation. This includes the field monitoring of the safety performance and security of the automated driving system, and any updates required to address vulnerabilities discovered after deployment. Updates would require very careful change management and retesting to ensure that changes in the system do not introduce new risks (see Section 3.7).

3.2 Key Challenges for V&V of L3 and L4 Systems

This section decomposes the challenge of V&V for SAE L3 and L4 systems into five separate challenges, and Section 3.3 outlines viable solutions that cope with these challenges. The overall challenge is to increase confidence that the system has achieved a positive risk balance compared to the performance of a human driver (see Chapter 1), considering all possible driving scenarios occurring with a noticeable exposure. As complete testing of every single scenario is neither appropriate nor technically possible, a viable method to statistically demonstrate system safety is defined further below.

CHALLENGE 1

Statistical demonstration of system safety and a positive risk balance without driver interaction

In SAE L0–L2 systems, the human driver is responsible for supervising vehicle controllability while driving. Thus, the safety assessment of these systems may primarily focus on the safety of actuators and electro-mechanical systems, and mostly considers only selected worst-case scenarios. In L3 and L4 systems, it cannot be assumed that the driver is fully alert in all scenarios. Thus, these systems require a much more thorough consideration of the automated driving system's ability to safely perform the driving function itself. This greatly increases the number of possible scenarios and implies the need to include statistical considerations in the overall safety argumentation.

CHALLENGE 2

System safety with driver interaction (especially in takeover maneuvers)

In the V&V of L3 (and to a minor extent also L4) systems, takeover scenarios also need to be assessed, as these impact the safety of automated vehicles. The human driver as the fallback must maintain mode awareness and receive an unambiguous indication of any mode transitions. Likewise, the system must support effective takeover capability to a reasonable extent during transition to support controllability for humans after takeover situations. In addition, long-term effects of prolonged use of an automated driving system may also desensitize the situational awareness of the driver. These effects need to be analyzed carefully and considered in the overall V&V and safety impact analysis.

CHALLENGE 3

Consideration of scenarios currently not known in traffic

New scenarios result from both previously unseen scenarios involving a single automated driving system and scenarios related to interactions between automated driving systems. These are an important aspect of automation risks. Furthermore, misuse scenarios are probable where (vulnerable) road users interact with automated vehicles. New scenarios can also occur due to changes in the real world (e.g. new traffic signs). These scenarios need to be considered in the overall validation method.

CHALLENGE 4

Validation of various system configurations and variants

An automated driving system comprises several elements that are likely to face software updates over its lifetime. Hardware changes may also occur, as parts of the system might be damaged when the vehicle is in customer use. Consequently, the number of configurations and system variants for L3 and L4 systems is expected to exceed the number of configurations for L2 systems. Each system configuration needs to be verified and validated.

CHALLENGE 5

Validation of (sub) systems that are based on machine learning

Several elements of automated vehicles may rely on algorithms based on machine learning. Compared to currently used algorithms in safety-related components, additional effort is required, and new validation methods need to be adapted to ensure overall system safety (e.g. based on the non-deterministic behavior of machine learning algorithms). Machine-learned (sub) systems and components cannot be decomposed and need to be tested as a whole, which increases testing efforts.

3.3 V&V Approach for Automated Driving Systems

The approach to verifying and validating the safety of automated driving systems needs to address the key challenges mentioned above. This publication emphasizes that V&V processes include both testing and other activities to ensure rigorous development and system design implementation (see Section 3.1). Within the verification process, testing evaluates whether the specified requirement is met. Within the validation process, testing evaluates whether the automated driving system fulfills the intended use cases. However, complete test coverage is not achievable to fully validate the safety of L3 and L4 systems. A proposed methodology is stated below.

Several testing activities contribute to the testing process and establish a specific test strategy. The relevant characteristics can be decomposed by answering the five W (who, what, where, when, why) and two H (how, how well) questions – also known as 5W2H (Ohno, 1988; Tague, 2005). Only by fully answering these seven questions can the specific test case be generated. The questions *when* and *who* should be answered in the same way as for the development of L0–L2 systems. This section outlines the general approach to validating system safety, responding to the key challenges by explicitly describing:

- **WHY AND HOW WELL:** Test goals including the scope, completion criteria and metrics (see Section 3.3.1)
- **HOW:** Test techniques (see Section 3.3.2)
- **WHERE:** Test platforms or test environments (see Section 3.3.3)
- **WHAT:** Test elements or object under test (OuT) (see Section 3.6)

Combining these characteristics establishes a specific test strategy (see Section 3.3.4 and Section 3.4) for L3 and L4 systems to respond to the key challenges and support safety validation. The test strategy greatly impacts the quantity of real-world driving tests (as discussed in Section 3.4), the simulation environment and the number of tests (see Section 3.5). Section 3.6 details the examples for verifying and validating specific elements and capabilities.

3.3.1 Defining Test Goals & Objectives (Why & How Well)

The product safety argumentation combines all necessary validation efforts in a coherent way (see description above) and consequently also describes all test goals and test objectives that should be achieved. For every single test case, the test goal needs to be quantified (e.g. test completion, stopping and resumption criteria) in accordance with the safety argumentation. Furthermore, objective metrics for test completion and test quality need to be defined for the entire test strategy. These are already used in L2 systems and so are not detailed here. For L3 and L4 systems, the metrics for the test coverage of large parameter distributions need to be defined carefully. Some of the most important considerations when defining the test goals include the twelve principles described in Chapter 1 and their related capabilities discussed in Section 2.1.6.

3.3.2 Test Design Techniques (How)

The test design technique defines how the object under test is tested. How defines which test parameters and their specific values of the tested elements are assessed. Various test design techniques are also used for L2 systems. As they highly impact the quality of every single test case and the quality of the overall confidence in the validation, design techniques play a fundamental role in the testing strategy for L3 and L4 systems. ISO 26262 recommends different test design techniques. Test design techniques are classified via knowledge about the object under test. For example, the box approach distinguishes between black box techniques (no or very little information about the OuT is available), grey box techniques (some information is available) and white box techniques (all relevant information, such as the OuT's structure, is available).

The following test design techniques for L3 and L4 systems require greater assiduity than for L0–L2 systems:

- Scheme-based test design techniques
 - Equivalence partitioning test design techniques (see Section 3.4)
 - Boundary value test design techniques
- Search-based test design techniques
 - Design of experiments
 - Mutation test design technique
 - Reactive test design technique

3.3.3 Test Platforms (Where)

Different test goals require test platforms that have been adapted to the respective OuT. The closer the test platforms are to the real world, the less additional tests are needed to transfer the results to the system in use. The Glossary describes all available test environments, descriptions and examples. The main difference in these test environments is in the application of virtual and real stimuli and in the items being tested. Table 9 classifies these stimuli and test items for each test environment.

Test Platform and Test Item					
Test Item ▶ ▼ Test Platform	Target SW (Code)	Target HW (ECU)	Vehicle	Driver	Driving Environment
SiL (Simulation in the Closed Loop)	Virtual	Virtual	Virtual	Virtual	Virtual
	Real			None	
SW Repro (Software Re-processing)	Virtual	Virtual	None	None	Virtual
	Real				
HiL (Hardware in the Closed Loop)	Real	Real	Virtual	Virtual	Virtual
				None	
HW Repro (Hardware Re-processing)	Real	Real	None	None	Virtual
DiL (Driver in the Loop)	Real	Virtual	Virtual	Real	Virtual
		None	None		Real
PG (Proving Ground)	Real	Real	Real	Real	Real
				Robot	
OR (Open Road)	Real	Real	Real	Real	Real

Table 8: Test Platforms and Test Items

3.3.4 Test Strategies in Response to the Key Challenges

To address the key challenges discussed in Section 3.2, test goals, objects under test, test techniques and test platforms can be combined to define a viable test strategy for L3 and L4 systems.

SOLUTION FOR CHALLENGE 1

Statistical demonstration of system safety and positive safety impact without driver interaction

To respond to this challenge, the automated driving system without a driver (Sense – Plan – Act as discussed in Chapter 2) needs to be covered. The following three strategies are combined to implement a general test approach for statistically demonstrating system safety:

- A Use statistical grey box testing in real-world driving tests to cover the variety of real-world driving scenarios to develop:
 - Statistical validation of the perception in real-world tests with final perception hardware in vehicles while using reference sensor systems
 - Validation of the complete closed-loop system in real-world driving conditions
 - Identification of driving scenarios available in the ODD as a basis for addressing challenge 3 (unknown unsafe test scenarios)
- B Implement scenario-based testing for the complete driving system as well as for specific elements in dedicated test platforms using useful test techniques, e.g.:
 - Software/hardware reprocessing: Validation of perception and sensor fusion (re-processing of field measurements with new software releases)
 - SiL: Validation of trajectory-planning and control algorithms in simulations with basic sensor models covering a wide range of variations in the scenarios
 - HiL: E/E failure tests of hardware components, fault injection tests, validation of SiL
 - Proving ground: Validation of the complete system in critical traffic scenarios, validation of SiL and HiL
- C Ensure field monitoring of the system over its lifetime to:
 - Quantify and assess previously unconsidered scenarios
 - Increase the confidence level of the validation with higher statistical power

Statistical testing in real-world driving as outlined in point A has the clear advantage of assessing a realistic system in a realistic driving environment. However, it does not ensure that all critical driving scenarios and driving environments are covered, even with extensive testing efforts. Equivalence class considerations are useful for assessing the quality of the real-world driving (with respect to the coverage of traffic scenarios). The necessary quantity of real-world test driving (distance driven) strongly depends on this quality, which Section 3.4 discusses in greater detail.

The disadvantage of the dedicated testing of specific scenarios as outlined in point B is that only the known traffic scenarios and environments can be covered and that a specific uncertainty remains in the test results, depending on the test platform used. For example, testing the system in heavy fog cannot be reproduced on a proving ground. At the same time, the test results of this scenario in simulations (with imperfect sensor models) has limited meaning. This example underlines the need to combine testing in different test platforms.

Field monitoring as outlined in point C enhances the coverage of scenario-based testing for a sufficiently validated automated driving system.

SOLUTION FOR CHALLENGE 2

Assessment of human driving performance (especially in takeover maneuvers)

The safety of the human driver and automated vehicle system is clearly affected by the human driving performance in combination with the HMI of the automated driving system. This is obvious in takeover scenarios. These safety aspects are tested in the DiL as well as during real-world closed-loop testing on proving grounds and open roads. For open-road testing, intermediate steps from L2 to L3/L4 are necessary, and different gates should be passed sequentially. The following is an example of such a sequence and the steps involved:

- 1 Simulation to find worst-case traffic scenarios for DiL using a basic driver model
- 2 DiL testing of driver performance in combination with HMI
- 3 Real-world testing on the proving ground with a closed-loop L3 or L4 system with a safety driver
- 4 Real-world testing on the proving ground with a closed-loop L3 or L4 system with expert drivers (no safety driver)
- 5 Real-world testing on the proving ground with a closed-loop L3 or L4 system with a representative sample of trained customers and an incremental increase of the ODD (e.g. increasing velocity), and drivers (no safety driver)
- 6 Real-world testing described in steps 3) to 5) on open roads
- 7 Reduced training of customers, and activation of the system in the full ODD
- 8 Field monitoring of system performance in the customer fleet (open-road testing, naturalistic driving studies).

Some steps must also consider the long-term behavior of the driver. The analysis considers foreseeable misuse and abuse in accordance with ISO/PAS 21448. These considerations of foreseeable misuse and abuse may result in safety goals to reduce the risk identified. The extent of coverage for foreseeable misuse and abuse is limited to understanding the imagination and behavior of the customer. The safety goals determine the scope of the V&V of foreseeable misuse and abuse.

Customer case studies should be carried out to demonstrate the level of vehicle controllability that the driver has for the known scenarios and to demonstrate that the defined response times are adequate for the driver to take over the vehicle, especially for L3 systems. Vehicle-level tests are also necessary for validation, and DiL tests should be incorporated (e.g. so that safety-related testing does not jeopardize the safety of the test drivers).

SOLUTION FOR CHALLENGE 3

Consideration of scenarios currently not known in traffic

To tackle this challenge, the human driver as well as the entire automated driving system needs to be examined. Essentially, the test strategy rests on the following test platforms:

- Simulation with bidirectional interaction of a fleet of automated vehicles (L3–L4) and multiple other, non-automated road users (including L0–L2), e.g. open pass (see Eclipse Foundation, 2019)
- DiL and open-road testing to assess unknown scenarios resulting from the interaction of human drivers in L0–L2 vehicles with automated vehicles

In both test platforms, the behavior of the simulated automated vehicles needs to cover a broad range of possible system implementations. Different manufacturers might use different system characteristics, resulting in different scenarios. In the simulation environment, the plant model of the automated vehicle could be modified with different parameterizations to cover this aspect. In addition to simulations, field monitoring focusing on the detection of scenarios may be useful.

SOLUTION FOR CHALLENGE 4

Validation of various system configurations and variants

Due to their complexity, automated driving systems are prone to numerous system configurations and variants in the field (e.g. mounting tolerances of the sensors and actuators, software variants to adapt to world changes). Regression testing is essential for focusing on the changes between configurations. Full traceability along the complete development process is required to identify elements and software components affected by small changes, e.g. in one line of code. For every change in one line of code, for instance, the elements and capabilities affected need to be identified. Testing can then focus on the impact the change has on the affected capabilities compared to the previously tested baseline configuration.

SOLUTION FOR CHALLENGE 5

Validation of (sub) systems that are based on machine learning

For the V&V of safety-related machine learning algorithms, it is crucial to define a safe design process for these algorithms in addition to testing them. Appendix B (particularly Section 6.3 to Section 6.6) examines machine learning in greater detail and describes the basic requirements for the V&V of machine learning algorithms.

SUMMARY OF THE TEST STRATEGY

In conclusion, a viable test strategy responds to the key challenges in the V&V of automated driving systems by carefully breaking down the overall validation objective into specific test goals for every object under test and by defining appropriate test platforms and test design techniques.

As an example, Figure 30 depicts an overview of test platforms combined with objects under test for the different test goals, depending on the specific design of the automated driving system. Single components such as sensors or actuators are tested primarily on the test platforms SiL/software reprocessing and HiL/hardware reprocessing. Different test goals are considered while doing so. In Figure 30, the example of the open road test platform is used to test different test goals, focusing on the entire system.

Summary of the Test Strategy					
	SiL/SW Repro.	HiL/HW Repro.	DiL	Proving Ground	Open Road
Components					
Sensor Fusion, Localization, Perception					
System without Sensors, Prediction (Drive Planning)					
Motion Control, Egomotion					
HMI, User State Detect., ADS Mode Manager					
Entire System					

▪ **Test Goal:**

Technical aspects of SOTIF	Security/penetration testing
Human factor aspects of SOTIF	Validation of virtual test platforms
Functional safety	

Figure 30: Summary of the Test Strategy

The time-consuming nature of some of the steps in the final safety validation testing process (see Section 3.4) prolongs the overall validation process, particularly as the steps cannot all be carried out in parallel (see the solution to challenge 2 in Section 3.3.4). As discussed in Section 2.1.1, automated vehicles should comply with regulations specifying design, construction, performance and durability requirements. Documentation should include the verification and validation process. In addition, the necessary homologation process should be started at an appropriate time (e.g. parallel to the validation

process) under consideration of the changing environment (new types of vehicles, new traffic signs, new roads, etc.) within the overall validation and homologation time span. For this, the homologation body (e.g. represented by a technical service team) and the OEM have to jointly define which kinds of tests are relevant for fulfilling the framework of homologation.

3.4 Quantity and Quality of Testing

As discussed in Section 3.2, one of the key challenges in validating automated driving systems is to statistically demonstrate a positive risk balance. Section 3.1 defines the main safety objective to be demonstrated. In a purely statistical, black box approach (i.e. this would require using the automated driving system as the customer would) “automated vehicles would have to be driven hundreds of millions of miles and sometimes hundreds of billions of miles to demonstrate their reliability in terms of fatalities and injuries” (Kalra & Paddock, 2016, p. 1). Due to the rarity of failure events, real-world test driving alone cannot provide high confidence in the safety of automated driving systems with respect to injuries and fatalities (Wachenfeld, 2017; here, 2019; Salay & Czarnecki, 2018).

To address this challenge, the test strategy proposed in Section 3.3.4 combines statistical testing in real-world driving with one or more of the approaches stated below:

- Defining equivalence classes or scenarios (see below for more details)
- Defining surrogate metrics or leading measures (e.g. crashes or path to collision) to track system safety – especially in regression tests (Fraade-Blanar, Blumenthal, Anderson, & Kalra, 2018). Surrogate metrics and leading measures are concepts stated in the RAND report and are different to equivalence classes, which categorize scenarios by controllability, exposure and severity. Leading measures are performance indicators that suggest a hazardous condition. One example might be infractions to traffic and road rules, which may not necessarily have the potential to cause a collision or injury but will lead to a harmful scenario when other (vulnerable) road users are present. These leading measures could further allow for better assessment of system performance and allow for fine-grained comparison against humans prior to on-road test validation activities.
- Decomposing the system into elements as discussed in Section 2.2, and testing these elements as discussed in Section 3.6.
- Combining different test platforms and test design techniques (e.g. stochastic variation in SiL) to increase test coverage as discussed in Section 3.3.4.
- Leveraging existing knowledge on relevant scenarios (e.g. from crash or accident databases), and carefully tracking these scenarios in real-world tests and/or simulation.

Combining these methods generates the quantity of required real-world and virtual test drives. Furthermore, an a priori assumption of the quantity of test driving should consider the following:

- System design
 - Definition of the ODD, (see also the development examples in Section 1.2), as the reference for the safety impact analysis and the test space are affected.
 - Robustness of the elements (e.g. via redundancy of sensors), which require evidence that the elements are reliable (e.g. reasonable evidence of independence must be presented for redundancy).
- Definition of the reference
 - Reference safety level, e.g. human driver in the ODD, similar to L3/L4 systems
 - Confidence level to be achieved
 - Assumed statistical distribution

Currently, this publication roughly outlines the methods. However, a joint publication on this at a later point would be useful to discuss these methods in greater detail. For L3 systems, at least a similar amount of real-world test driving as for state-of-the-art L2 systems seems appropriate. Consequently, several million kilometers should be driven in the real world for the development and validation of the Highway Pilot (see Section 1.2). This may then differ when the new methods outlined above are used. For other systems such as the Traffic Jam Pilot, Urban Pilot and Car Park Pilot, other metrics adapted to the systems' ODD may be applied, and so the comparison of driven distance may not be reasonable. Validation should be conducted continuously until the required confidence level is achieved.

3.4.1 Equivalence Classes and Scenario-Based Testing

Equivalence class considerations are useful to maximize testing efficiency. The parameter space of influencing factors to be tested is partitioned into classes. For each of these classes, the necessary amount of relevant test cases is then defined. Within one class, a mostly small number of tests suffices to demonstrate the statistical safety of the system in the whole class. The criteria for defining equivalent classes can be supported by the exposure, severity and controllability levels (as defined in ISO 26262) as a representative sample of operational scenarios to be grouped in equivalent classes. All influencing factors are modeled in the SiL/software reprocessing test platforms. Consequently, each equivalence class can be covered precisely during testing in simulation. When interpreting the simulation results, modeling inaccuracies and approximations must be considered to the extent practically possible.

In real-world test driving, equivalence class considerations are useful for the test design on proving grounds and open roads. They are also useful for assessing the quality of real-world test drives in the context of sufficiently covering traffic scenarios. It is impossible to cover all parameter combinations during proving ground testing, because some parameters cannot be controlled in the test setup. Moreover, there may be potentially dangerous combinations of parameters that should be tested only on proving grounds without human test drivers. Finally, not all influencing factors can be directly controlled in real-world test driving on open roads. The probability of sufficiently covering all non-controllable parameter combinations (and hence all equivalence classes) increases with the amount of real-world testing in customer-like environments.

An objective assessment of the test results (essentially long-term measurements from the test drives) reveals the coverage of the equivalence classes. This is therefore a way to monitor the quality of the test drives.

The safety of an automated driving system is influenced by several factors, which can be grouped according to the three entities of traffic: The driver, the vehicle with the automated driving system and the traffic environment. The traffic environment or the traffic scenario contains several characterizing factors that are split into six layers of a scenario [PEGASUS, 2019]:

- LAYER 1** Street layout and condition of the surface
- LAYER 2** Traffic guidance infrastructure, e.g. signs, barriers and markings
- LAYER 3** Overlay of topology and geometry for temporal construction sites
- LAYER 4** Road users and objects, including interactions based on maneuvers
- LAYER 5** Environment conditions (e.g. weather and daytime), including their influence on levels 1 to 4
- LAYER 6** Digital information (e.g. V2X information, digital map)

Furthermore, three levels of abstraction for scenarios are proposed: Functional scenarios, logical scenarios and concrete scenarios. [Menzel, Bagschik, & Maurer, 2018]. Consequently, scenarios and the scenario parameters at the different layers provide a holistic concept for defining equivalence classes. Systems can also be used to assess the safety of an automated driving system in simulations (see Section 3.5.2).

3.5 Simulation

In its broadest sense, simulation can help us to understand the possible behaviors and outcomes of a system in a virtual setting that we can directly control, with much higher efficiency compared to real world testing. Furthermore, some of the tests are only possible in a virtual environment as safety in the real-world test cannot be guaranteed. For automotive applications, simulations may consider an entire system (e.g. a full vehicle with tires and automated driving system functions), a subsystem (e.g. an actuator or a hardware controller) or a component (e.g. a sensor or a communication bus). Simulation introduces models to represent the behavior of the system of interest, for example. Models are abstractions from the physical reality and rely on simplifications of the true complexity in the real world. For example, a vehicle dynamics model may capture the forces acting on the vehicle as a result of actuation, friction and the earth's gravity, but exclude the effect of electromagnetic forces or lunar gravity on the vehicle. Consequently, simulations can be accurate only to some degree. Understanding the accuracy offered by a simulation is key to determining and arguing its use during development and validation activities [Koopman & Wagner, 2018; Winner, Hakuli, Lotz, & Singer, 2016]. The level of accuracy required for a simulation model depends on the test goal and should be established through a separate activity of validating simulation (see Section 3.5.3).

Simulation serves primarily two purposes: To assist the development of a (robust) function and to test and validate the function before release. The following briefly describes both applications of simulation before discussing in more detail the use of simulation for validation and the requirements for validating the simulations.

During development, structured stress testing that challenges the system's software and/or hardware can help to discover and eliminate safety failures, investigate corner cases and determine the boundaries of the system's capabilities. Examples of structured tests applied during development include:

- Exposing the planning and control algorithms to virtual test scenarios
- Determining the limits of the vehicle vision and perception system via synthetic input generation, e.g. generated by a 3D image rendering engine
- Jointly evaluating the performance of perception and planning (perception-in-the-loop)
- Jointly testing camera hardware and vision algorithms
- Running planning and control tests on the vehicle ECU
- Simulating parts of an in-vehicle bus system and testing ECU-to-ECU communication

For validation activities, simulation usage depends on the overall validation strategy and the level of fidelity or accuracy reached by the simulation models. In the case that model accuracy can be shown to sufficiently match real world behavior, simulation could conceivably be used to argue safety directly without real-world driving activities. However, doing so would require a representative sample of simulation scenarios (i.e. representative of the intended use of the system after validation) or a defensible mathematical expression for the contribution of each simulation scenario to the total statistical confidence in the system, likely based on the frequency of occurrence of scenarios in real-world driving. Where models remain inaccurate, simulation could still aid in focusing real-world testing activities to areas of expected system weaknesses as discovered by simulation. Simulation may also increase confidence in the safety of the system. However, arguing statistical safety directly from simulation results remains challenging, because doing so would have to properly account for the uncertainty introduced by the simulation's limited accuracy.

Given the challenges of using simulation results in a statistical argument, real-world driving will remain important, and simulation cannot replace all real-world testing. Nonetheless, real-world driving retains its importance and may in fact aid in the generation of realistic simulation scenarios and in establishing the accuracy of the simulation models:

- Real-world data for vehicle and component model validation: Vehicle data and data measured via vehicle sensors are important sources for quantifying and arguing model accuracy (e.g. vehicle dynamics or sensor models).
- Real-world data for scenario accumulation: Fleet data may help determine which relevant cases to simulate.
- Real-world data for traffic modeling: The generation of novel scenarios in simulation requires realistic road user behavior for virtual simulations in order to remain meaningful and representative.

In summary, simulation for validation can achieve different objectives, depending on the overall validation strategy and the accuracy of the simulation tools:

- Provide qualitative confidence in the safety of the full system
- Contribute directly to statistical confidence in the safety of the full system (caveats apply)
- Provide qualitative or statistical confidence in the performance of specific subsystems or components
- Discover challenging scenarios to test in the real world (e.g. closed course)

3.5.1 Types of Simulation

Numerous different types of simulation exist and can contribute towards different testing goals (see Chapter 3). Regardless of the type of simulation, any simulation result should be reproducible at a later point for traceability and maintenance purposes (see also Section 3.7). For SiL or software reprocessing, this means that the simulator will repeatedly produce the same results for given initial conditions, input data and random seed. For HiL or hardware reprocessing, this means that the hardware configurations, test conditions and any hardware burn-in is comprehensively documented.

Simulation for functional safety testing focuses on detecting system malfunctions and should follow the same approach as for SAE L0–L2 systems [ISO 26262]. System malfunctions can occur due to a failure in any of the software, hardware, software/hardware interactions, software/software interactions, hardware/hardware interactions, or hardware/chemical/physical environment interactions. In addition, where the functional safety concept relies on human intervention (e.g. as a fallback in L3 systems), functional safety testing must ensure the appropriateness of the safety-related human-machine interfaces and controllability in avoiding unreasonable risk. Accordingly, simulation for functional safety testing includes all test platforms:

- SiL or software reprocessing testing to validate the absence of unreasonable risk due to failures in software and software/software interactions (and software/hardware interaction, e.g. through time models included in a SiL or software reprocessing).
- HiL or hardware reprocessing testing to validate the absence of unreasonable risk due to failures in hardware, hardware/hardware, software/hardware, or hardware/chemical/physical environment interactions. This may occur at the component level (e.g. sensors) or subsystem level (e.g. system without sensors, controller).
- DiL testing to validate the absence of unreasonable risk due to failures in software/human and hardware/human interactions.

Using simulation for technical safety in use falls under the still developing domain of SOTIF [ISO/PAS 21448]. Unlike functional safety testing, simulation for technical safety in use focuses on demonstrating safety in the absence of any malfunctions. Its primary purpose is to contribute to confidence (statistical or other) in the system's safety across both known and unknown scenarios. Similar to the use of simulation for functional safety testing, simulation for technical safety in use may involve SiL, software reprocessing, HiL, hardware reprocessing and DiL at the component, subsystem and full system levels, and DiL for human-machine interactions.

Different levels of fidelity may complement each other to enable validation at the full system level. For example, the vehicle perception system may be validated (and its error and noise characteristics assessed) with real-world data or realistic, compute-intensive sensor models. Once the perception performance is assessed, follow-up tests (e.g. of the behavior planning module) may be decoupled from the realistic sensor models and draw on more abstract failure models of the perception module (e.g. through fault insertion testing with the noise models derived in the lower-level validation activities).

Using simulation for human factor safety in use may involve SiL to demonstrate sufficient safety of subsystems that involve human interaction. SiL remains limited, because actual human behavior may differ from modeled human behavior. Therefore, SiL could be complemented with DiL to validate this safety performance when actual human drivers or passengers are in the loop. However, safety-related traffic scenarios with other traffic objects cannot be tested in the real vehicle.

3.5.2 Simulation Scenario Generation

For functional safety testing, simulation scenarios mainly derive directly from testable safety requirements in the safety design or vice versa. For using simulation to test technical safety in use or human factor safety in use, simulation scenarios may also come from different sources, including:

- Challenging scenarios previously encountered by the system during real-world testing
- Scenarios (systematically) collected through real-world driving
- Individual human driver crash scenarios observed in the real world
- Systematic variation of generic human scenarios known to result in crashes involving human drivers (pre-crash scenarios)
- Systematic enumeration of road infrastructure variations present in the ODD of the object under test (feasible for limited ODDs)
- Informed brainstorm of challenging scenarios based on engineering knowledge of the system's weaknesses

Simulation commonly relies on particular scenarios (conditions to test the system in) described in some data format (e.g. OpenSCENARIO, 2017; OpenDRIVE, 2018; Hanke, et al., 2017). A challenge arises from the sheer number of scenario variations that can be constructed from each of the above sources due to the high number of variables involved (most of which are continuous). Even with continuous variables discretized, the possible number of combinations becomes practically infeasible to test. Adding to that, some of the influencing factors are random (e.g. sensor noise) and have to be captured by simulation. More details can be found in Wachenfeld, 2017.

As mentioned, deriving any statistical confidence from even an excessively large number of scenarios would require a solid argument about the representativeness of the scenarios and the accuracy of the simulation models. The approach of equivalence classes (see Section 3.4.1) could be considered.

3.5.3 Validating Simulation

As mentioned, any simulation comes with finite accuracy. Validating simulation aims to demonstrate that the simulation tools and models combined are accurate enough. This naturally raises the question of what accurate enough means. One may generally answer this question by asking whether eliminating the model simplifications used by the tool would alter the outcome of the test. This requires either testing against a more complex and realistic model or testing against real-world experience. In the hierarchical approach to the usage of the test platform, each level of simplification can be validated against the next higher level of sophistication, with the most sophisticated level validated against real-world driving. However, each level may introduce some degree of uncertainty into the validity of the simulation. Moreover, it will be practically infeasible to test the validity of the simulation across all possible corner cases. Instead, this publication proposes testing the validity of the full system simulation for a subset of corner cases against real-world experience. The confidence in the validity of the simulations across all corner cases needs to be increased to an acceptable degree by further validating the simulation models at the element level. The final confidence statement about the automated driving system safety should account for the remaining uncertainty about the validity of the simulation.

3.5.4 Further Topics in Simulation

The examples above have focused on the testing of the SAE L3/L4 automated driving systems in interaction with the surrounding (vulnerable) road users. Additional simulation tests may serve to test the wider vehicle ecosystem, including maps and infrastructure. For example, fleet simulation can be used to test backend functionality such as the algorithms used for calculating hazard warnings (see here, 2019) by sending notifications of virtual hazards to the backend.

Another specific role for simulation may be to estimate the system's behavior after a human takeover. Since real-world driving of a yet unvalidated system would require a safety driver to avoid exposing other road users to undue risk, the safety driver will take over before the automated driving system fails. Determining whether the crash would have resulted may include consideration of reprocessing results. Similarly, reprocessing may help to determine how different subsystems would have behaved (e.g. an automated emergency braking system), which could help to determine the performance of said subsystems.

3.6 V&V of Elements

To address key challenge 4, and due to the high number of combinations of factors and their concrete values (see Section 3.4), this publication suggests decomposing the system into subsystems and components to individually validate and verify these elements. This section explains in greater detail the specific V&V of each element listed in Section 2.2.

Some sections below are more detailed, because the V&V of these elements changes more for L3 and L4 systems. Regarding the elements and capabilities listed in Section 2.2, this publication focuses on elements that are verified and validated differently to L0–L2 systems. The elements with presumably no or minor V&V changes are:

PROCESSING UNIT

Typically, other SoCs and MCUs are used for L3 and L4 systems. However, the V&V methods are the same.

POWER SUPPLY

The power supply should be redundant. However, the single paths are tested as for L0–L2 systems. Additionally, switching from one supply to the other should also be tested.

COMMUNICATION NETWORKS AND BODY CONTROL

For the communication network and body control, the V&V methods and procedures are similar to those for L0–L2 systems.

EGOMOTION (INCLUDING ODOMETRY)

This should be more accurate for L3 and L4 systems. However, V&V of this subsystem is generally the same as for L0–L2 systems.

MOTION ACTUATORS AND BODY CONTROL WITH SECONDARY ACTUATORS

The actuators themselves are tested as for L0–L2 systems. Interaction with the system is tested for motion actuators as described in Section 3.6.5, in Section 3.6.7 for body control and secondary actuators and in the solution for key challenge 2 in Section 3.3.

3.6.1 A-Priori Information and Perception (Map)

Because reality is continuously changing, map V&V should be a continuous process over the service life of systems using the map. V&V of the map within this framework occurs at three levels:

- The map as a subsystem in automated driving operations
- The map as a holistic reflection of reality
- Specific map phases that might introduce errors (source, process and publication as identified in Section 2.2.2.2.1).

Automated driving system V&V is detailed in Section 3.3 and Section 3.4, which outline the testing that should be conducted to validate and verify the safety of the automated driving system in case of map/perception mismatch. This testing should focus on scenarios where map data is critical or less controllable for operation and/or a mismatch is predictable. Therefore, this section will focus on the map as a reflection of reality and the phases of map creation with regard to V&V.

From a safety perspective, the end-to-end V&V of the safety relevant map content (RMAs) should be verified by comparing the data to a reference dataset. The reference data should be constructed using a methodology to ensure the highest fidelity representation of reality possible at a given point in time. This enables direct end-to-end testing of the map and RMAs before incorporation into a full automated driving system for field testing.

The output from system tests that implicate the map as a possible source of error, particularly with respect to dynamic data such as traffic incidents, should be tracked and investigated as a broader result from fleet-vehicle testing. Some initial assessments can also be performed using fleet data from non-automated driving fleets. However, these results will primarily reveal the differences between systems, and reference data should again be employed for more quantitative assertions regarding correctness. Furthermore, system tests must be performed to ensure that the automated driving system is safe (e.g. statistical demonstration, requirement-based testing) in the case of map/real-world mismatch. Testing will be focused on scenarios where map data is critical or less controllable for operation and/or a mismatch is predictable.

However, different test methodologies should be applied within each phase of map creation (sourcing, processing and publication):

- Source data errors should be addressed using safety by design, as there is no reference data available and simulation cannot be used to detect source data errors in most cases.
- Processing errors should be addressed using a combination of safety by design techniques and traditional statistical assessment of a sufficiently large sample. Safety by design in this instance is primarily implemented through process analysis (e.g. in the form of FMEA, FTA and other such techniques).
- The confirmation of the effectiveness of measures concerning publication errors should be tested according to established testing methods, e.g. fault injection. However, process confirmation measures may support these steps, e.g. double publication of map, read/write confirmation of data transfer and appropriate tool qualification.

3.6.2 Localization (Including GNSS)

This section pertains to devices for determining the position of the vehicle relative to Earth surface coordinates. The input to the location system of the vehicle may comprise direct observation of global position (e.g. from the global navigation satellite system (GNSS)), local landmarks or information from V2X. This data is used in conjunction with other egomotion sensor data on the vehicle to ensure that the vehicle is positioned in an appropriate lateral position on the roadway and that curves in the roadway are appropriately anticipated with corresponding longitudinal speed adjustments. This is referred to as localization. The devices use GNSS. When the satellite system is not available, the vehicle systems defer to an inertial measurement unit (IMU) capable of measuring accelerations that are doubly integrated with respect to time to render position vs time. The IMU error is an error in acceleration. Thus, doubly integrating the error causes it to grow with the square of time. For this reason, IMU data is used briefly before it is reset. The error of satellite-based GPS data is generally time-independent, except for brief randomly distributed infrequent events for which the coordinate data are vastly incorrect. The sensors on the vehicle compare contextual information with the localization provided from Earth surface coordinates, which are imposed on map data, to determine whether localization from map/Earth surface coordinates data is usable. The performance specifications of GNSS devices should include both a target average accuracy and upper bounds on the frequency and duration of vastly incorrect estimates. The performance specifications of the IMU relate to an upper bound on the instantaneous acceleration error. The performance targets of the production system are achieved via extensive testing using ground truth systems that are usually close to ten times more accurate than the OuT.

In terms of V&V for the localization system with respect to functional safety and safety of the intended functionality, dedicated testing is needed to ensure that the vehicle's behavior is safe on the roadway. For instance, functional safety testing includes fault injection on the IMU or GNSS system.

3.6.3 Environment Perception Sensors, V2X and Sensor Fusion

Due to the high complexity of the real world and the insufficiency of purely synthetic perception input, the validation of this subsystem is based mainly on the (re-)processing of representative proving ground and open road datasets and subsequent comparisons with appropriate reference data (e.g. ground truth data). Synthetic perception input data (e.g. superimposed with various types of noise) will help to identify possible corner cases. Furthermore, proving ground testing can be performed for events that rarely occur in the real world. The following constraints should be fulfilled:

- (Re-) processing must be conducted within a validated environment (software and hardware reprocessing).
- A representative dataset includes not only a statistically significant amount of data (e.g. using the method of equivalence classes) but is also a sufficiently exhaustive description of the perception input within the ODD.
- Appropriateness of the reference data is assured if said data enables the assessment of the OuT, so that statements regarding the fulfillment of validation objectives can be derived. Therefore, the necessary contents of the dataset will vary depending on the OuT.

- Any dataset used to validate environment perception sensors, V2X and sensor fusion algorithms must be separate from the dataset used in development (see also Appendix B).

3.6.4 Interpretation and Prediction, Drive Planning and Traffic Rules

The input for the Trajectory Planner is an object list with specified attributes and parameters. Therefore, the necessary complexity is manageable with only synthetic inputs using readily available sensor and/or fusion models. This enables the use of V&V techniques that are mainly based on SiL of scenarios (as per Section 3.6). The SiL environment enables the use of search-based or reactive test approaches that allow for a highly efficient penetration of the parameter space. The aim of these simulations is to achieve a sufficient penetration of the relevant parameter space from a statistical point of view. To assess the behavior, an appropriate reference dataset is necessary. For example, a planned trajectory should not encounter a dynamic or static object on the road within a forecasted time frame. In addition to SiL, software reprocessing of open road data is highly recommended for the V&V of prediction and planning.

3.6.5 Motion Control

In this subsystem, the robustness against different variants of actuators, chassis, tires, aerodynamic, friction level must be considered. Two main approaches exist for this. Classically different variations are tested in the vehicle on the proving ground and later on open roads. Another approach is to simulate a multitude of parameter combinations to obtain the worst-case combinations. Additionally, a smaller number of tests are done in the vehicle to validate the simulation (see Section 3.5.3).

3.6.6 Monitor, ADS Mode Manager (Including the Vehicle State)

All the monitors (nominal and degraded performance, vehicle and user state, ODD) and corresponding state machines are tested at the software or component level and typically in software or hardware reprocessing. At the system level (e.g. the vehicle state monitor together with the actuator), tests are performed in addition on the proving ground and/or on open roads. Where appropriate, DiL instead of software or hardware reprocessing should be used for testing the user state at the system level.

The ADS Mode Manager (ODD determination, activation and deactivation state, including the User State Manager) is typically tested in SiL/HiL. At the system or vehicle level, the tests are carried out on the proving ground and on open roads.

3.6.7 Human-Machine Interaction

In general, this subsystem should be tested in DiL, and the vehicle on the proving ground and open roads. The basic behavior of the HMI (timing of the different interfaces) can be tested in the HiL to verify the basic HMI requirements. In the DiL and vehicle, HMI is validated by expert and thorough customer studies. These tests verify and validate the awareness of most drivers concerning the SAE automated driving level the system is currently in. The transition of the different SAE automated driving levels is also tested.

3.7 Field Operation (Monitoring, Configuration, Updates)

During the deployment and operation of the automated driving system and supporting functions, close coordination between field monitoring and configuration management is essential. Whenever changes are made to the automated driving system (e.g. hardware and software configurations or updates), validation should focus on the differences (delta) to the previously validated automated driving system (e.g. via regression testing). It is assumed that the companies involved with the development and field operation will follow the appropriate data management practices (e.g. the EU's GDPR) to account for privacy. The following sections describe the steps recommended for safe field operation of the automated driving system.

3.7.1 Testing Traceability

For successive software releases, a test plan must be assembled at the vehicle and element level that is traced to the capability and which provides insight to any regression observable at the vehicle and element level. Such traceability combined with the following procedure based on Figure 29 makes it possible to establish a relevant set of tests to run for every new configuration. The proposed cycle may be iterated more than once before the target system safety is achieved:

- Distinguish whether the changes influence the safety of the system.
- Then analyze which safety relevant parts are changed or influenced.

Reduce the number of influencing factors to be tested at the entire system level as follows:

- Test the influencing factors at the component/subsystem level and demonstrate robustness. Only factors influencing the entire system should be tested at the entire system level. As the test space (combinations of factors) is huge and statistical proof cannot be shown with a confidence of 100 %, a small residual safety risk is acceptable. To minimize the effects, the continuous field monitoring is highly recommended, especially at the beginning of operation.
- Check which test cases and scenarios need to be repeated and add additional tests.
- If the impact is not known in advance, the procedure must be carried out for at least a few test cases and repeated based on these results. Using this procedure during the development phase builds up substantial knowledge that is used for the road clearance of software and hardware updates.

The safety by design approach means that changes in certain elements do not affect the safety of the automated driving system (e.g. safe planner with a safety checker). The recommendation is to evaluate the risk and define the validation process for the change.

As the software is released on open roads, the exposure to a safety or security critical function or malfunction will rapidly increase with the number of vehicles in operation, requiring the fast and complete implementation of a response. In order to support a tightly coupled field monitoring operation, there are several focus areas that should be implemented to allow for the rapid discovery and correction of safety-relevant issues. These focus areas comprise test plans segregated by function and/or capability (nominal or degraded), a robust configuration and change management process, system analysis, regression prevention and enforcement of corrective actions.

3.7.2 Robust Configuration and Change Management Process

If a company is developing and producing safety-relevant software applicable to L3 or L4 automation, it is assumed for the purposes of this publication that the ISO 26262 standard or else some equivalent process and maturity is being implemented. With such software tools used to implement and enforce this process, it is possible to achieve the desired outcome of not only a reduced test set or plan, but also the support for multiple field variants that are specific to region, city, system configuration, or even individual routes based upon the ODD. To achieve this, it should be possible to describe the full system and all its software and hardware components using a unique identifier, and they should each be individually authenticated. Each of these variants should then be identifiable and traced by the operations group responsible for field monitoring (as discussed below).

As the system comprises software and hardware components, the test plan strategy mentioned in Section 3.7.2 provides the ability to test and define system safety based on hardware components and configuration changes. As it is foreseeable that a variant may be produced based on individual supported routes, a natural extension of this approach involves digital high definition maps. Map-related errors or malfunctions that could contribute to unsafe on-road performance must be tested with the appropriate countermeasures implemented to detect and mitigate these issues from manifesting to an unsafe system malfunction. It also becomes important to quantify the error rate or threshold for which the update frequency of the elements may be determined and enforced. System-level safety is closely coupled with the version of the elements that are deployed, and the configuration of the elements must also be matched with the version of the hardware and software that has been released.

3.7.3 Regression Prevention

To prevent changes that decrease system safety, there are several methods or approaches to a hardware and software maturity process that will assist in assessing the candidate for road release. The current approach to fulfilling requirements (see Section 3.1) from ISO/PAS 21448 is to define capability via a list of known known and known unknown operations to capture the unknown unknowns. These captured scenarios can then be used to protect against future regression in system performance via their inclusion in a simulation or the reprocessing of recorded data.

Additionally, new software features may be deployed to the OuT as it continues to operate in absence of failure. In this arrangement, the software being tested would be able to accept sensor inputs but would not have the authority to command vehicle actuation. This would provide the system integrator or designer with the ability to assess the performance of the software against the current configuration. The most difficult part of this approach is that it may be difficult to assess the performance of the newer software without a method to overlay or understand the function of the new software as if it had been able to affect the trajectory of the vehicle. This may be done on site, via the reprocessing of recorded data or possibly in simulation. This approach, when implemented with a fleet of test vehicles, reduces the potential exposure to voluntary customer participation.

3.7.4 Security Monitoring and Updates

The previous discussion about security in Section 2.1.5 discusses the processes and controls used to defend the systems and to find and fix vulnerabilities pro-actively. However, security efforts do not end after a first release is successfully evaluated. New attack techniques are discovered, and existing techniques continue to improve long after a vehicle has been built and sold. For these reasons, it is imperative to maintain a constant state of vigilance to detect and address new threats and previously undiscovered vulnerabilities affecting released systems.

The risks posed by highly automated vehicles lead to the conclusion that the ability to discover problems in fielded automated driving systems goes beyond what is typically implemented in current vehicles. Existing approaches such as threat intelligence and participation in the security community (e.g. Auto-ISAC, CERTs, conferences, etc.) remain important but are not enough. Automated vehicles require a level of security monitoring and information and event management that is more familiar to the IT industry. It is particularly important that the information required to (1) quickly discover new attacks against automated vehicles and (2) understand the underlying weaknesses that enabled the attacks can be collected quickly.

The statements mentioned in Section 3.7 also hold for security incidents. With these capabilities, automated vehicles will adapt with the threat landscape. To respond effectively, the means to quickly update released systems can be used also for cybersecurity. Furthermore, the lessons learned must be captured from these incidents to feed back into the development processes, helping to ensure that the products evolve to become more secure. The substantial re-use of automated driving systems in fleet vehicles and privately owned vehicles can contribute toward overcoming this challenge. Problems detected in more heavily monitored fleet vehicles should result in security fixes to privately owned vehicles that are based on the same driving system. This is important, as lightly monitored, privately owned vehicles are potentially more attractive targets due to reduced risk of detection. However, given the interaction between safety and security discussed in this publication, adequate measures will ensure vehicles are difficult to compromise and if they are compromised, it will remain difficult to cause a safety issue based on automated driving functions.

3.7.5 Continuous Monitoring and Corrective Enforcement.

Upon conclusion of the field monitoring and hardware/software change process, the new hardware/software will need to be distributed and applied to the fleet of vehicles it is intended for. These changes may be triggered by several actions, e.g. a planned system configuration change or increase in functionality, a requested safety or security-related change from a supplier or customer or a change initiated by a safety or security impact observed in the field. For each of these triggering actions (see Figure 31), there will be an internally assessed risk level associated with this proposed change. A simple example may be a scoring scheme from 1–4, as described in Table 9.

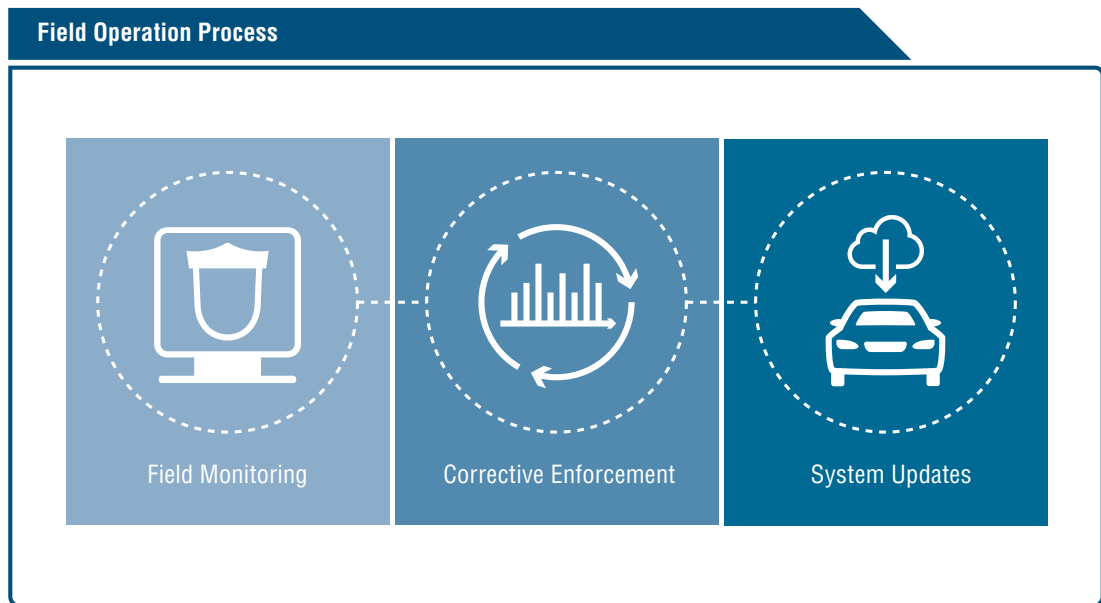


Figure 31: Field Operation Process

Corrective Enforcement		Safety Impact Score			
		1	2	3	4
Description	No safety or security impact with the planned function release	Potential safety or security impact without recognizable change in the functionality or HMI	Potential safety or security impact with recognizable change in the functionality or HMI	Immediate safety or security impact without proposed change or solution	
	No customer training	No customer training	Training may be required	Immediate customer notification, customer acceptance not required, training may be required	

Table 9: Corrective Enforcement

It is possible to automatically update all elements for all risk levels in Table 9. In addition, a customer notification is required for all Levels in Table 9. In the event of changes to rider interaction or the perceived functionality, an analysis of foreseeable misuse would indicate the possibility of the end user misunderstanding or receiving incorrect information regarding the capabilities of the updated system. To compensate for any risks in such cases, customer training would be prerequisite for the release of the software change. As the customer would be required to undergo training and confirm that they have done so, the function would be disabled even though it may be automatically updated. The final and highest risk level (L4) would result in the immediate disabling of the function or feature to contain the risk. For situations in which the function is disabled as a risk-based policy, the software could be downgraded to a previous version. However, it should be noted that in these cases the preferred method for traceability and configuration management purposes is not to downgrade the software to a previous version, but rather to use the same function/software version and to update the version ID instead. This workflow is illustrated in Figure 32.

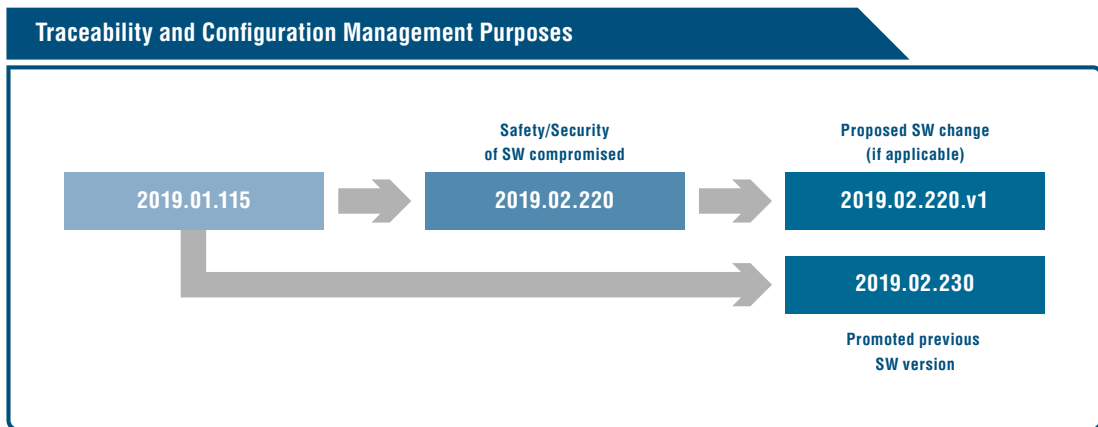


Figure 32: Traceability and Configuration Management Purposes

Chapter

04

CONCLUSION AND OUTLOOK

4 Conclusion and Outlook

This publication provides an overview of widely known safety by design and verification and validation (V&V) methods of SAE L3 and L4 automated driving, thereby creating a foundation for the development of automated driving solutions that lead to fewer hazards and crashes compared to the average human driver. An initial step involves deriving twelve principles from the overall goal of achieving this positive risk balance. Based on the twelve principles for achieving the positive risk balance, this publication devises a possible overall structure for the safety by design and the validation and verification process.

This publication establishes traceability between the top-level twelve principles and specific existing elements by introducing capabilities of automated driving based on the three dependability domains of safety by the intended functionality, functional safety, and cyber security. A generic architecture is outlined to connect the elements, and the architecture of the development examples is discussed. The suggested V&V approach combines safety by design and testing with the main strategies applied in V&V to solve the previously described challenges. The strategy addresses the key challenges, but it also shows that it is impossible to guarantee absolute safety with 100% confidence. Thus, there will still be some residual risks. Field monitoring is obligatory in order to iteratively learn and improve the systems.

In addition to the two main pillars of safety by design and V&V that underpin the twelve principles, Appendix B proposes a framework for a deep neural networks (DNNs) safety development. DNNs can be used to implement safety-related elements for automated driving. The framework includes recommendations for the safety-related design and artifact generation for each of the following three phases: The definition phase, the specification phase, and the development and evaluation phase. In addition, guidance is given regarding the deployment management aspects of a DNN with an emphasis on real-time field monitoring. Further steps include requesting feedback on this publication from all over the world in order to further develop this publication into an overall valid and accepted international standard.

This is not a one-off publication but should be viewed as a first version. The next version should expand the V&V process to include defined solutions with the necessary detail. This could be described via confidence levels and a combination of various testing methods and test results. The next version is intended to be put forward as a proposal for international standardization.



Chapter

05

**APPENDIX A:
DEVELOPMENT EXAMPLES**

5 Appendix A: Development Examples

This chapter demonstrates how the examples from Section 1.2 may be implemented in accordance with the derived generic logical architecture from Section 2.3. First, the four examples and their MRCs and MRMs are defined using the IDs introduced in Section 2.1.6. It is shown how, based on the specific MRCs and MRMs, the capabilities can be interpreted and implemented by the elements and what the resulting architectures may look like.

L3 TRAFFIC JAM PILOT (TJP)	
<p>NOMINAL FUNCTION DEFINITION</p> 	<p>L3 Traffic Jam Pilot (TJP) as an option for vehicle customers: Vigilant driver with driver's license, driving only on structurally separated roads, typically no pedestrians or cyclists, 60 km/h max, only with leading vehicles, no lane changing, no construction sites, only during daylight, without rain, only temperatures higher than freezing point</p>
<p>MINIMAL RISK CONDITIONS</p>	<p>TJP_MRC_1.1 Driver has taken over control.</p> <p>TJP_MRC_3.1 Vehicle is stopped in-lane. Note: In this example, TJP has no limited operation (MRC_2) minimal risk condition.</p>
<p>MINIMAL RISK MANEUVER</p>	<p>TJP_MRM_1.1 Hand over driving task to driver by issuing a takeover request and detecting takeover.</p> <p>TJP_MRM_3.1 Reduce speed until vehicle is stopped in-lane. Avoid collisions with leading vehicles by braking.</p>
L3 HIGHWAY PILOT (HWP)	
<p>NOMINAL FUNCTION DEFINITION</p> 	<p>L3 Highway Pilot (HWP) as an option for vehicle customers: Vigilant driver with driver's license, driving only on structurally separated roads, 130 km/h max, with and without leading vehicles, lane changing, construction sites, at night and during daylight, moderate rain and snow</p>
<p>DEGRADED MODE/ MINIMAL RISK CONDITIONS</p>	<p>HWP_MRC_1.1 Driver has taken over control.</p> <p>HWP_MRC_2.1 Vehicle is driving in-lane with speed reduced to 80 km/h.</p> <p>HWP_MRC_3.1 Vehicle is stopped in-lane.</p>

MINIMAL RISK MANEUVERS

HWP_MRM_1.1

Issue takeover request to driver.

HWP_MRM_2.1

Reduce speed to 80 km/h. Continue longitudinal and lateral vehicle control (avoid collisions and keep lane).

HWP_MRM_3.1

Reduce speed until vehicle is stopped in-lane. Continue longitudinal and lateral vehicle control (avoid collisions and keep lane).

HWP_RECOVERY_1

After HWP_MRC_2.1 has been attained due to reduced sensor vision, the system may return to nominal operation if all capabilities are restored, e. g. after the impaired sensor has been cleaned.

Figure 33 depicts a possible functional architecture of the Highway Pilot. It is created from the generic architecture discussed in Section 2.3. Redundant instantiations of relevant elements are introduced to enable the availability of degraded mode. The performance of the respective elements is adjusted to fulfill the capabilities in degraded mode.

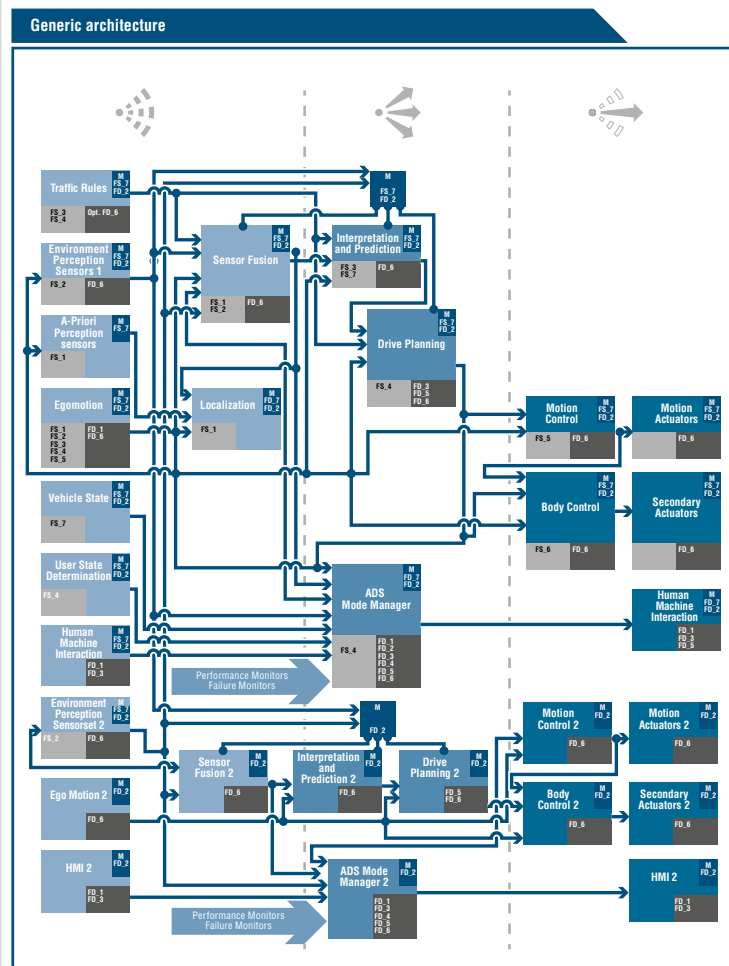




Figure 33: Generic Architecture

L4 URBAN PILOT (UP)

<p>NOMINAL FUNCTION DEFINITION</p> 	<p>L4 Urban Pilot (UP) in fleet operation in urban areas: Non-vigilant driver, not capable of driving, no driver's license necessary, max. 70 km/h max, large ODD with safety driver, very limited ODD without safety driver, allows indirect teleoperation if necessary</p>
<p>DEGRADED MODE/ MINIMAL RISK CONDITIONS</p>	<p>UP_MRC_2.1 Vehicle is driving in-lane with speed reduced to 15 km/h.</p> <p>UP_MRC_3.1 Vehicle is stopped in a safe location and secured; the (remote) operator is informed and decides on the course of further actions (e. g. towing vehicle).</p> <p>UP_MRC_3.2 Vehicle is stopped in-lane. Note: Because there is no driver present and no teleoperation intended in this example, UP has no (MRC_1) minimal risk condition for driver takeover.</p>
<p>MINIMAL RISK MANEUVERS</p>	<p>UP_MRM_2.1 Reduce speed to 15 km/h. Keep lane and avoid in-lane collisions by braking.</p> <p>UP_MRM_3.1 Reduce speed to 15 km/h. Stop at the next safe area (not on/in front of train tracks or in intersections). Inform the operator about the current state and position.</p> <p>UP_MRM_3.2 Immediate stop in current location. No collision avoidance.</p>

L4 CAR PARK PILOT (CPP)

<p>NOMINAL FUNCTION DEFINITION</p> 	<p>L4 Car Park Pilot (CPP) as an option for vehicle customers and in fleet operation: Driverless movement within certified parking structures or areas (no vigilant driver, no driver's license necessary), max. 10 km/h, ODD focus on off-street parking and logistic areas, scalable use of infrastructure (infrastructure not mandatory but possible up to teleoperation)</p>
<p>DEGRADED MODE/ MINIMAL RISK CONDITIONS</p>	<p>CPP_MRC_2.1 Vehicle is driving at crawling speed and avoids collisions.</p> <p>CPP_MRC_3.1 Vehicle is stopped in a safe location and secured; the (remote) operator is informed and decides on the course of further actions (e. g. towing vehicle).</p>
<p>MINIMAL RISK MANEUVER</p>	<p>CPP_MRM_2.1 Reduce speed to crawling speed. Do not enter intersections or ramps.</p> <p>CPP_MRM_3.1 Stop in a safe location and inform the remote operator (if available) or vehicle user.</p>

5.5 Selection of the Discussed Elements

The following sections discuss exemplary differences between the element implementations for the four development examples. Depending on the nominal function definition, the element requirements derived from capabilities may differ considerably. To highlight these possible differences the following element/capability combinations are outlined for the selected examples:

- Sensing elements requirements resulting from FS_1 Localization
- Sensing elements requirements resulting from FS_2 Perceive Relevant Static and Dynamic Objects
- Interpretation and Prediction element requirements resulting from FS_3 Predict Future Movements
- Acting elements requirements resulting from FS_5 Execute Driving Plan and FD_6 Perform Degraded Mode
- ADS Mode Manager element requirements resulting from FS_7 Detect If Nominal Performance is not Achieved and FD_4 React to Insufficient Performance
- User State Determination element requirements resulting from FD_1 Ensure Controllability for Operator
- HMI element requirements resulting from FD_1 Ensure Controllability for Operator and FD_6 Perform Degraded Mode
- Monitor element requirements resulting from FS_7 Determine if Nominal Performance is not achieved and FD_2 Detect when Degraded Performance is not available

5.5.1 Sensing Elements for FS_1 Localization

TRAFFIC JAM PILOT L3

The vehicle's location in the world is required to determine whether the vehicle is on the highway. Thus, road type classifications, e.g. via. vision sensors might be sufficient. Detection of highway-specific features such as traffic signs or features that indicate the vehicle is not on highway is possible.

HIGHWAY PILOT L3

Localization should determine the location of the vehicle on the map. Higher lateral than longitudinal localization precision is required. Localization needs to align the perception capabilities with map matching needs. For example, landmarks that are included in the map attribute need to be captured by vision sensors. Furthermore, GNSS can be used to determine location in cases where landmarks are not available. Additionally, fused outputs of active and passive vision sensors may be required to achieve precision and dependability.

URBAN PILOT L4

High lateral and longitudinal localization precision is required, e.g. to determine the precise remaining distance to intersections or stop lines. Thus, more attributes need to be available on the map.

CAR PARK PILOT L4

High lateral and longitudinal localization precision is required, e.g. for parking and maneuvering in tight spots. Due to poor GNSS performance within parking garages, localization should be based on map matching possibly specific features (e.g. artificial landmarks) within HD (indoor) maps.

5.5.2 Sensing Elements for FS_2 Perceive Relevant Objects

TRAFFIC JAM PILOT L3

Leading vehicles in front of the ego vehicle should be detected with the highest possible dependability. Lane markings are also relevant static objects. Even though (vulnerable) road users are excluded from the ODD, sensors should be capable of detecting ODD violations.

Diversity object detection methods are preferred to cover the performance weakness of single sensors. High-level object fusion is considered a meaningful measure.

HIGHWAY PILOT L3

In addition to the Traffic Jam Pilot, the following relevant objects should be detected with the highest possible dependability:

- Vehicles at large distances in front of and behind the ego vehicle, and vehicles at close distances in the adjacent lanes
- Obstacles in front of ego vehicles
- Road types, lane types
- Free space detection
- Remote hazard information
- Traffic signs such as speed limits

The map may be the only source of information for detecting some static objects. Radar and camera sensors could be used to detect dynamic objects, such as vehicles behind the ego vehicle. The capability of detecting objects could be improved if the V2X element is reliably available. Due to the increase in velocity between the TJP and HWP, the detection range of the sensor set to the front needs to be increased and sensor sets added to the side and back.

URBAN PILOT L4

Compared to the Highway Pilot, this scenario becomes much more complex and unstructured due to the:

- Variation of objects and their degrees of freedom to move (particularly (vulnerable) road users)
- High probability of occlusion
- Traffic guidance elements
- Additional infrastructure elements and layout

The sensor set capability should be enhanced to detect the above situations via:

- 360-degree coverage and increased elevation
- Additional redundancy and diversity to cover individual sensor weaknesses and increase overall performance
- Highly reliable detection of traffic guidance, e.g. traffic lights; if this cannot be achieved by environment perception sensors, the V2X element could be used.

CAR PARK PILOT L4

See the Urban Pilot. In addition, the following challenges may apply:

- Objects on or close to ramps
- Objects underneath the ego vehicle (e.g. following vehicle Wake Up where a-priori information is limited)

V2X could be used to increase perception performance, particularly in challenging scenarios that involve occlusions etc.

5.5.3 Interpretation and Prediction in FS_3 Predict Future Movements

TRAFFIC JAM PILOT L3

The ego vehicle could assume that the leading vehicle will remain in its current state unless deviations occur.

HIGHWAY PILOT L3

The current situation has to be interpreted before a complete scene description can be generated by combining the present world model and its predicted progression. This is true not only for interpreting a dynamic object's intention based on its classification but also for the current driving situation, which can also be classified. For instance, the future behavior of other (vulnerable) road users when driving in a traffic jam differs vastly to their behavior in flowing traffic. This classification of the current driving situation can be enriched by applicable driving laws. Combining the current classified scene with the intended behavior of dynamic objects (e.g. the probability of changing lanes) can then be used to predict future motion.

The sensed current world model as the output of FS_2 is not sufficient as an input for the collision-free and lawful creation of a driving plan (FS_4). Instead, it should be extended to reflect not only the current but also the estimated future state of the world model to generate a complete description of the dynamic driving situation or scene. The intention of all relevant dynamic objects has to be interpreted, as this forms the basis for predicting future motion.

URBAN PILOT L4

In this case, the Interpretation and Predict element has to take new (vulnerable) road users into account. For this development example, (vulnerable) road users may have a much more complex motion behavior than for the Traffic Jam Pilot or Highway Pilot, where the moving vectors are mostly aligned and are travelling in the same direction. In contrast, the moving vectors can be much more diverse in the Urban Pilot example. The interpretation and prediction model should take this into account.

CAR PARK PILOT L4

The challenges for this development example are comparable with those of the Urban Pilot.

5.5.4 Acting Elements in FS_5 Execute Driving Plan and FD_6 Perform Degraded Mode

TRAFFIC JAM PILOT L3

NOMINAL FUNCTION

Transform trajectory to a longitudinal and lateral vehicle movement up to 60 km/h. Realize a trajectory within given limits derived from lane, other objects and ego-vehicle width with the given and nominal performing actuators.

MINIMAL RISK MANEUVER

TJP_MRM_1.3: Immediately stop the vehicle with fixed deceleration, lateral vehicle movement based on last valid trajectory.

HIGHWAY PILOT L3

NOMINAL FUNCTION

Transform trajectory to a longitudinal and lateral vehicle movement up to 130 km/h. Realize a trajectory within given limits derived from lane, other objects and ego-vehicle width with the given and normal performing actuators.

MINIMAL RISK MANEUVER

HP_MRM_2.1: Transform trajectory to a longitudinal and lateral vehicle movement up to 80 km/h. Realize a trajectory within given limits derived from lane, other objects and vehicle width with the given and nominal performing actuators.

HP_MRM_3.1: Realize a vehicle stop with the last known valid trajectory with the available actuators. There is a certain risk that the vehicle will leave its lane, but this has a very low likelihood of occurrence. This mode is free of unreasonable risk.

URBAN PILOT L4

NOMINAL CAPABILITY

Transform trajectory to a longitudinal and lateral vehicle movement up to 70 km/h. Realize a trajectory within given limits derived from lane, safety distances to other objects, (vulnerable) road users and ego-vehicle width with the given and nominal performing actuators.

MINIMAL RISK MANEUVER

UP_MRM_2.1: Transform trajectory to a longitudinal and lateral vehicle movement up to 15 km/h.

UP_MRM_2.2: Transform trajectory to a longitudinal and lateral vehicle movement up to 15 km/h.

UP_MRM_2.3: Realize a vehicle stop with the last known valid trajectory with the available actuators.

There is a certain risk that the vehicle will leave its lane, but this has a very low likelihood of occurrence.

This mode is free of unreasonable risk. Ensure vehicle standstill.

CAR PARK PILOT L4

NOMINAL CAPABILITY

Transform trajectory to a longitudinal and lateral vehicle movement up to 60 km/h. Realize a trajectory within given limits derived from lane, other objects and ego-vehicle width with the given and nominal performing actuators.

MINIMAL RISK MANEUVER

Realize the last known valid trajectory with the available actuators. Degrade mode is transitioned into in the event of a failure. Based on its definition, this means that the vehicle will stop in its lane.

CPP_MRM_3.1: Stop in a safe location and inform the remote operator (if available) or vehicle user.

5.5.5 ADS Mode Manager in FS_7 Detect Nominal Performance and FD_4 React to Insufficient Performance

TRAFFIC JAM PILOT L3

Checks the activation conditions based on the input information. In this case, the vehicle is in a traffic jam on a highway and travelling at less than 10 km/h. It also checks the deactivation conditions to ensure that the vehicle has either reached a fail-safe state or that the user has safely taken over control.

The ADS Mode Manager switches to degraded operation based on the outputs of the Monitor.

MINIMAL RISK MANEUVER

TJP_MRM_1.1 and TJP_MRM_3.1: Deactivate as soon driver has control or the vehicle is stopped.

HIGHWAY PILOT L3

The change to the Traffic Jam Pilot is tied to the ODD specifics. In this case, the vehicle is on a highway and travelling at less than 130 km/h.

MINIMAL RISK MANEUVER

Select the appropriate MRM. For example, reduced sensor performance due to reduced visibility leads to HWP_MRM_2.1. Reaching the end of the ODD leads to HWP_MRM_1.1 or HWP_MRM_3.1 to ensure either a takeover by the user or a safe stop at the end of the ODD.

URBAN PILOT L4

This could mean that the vehicle is inside a geofenced area, for example. It also checks the deactivation conditions to ensure that the vehicle has reached a fail-safe state. Additional states and transitions should be introduced for the option of operating the vehicle by a remote operator. The ADS Mode Manager switches to degraded operation based on the outputs of the Monitor.

MINIMAL RISK MANEUVER

Select the appropriate MRM. For example, reduced Localization sensor performance leads to UP_MRM_2.2. Cases where driving cannot continue due to a blocked lane or a solid lane marking lead to UP_MRM_2.2. Switch to UP_MRM_2.3 once a rear-end-collision has been detected. Secure the vehicle as soon as a full vehicle stop has been reached.

CAR PARK PILOT L4

Checks the activation conditions based on the input information. In this case, the vehicle is in a parking lot or logistics area, the vehicle perception signals nominal parameters and there is no driver present. It also checks the deactivation conditions to ensure that the vehicle has either reached a fail-safe state or the user has safely taken over control of the vehicle. The ADS Mode Manager switches to degraded operation based on the outputs of the Monitor.

MINIMAL RISK MANEUVER

Ability of a product to deliver a function, feature or service mode based on the failure.
Switch to an appropriate degraded mode based on the failure.

5.5.6 User State Determination in FD_1 Ensure Controllability for Operator

TRAFFIC JAM PILOT L3

The vehicle operator is the user in the vehicle. Indicates the current ability of the user to take over the driving task immediately after requested to. Examples include whether the user's eyes are open and whether the user is sitting in the driver's seat.

HIGHWAY PILOT L3

The vehicle operator is the user in the vehicle. Potentially no increase to the Traffic Jam Pilot.

URBAN PILOT L4

In this case, there may be two operators who require consideration:

- User in the vehicle: Indication of whether vehicle users are interfering with the driving functionality is necessary.
- Remote operator: Monitoring the remote operator is not necessary, because they are considered to be a trained expert.

CAR PARK PILOT L4

In-vehicle HMI is not necessary while the function is activated, because the user is not required to take any action. Thus, HMI can be used for informational purposes. Two other operators could be present:

- User in the vehicle: Indication of whether vehicle users are interfering with the driving functionality is required.
- Remote operator: Monitoring the remote operator is not required, because they are considered to be a trained expert.

5.5.7 HMI in FD_1 Ensure Controllability for Operator and FD_6 Perform Degraded Mode

TRAFFIC JAM PILOT L3

The HMI explicitly displays the current level of automation (system state) to the user. This is important for communicating the degrees of freedom, and responsibilities to the user. Furthermore, the HMI elements communicate takeover requests to the user.

The HMI detects when the user undertakes deliberate action to activate or deactivate the Traffic Jam Pilot or to accept a takeover request.

HIGHWAY PILOT L3

No additional requirements.

URBAN PILOT L4

The HMI aspect refers to navigational interaction. Interaction to initiate an immediate stop is considered a navigational interaction.

CAR PARK PILOT L4

In-vehicle HMI is not necessary while the function is activated, because there is no driver present.

5.5.8 Monitors in FS_7 and FD_2

The Monitors should monitor the error states of the elements. The main differences between the Monitors in the development examples are the number of elements, their properties to be monitored and the number of possible error states. That leads to an increase in interfaces to the monitor layer.

TRAFFIC JAM PILOT L3

The Monitor should monitor the performance of the front sensor, the driver's state, the deceleration elements and the power supply.

HIGHWAY PILOT L3

The Monitor should also monitor the performance of the additional sensors and the driving dynamic elements (e.g. steering or braking). This expanded scope means that a larger set of sensors and actuators need to be monitored.

URBAN PILOT L4

In this case, there is the additional need to monitor the energy resources to ensure a longer operating period. The User State Determination may no longer need to be monitored.

CAR PARK PILOT L4

In this case, there is the additional need to monitor the energy resources to ensure a longer operating period. The User State Determination may no longer need to be monitored.

Chapter

06

APPENDIX B

**USING DEEP NEURAL NETWORKS TO
IMPLEMENT SAFETY-RELATED ELEMENTS
FOR AUTOMATED DRIVING SYSTEMS**

6 Appendix B: Using Deep Neural Networks to Implement Safety-Related Elements for Automated Driving Systems

DISCLAIMER: The aim of this chapter is to provide an overview of the challenges for achieving and assuring the safety of DNNs in automated driving, propose potential solutions that address the safety challenges, and conduct a brief survey on the current state of the art regarding these challenges (with no claim of being exhaustive). This chapter does not provide a complete solution, but instead proposes potential solutions that can be used as guidance for the development of supervised deep learning. The aspects outlined here may be revised and updated continuously in the future, depending on advances in research and application.

6.1 Motivation and Introduction: Machine Learning in Automated Driving

Machine learning is a set of tools that enables computers to learn a task by using data and not by being explicitly programmed or defined through human-understandable rules. Due to their powerful performance, machine learning algorithms are becoming more widespread, and machine learning is seen as a crucial technology for automated driving systems (Bansal, Krizhevsky, & Ogale, 2018). Consequently, the development process for machine learning algorithms responsible for executing safety-related tasks of automated driving systems must undergo strict assessment.

Established safety engineering processes and practices have been successfully applied in traditional model-based system development. These processes and practices are also described in the two automotive safety standards ISO 26262 and ISO/PAS 21448. However, the safety standards available within the automotive and any other industry have been defined without explicitly considering the specifics of machine learning algorithms such as dataset collection and its requirements, defining performance evaluation metrics, handling uncertainty, etc. (Salay & Czarnecki, 2018). This leads to a challenging issue today for automated driving system manufacturers and suppliers who are determined to incorporate machine learning for automated driving.

(Deep) neural network concepts evolved from probabilistic modelling, which incorporates random variables and probability distributions to model a situation or event. While a deterministic model returns a single possible outcome for an event, a DNN model returns a probability distribution as an output. These models take into account the fact that it is rarely possible to know everything about a situation. In machine learning in general, the main idea is to parameterize the function by the data. The quality of the dataset, which is ultimately used to learn the parameters, should be continuously improved in order to find an optimal model. Another dimension of model optimization is the choice of the DNN architecture.

Machine learning algorithms infer results (output) from data (input) from a previous process called training, which can either be supervised or unsupervised learning (Hinton & Sejnowski, 1999). In supervised learning, the machine learning model is presented an input and the desired output at training. This means that the data is already labeled with the correct answer. Unsupervised learning algorithms are trained using a dataset that does not have any labeling at all. The unsupervised learning algorithm is never told what the data represents, and the goal of the training is to automatically infer structure from the data and discover new dependencies or patterns. Reinforcement learning is the third paradigm of machine learning and is similar to unsupervised learning in that the training data is unlabeled (Sutton & Barto, 1998). This machine learning model learns via reward or penalty feedback received based on the interaction within the environment.

This chapter focuses only on supervised learning for DNNs (Schmidhuber, 2015), because these are most commonly used in automated driving, and its scope excludes end-to-end DNN approaches (e.g. a DNN is trained to infer the control commands directly from raw sensor data, see (Bojarski, et al., 2016)). The chapter is structured according to Figure 35. For greater clarity for developers and assessors, this publication recommends defining a modular-based system architecture (ISO 26262) in which machine learning algorithms are used as a software component. A typical example of such a component is 3D object detection (see also the Environment Perception Sensors from Section 2.2.2). 3D object detection (Arnold, et al., 2019) based on a DNN is used as an example in order to easily grasp the concepts described in this chapter. These algorithms infer objects represented by bounding box position coordinates and dimensions together with a label of the object class (e.g. car, pedestrian) from images and/or LIDAR point clouds (see Figure 34).

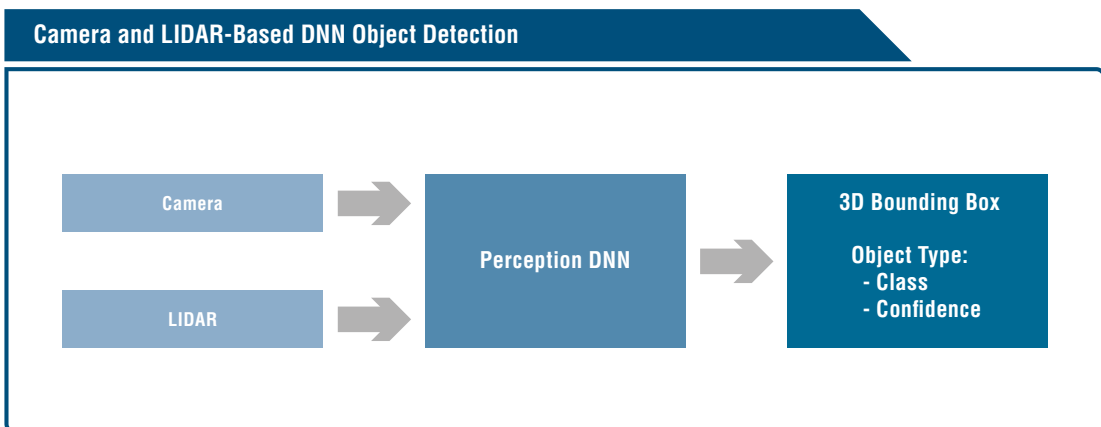


Figure 34: Camera and LIDAR-Based DNN Object Detection

Define, Specify, Develop and Evaluate, and Deploy and Monitor are the development steps of DNNs and provide the safety artifacts that support the safety case (see Figure 35). These steps and safety artifacts are discussed in greater detail in the sections below.

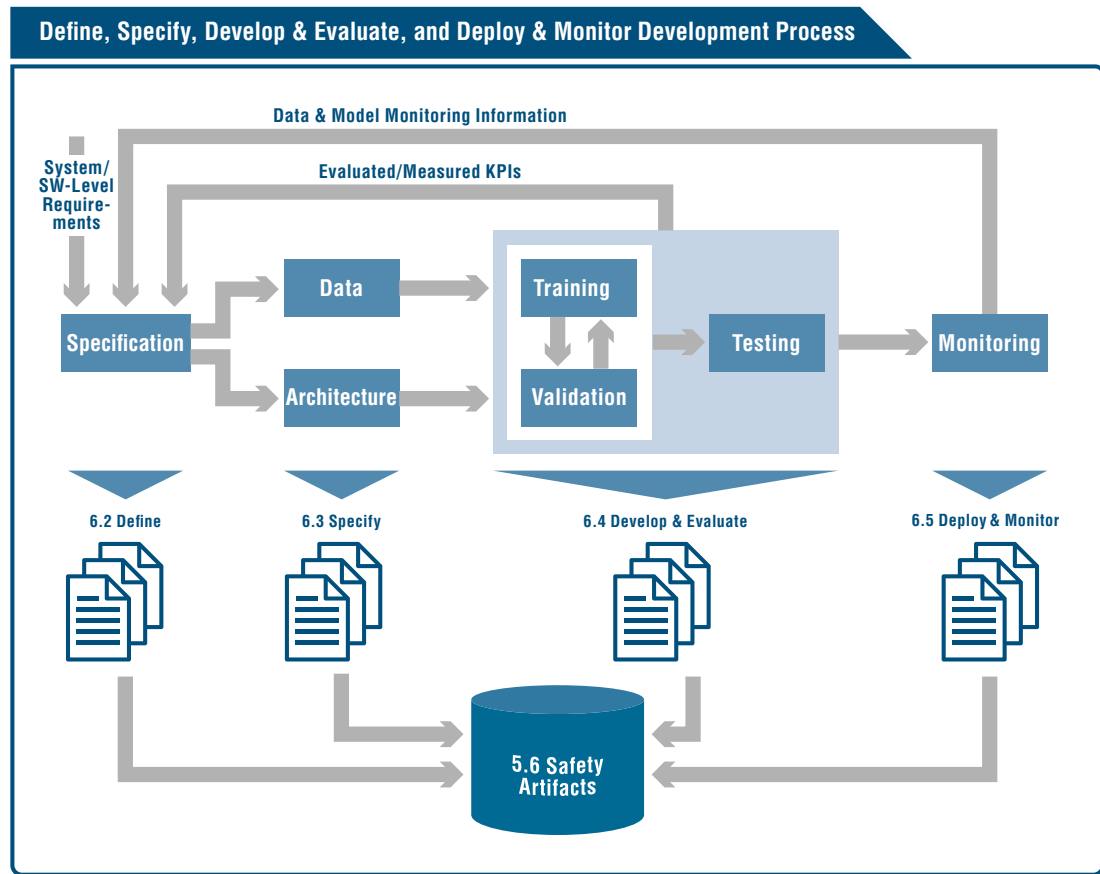


Figure 35: Define, Specify, Develop & Evaluate, and Deploy & Monitor Development Process

6.2 Define (What and Why)

Using a modular design approach, the requirements of the software components that form the automated driving system must be defined (inputs, outputs, technical safety requirements, software safety requirements, functional requirements, etc.). The specification process has to be adapted when specifying a DNN. The additional influencing factors that need to be defined to create a robust and safe DNN include specifying the ODD, the outputs (adapted to a probabilistic output), the dataset attributes critical for the DNN objective and the identification of the relevant safety requirements interpreted with measurable KPIs.

Updates to the requirements and artifacts concerning the data-driven development process are anticipated. For example, a training specification should be developed, and data-specific information should be included in the artifact requirements.

Possible Specification Characteristics to Consider when Defining a DNN				
ODD	Dataset Attributes	Probabilistic Output	KPIs	Hardware
Weather conditions	Characteristics of the target classes	Confidence	Robustness measure	Memory footprint
Geographic domain	Labeling classes	Temporal properties	(Class-wise) performance measures	Latency (timing)
Background scene	Labeling quality		Confidence quality	Optimization
Dynamic properties of the scene	Data coverage		Reproducibility	Sensor calibration

Table 10: Possible Specification Characteristics to Consider when Defining a DNN

Regarding the use case of 3D object detection, the characteristics outlined in Table 10 may be considered.

The following questions can be used as guidance in developing a specification of the 3D object detection function:

- How many different types of objects are necessary for the function to reliably detect in the environment?
- What is the required detection rate?
- Is it necessary to detect the object in the ego lane, in the adjacent lane, on the shoulder?
- What is the correct specification of the object classes (e.g. size, shape, color, etc.)?
- What is the ODD for the function?
- How much data is available for training, validation and testing?
- What methods would be practical and necessary for collecting the object detection data under consideration of sensor calibration information?
- What is the target platform, CPU or GPU, and the performance restrictions for the detection algorithm?

This list is not exhaustive and further questions may apply. At the software architecture level, special care should be taken when mapping the safety goals and requirements to measurable and reachable KPIs to train DNNs for automated driving systems, and when evaluating the safety of the resulting DNN models.

For example, such KPIs cover:

- Performance of a DNN on a testing dataset with respect to the safety goals
- Robustness of a DNN against perturbations and, in particular, adversarial attacks
- Understandability of a DNN
- Sanity of the resulting DNN-based software component
- Latency of the resulting DNN-based software component
- Generalizability of the DNN to unseen data within the ODD

The following artifacts are expected from this phase of development:

- Dataset specification (specification of the global dataset attributes)
- Labeling specification (specification of the classes, boundaries, labeling guidelines)
- DNN requirements specification (specification of the ODD, functional objective requirements, technical safety requirements, etc.)
- KPI specification (measurables such as dataset coverage, algorithm robustness, dataset quality, etc.)

6.3 Specify (How)

Once the intended functional requirements and important characteristics have been defined, the dataset can be specified and the DNN architecture designed.

6.3.1 Defining and Selecting the Data

A DNN-based component is developed using three disjoint datasets: Training, validation and testing (Figure 36). Models are fitted using the training dataset, while the validation dataset is used during the training process to verify the quality of the current fitting. The testing dataset is used to verify the performance of trained models after training has finished. All three datasets are carefully constructed from a finite dataset of input and output pairs matching the attribute requirements from the Define phase. The datasets should sufficiently cover the input domain. The datasets should also be highly representative and complete, particularly regarding corner case inputs such as object detection of a pedestrian at night or during bad weather conditions. For example, the datasets for a 3D object detection algorithm (that includes pedestrian as a class) should have enough heterogeneous examples of pedestrians in such challenging environments. Furthermore, the datasets should include a measure of negative data for the main purpose of allowing the machine learning module to understand *what is not* in order to reduce false alarms.

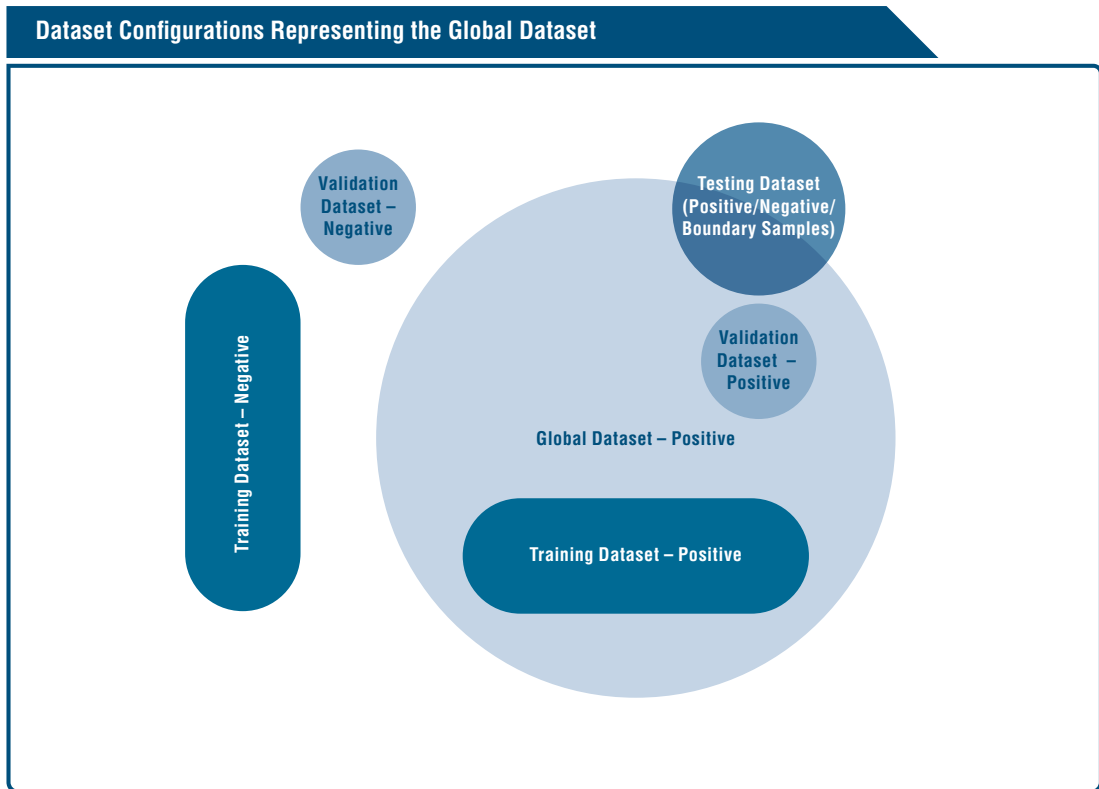


Figure 36: Dataset Configurations Representing the Global Dataset

A uniformly distributed dataset is recommended, and the problem is multidimensional. For the example of a 3D object detection algorithm, the objective may be to detect and classify pedestrians, vehicles and bicycles. The statistical distribution should be considered for the classes and separately for the data attributes within each class, defining the class itself as well as environmental attributes that may be encountered within the ODD (see Table 11).

Example Attributes for 3D Object Detection			
	Class		
	Pedestrian	Vehicle	Bicycle
Class Attributes	<ul style="list-style-type: none"> ▪ Size ▪ Position ▪ Pose ▪ Clothing ▪ ... 	<ul style="list-style-type: none"> ▪ Size ▪ Position ▪ Color ▪ Type of vehicle ▪ ... 	<ul style="list-style-type: none"> ▪ Number of wheels ▪ Orientation ▪ Human presence ▪ Attachments (trailer) ▪ ...
Environmental Attributes	<ul style="list-style-type: none"> ▪ Background colors (trees, buildings, ground cover) ▪ Occlusions ▪ Weather ▪ Lighting ▪ Adversarial perturbations ... 		

Table 11: Example of Attributes for 3D Object Detection

A DNN model requires data containing information relevant to the scenarios defined by the ODD. The following is a minimal set of quality metrics that are important for quantifying the sufficiency of the dataset:

- Coverage
- Relevance
- Equivalence classes (negative and positive examples)

The dataset should be continuously improved as new scenarios are discovered, reducing the unknown space. Over time, the characteristics of the data might change in the operating environment, and so the dataset should reflect this to ensure the DNN-trained model remains accurate. As insufficiencies are found with the diversity of the dataset attributes, a data collection campaign may be necessary. Data may be collected using various methods such as campaigning, fleet services, individual data recording and the use of 3rd party datasets. In order to enable traceability and separation between dataset splits, data management should include the concept of bookkeeping and tagging. Tagging is crucial for recording information such as location, weather, sensor parameters, etc. Such information allows the data to be transformed as needed. Collecting data can be enhanced for rare cases utilizing different techniques, e.g. augmentation or synthesis. However, real data should be present and dominant to ensure safety.

DATASET LABELING

An expert should carefully define the labeling specification to ensure that the labeling characteristics are defined sufficiently and can efficiently relate to the target task.

There are many approaches for tagging. It can be carried out manually by human annotators or semi-automatically where, for example, DNNs first try to detect objects and then human annotators correct the results. Another common practice when labeling time sequences is to apply tracking algorithms to follow objects in a scene automatically, so that human annotators do not need to label frame by frame.

Quality control processes should be in place to ensure data is properly labeled, regardless of the labeling method used, to ensure error injection caused by the labeling process is minimized. Typical labeling errors in the case of 3D object detection would be:

- Incorrect classification of objects
- Overseen objects
- Wrongly positioned bounding boxes
- Bounding boxes with the wrong size or pose
- Split bounding boxes due to partial occlusion

It is recommended to carefully choose the set of labeling classes: If the concepts are difficult to separate (e.g. “child” from “grown-up person”, “commercial vehicle” from “truck”), DNNs will perform poorly. If the concepts are chosen too coarsely (e.g. just “dynamic object”), subsequent modules will encounter problems in reacting safely. Moreover, DNNs can perform safely only if the underlying training dataset is consistent. Such consistency can be reached only by clearly defining the limits of labeling classes (e.g. does a Segway belong to the class “person” or “cyclist”?). Compliance with these limits has to undergo targeted quality assurance.

The following artifacts are expected from this phase of development:

- Refined labeling specification
- Refined dataset specification
- Labeling quality report
- Labeled dataset (representative global dataset including the data splits for training, validation and testing datasets)
- Dataset KPI report (measurables such as dataset coverage, algorithm robustness, dataset quality, etc.)
- Scripting tools (dataset creation, labeling, measurement of KPIs, etc.)

6.3.2 Architecture Design for DNNs

An architecture design should be developed to meet the requirements of the characteristics described in the Define phase in Section 6.2. This can be achieved by considering different architectural design patterns at the software architecture and DNN architecture levels (see Figure 37), which are described below.

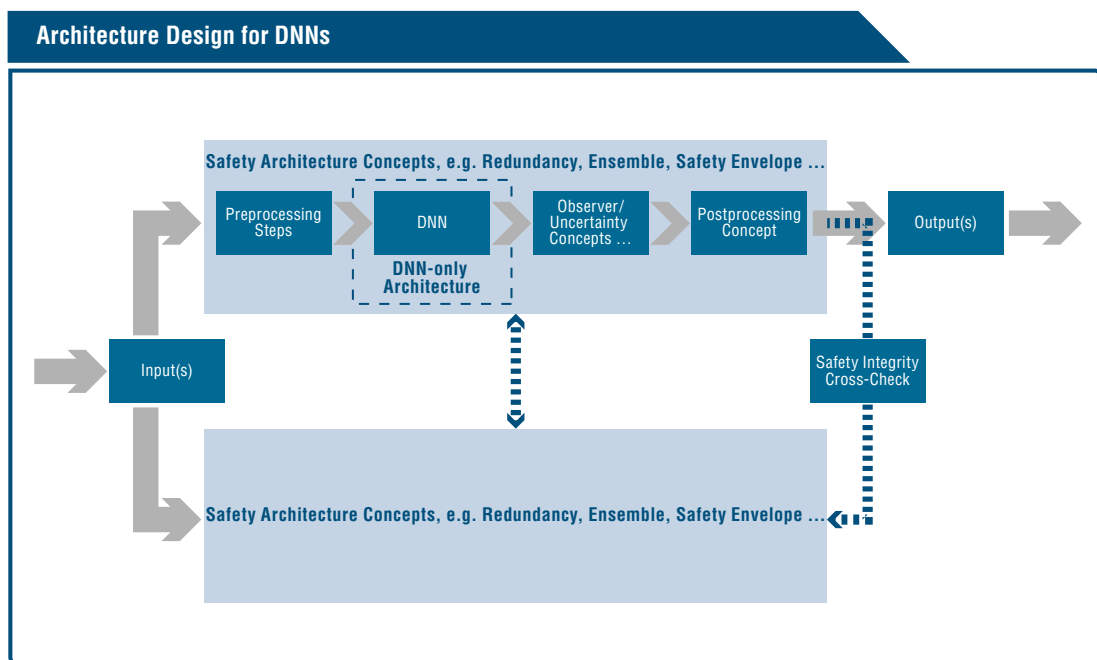


Figure 37: Architecture Design for DNNs

DNN-BASED SOFTWARE ARCHITECTURE LEVEL

Software architecture incorporating a DNN should be able to handle or mitigate the unexpected behavior of the DNN. For instance, the software architecture may provision for the implementation of concepts or components that measure the internal state of a DNN and/or enable observation of the output of the DNN at inference time. Two possible concepts for this include uncertainty and observers:

- **Uncertainty:** There are two different sources of uncertainty, the first being aleatoric uncertainty, which arises from the uncertain nature of the real world itself, and the second being epistemic uncertainty, which is model-related and could in principle be reduced by replacing the model (e.g. via more training data, changed training strategies, etc.). Aleatoric uncertainty is inherent to nature itself and could be a possible input to a driving function to increase safety. In contrast, epistemic uncertainty is a possible input for data selection and the choice of training strategies. Existing literature adopts different approaches for measuring uncertainty at inference, e.g. MC dropout (Gal, 2016; Esser, Sutter, & Ommer, 2018).
- **Observers:** Observers are additional software components working in parallel to the DNN itself to supervise its behavior with reference to safety. Such components could include:
 - Plausibilization methods that check the model output for consistency (e.g. checking for implausible positions, sizes, dynamic properties of detected and tracked objects).
 - Input observation methods that check the input is statistically close to the training dataset.
 - Saliency maps that check for plausible sensitivity of the DNN toward regions in the input.

Other additional software architectural approaches that enable the overall system to meet the safety requirements may include applying the following strategies:

- Redundancy and ensemble concepts, wherein the inferences of several DNN models are used to generate an aggregated result. In addition to ensemble inference from many DNNs, a simple rule-based approach can also be applied to perform basic safety functionalities.
- Detection (e.g. plausibilization) mechanisms for abnormal neural behavior or abnormal input data can be used to identify situations of possible failure.
- Attention mechanisms (e.g. heatmap) can be employed to steer the impact of a particularly critical input on the model's output.
- Safety envelope concept, with a typical example being the doer/checker approach (Koopman & Wagner, 2018). This generally attempts to specify the safe and unsafe region. A doer concept normally operates in the safe region. Once the output of the doer drops below the safety integrity level, the checker function is activated. The boundaries between the doer and checker transition need to be specified precisely to avoid any major system failure.

The overall DNN-based software architecture may also include input preprocessing steps such as resizing, resampling, etc. These steps need to be a part of the architecture so that the interface of the DNN architecture can be adapted. In a similar way, post-processing steps may be required to ensure the DNN output is compatible with the output interface of the subsequent subsystem.

DNN-ONLY ARCHITECTURE LEVEL

This architecture level focuses on the DNN itself. The type and combination of DNN layers in the architecture should be configured in accordance with the use case, and the specification requirement defined in the Define phase. On the DNN architecture side, deciding factors may include the input and output (data types and dimension) as well as the decision regarding the size and category configuration of the model. Furthermore, the activation function also needs to be selected carefully, as it plays an important role in function approximation. This can also speed up the convergence of the DNN model. Various other aspects to consider can include architecture at the DNN architecture level such as the type of pooling layer, the use of striding and the use of recurrence, etc. Moreover, this can be further modified in the Develop and Evaluate phase based on the generalization of the network.

The following artifacts are expected from this phase of development:

- Architecture specification (specification defining the chosen DNN design architecture to solve the objective defined for the system)
- Code and objective of uncertainty and observers
- Report on additional mechanisms to reach safety requirements

6.4 Develop and Evaluate

Having defined the function by means of requirements (i.e. the actual DNN through a corresponding dataset and the model architecture together with specific functional and non-functional DNN KPIs to be reached), the DNN has to be trained, optimized, evaluated and integrated into the overall automated driving system before a final safety argument can be carried out. This section covers the steps before this integration.

The parameterization of a DNN model using labeled data (training) is defined by the loss function that measures the differences between the model outputs and the labels in a specified manner (e.g. cross entropy, mean squared difference, etc.). After averaging the error over (randomly selected) training dataset samples, the model parameters are changed through back propagation of the corresponding gradient (e.g. stochastic gradient descent) aiming at the DNN model to minimize the training loss. The choices of the loss function and possible regularization could have a strong impact on the robustness of the resulting network. Therefore, some restrictions in the choice of possible loss functions for the training phase may need to be specified. Loss function is related to the data's statistical distributions. For example, the L2 norm and L1 norm of distance metric implement different probability distributions of data.

As previously stated, the loss function traditionally aims at maximizing the correctness of a DNN model. It is the key component for the training of a DNN, as it specifies the learning goal. It is important to note that the safety requirements (e.g. reliability, robustness, time stability, criticality of particular error types, etc.) are not necessarily being taken into account when designing the loss function, and so the trained model should be tested against the safety requirements. Possible solutions to ensure safe functioning could include the injection of measurable safety requirements into the loss function (using additional terms in the loss function to compensate for safety-related fitting goals).

In addition to this, a set of hyperparameters need to be specified for training:

- Concrete types of layers (type of pooling, type of up-sampling, hyperparameters for convolutional layers)
- Regularization terms (batch normalization, drop out)
- Update parameters (solver, learning rate, batch size).

All these choices will need to be tracked, as they influence the resulting functional and non-functional properties of the DNN.

The training & validation process is iterative as depicted in Figure 35. A trained DNN's knowledge is limited to the examples it has seen during training. The validation and testing datasets should adequately cover the space of possible inputs to obtain a better understanding of the actual performance of the model. Moreover, the validation and testing dataset should contain data that has never been shown to the DNN. The quality of the resulting network is indicated by its performance on a validation dataset (usually measured by means of performance KPIs such as intersection over union, mean average precision, false positive/negative rate, etc.). DNNs might fail for some validation data, in which case the failed data should be supplemented with more data representing those failure cases and added back to the training dataset to improve the DNN's performance. Furthermore, the DNN might fail for rare cases that are underrepresented in the data. How the DNN would perform on such unseen and underrepresented data is very important when studying failure cases.

Once DNN failure cases have been identified, training and potentially the model and the dataset should all be adapted. This is possible via a variety of means, e.g. expanding the dataset, changing the network architecture, changing the hyperparameters mentioned above (learning rate, batch size, batch normalization, regularization, activation functions, optimization methods, etc.). Given the nature of deep learning, the primary method is the expansion of the training dataset while respecting the relevant statistical distributions.

It is recommended to initially test the DNN on an application-specific dataset collected by the target sensor setup to discover failure cases, and to use software and/or hardware reprocessing to emphasize the failure cases during testing. The testing dataset should be independent of the validation and training dataset. The software and/or hardware reprocessing testing data will inherently include negative data to test whether the DNN would generate false alarms. During durability runs, data should be collected for validation and training in case failures are identified.

Transfer learning attempts to pass knowledge learned from one DNN to another DNN for potentially different tasks. The difference could arise from sensors, location, other datasets, etc. Transfer learning is often used to speed up the learning process. However, it could negatively impact functional safety. The use of any type of transfer learning should be disclosed and documented. The DNN should go through the same validation and testing process to discover failure cases, and subsequent re-training.

Usually, DNN optimization is performed after successful training to reach runtime and memory footprint requirements. DNN optimization is an iterative process whereby compression steps and fine-tuning steps (short training phases of the changed model using the training dataset) are intertwined. It is important to note that the choice of concrete optimization methods (e.g. level of quantization, ranking metric for pruning, etc.) considerably impacts the robustness of the resulting model. Therefore, the tracking of functional and safety focused KPIs is highly relevant. A balance should be found to achieve both the previously defined KPIs and the system hardware KPI constraints. Thus, a different performance can be expected depending on the optimization target.

There is an abundance of possible optimization and compression technologies to choose from:

- Quantization of DNN parameters and computations (reduction of parameter bandwidth)
- Pruning (removal of less important parameters from the DNN model)
- Student-teacher compression
- Hardware-specific compute acceleration

Together with the evaluation of relevant KPIs and their correlation with the safety requirements, measures and tools should be developed. These measures can involve:

- Computations based on the model output or internal state (e.g. output variance, neural activation patterns)
- Comparisons of the model output with other data (e.g. labels, different model output, etc.)
- Modification to the input or output data stream (e.g. addition of noise)
- Modification to the internal network structure (e.g. dropout)
- Backward propagation of information through the networks (e.g. gradients, inverse activation)
- The usage of additional software or knowledge sources (e.g. priors in the dataset, additional cost functions, geometrical output analysis)

The following artifacts are expected from this phase of development:

- Refined labeled dataset
- Chosen hyperparameters (architectural decision points, random seeds, etc.)
- Baseline of training (model parameters, hyperparameters, data points for each training step)
- Intermediate validation and testing report
- Code and evaluation report for DNN observers
- Chosen optimization and compression methods, parameters and baselines

6.5 Deploy and Monitor

Due to the infinite nature of the dynamically changing environment, the characterization of the behavior of an automated driving system is not finite, therefore runtime monitoring becomes necessary. Runtime monitoring of DNNs is one process that needs to be planned and executed during deployment. The following challenges should be understood when defining the runtime monitoring approaches of the DNN input-output mapping:

CHALLENGE 1

Being a probabilistic model and trained in a supervised manner, DNNs cannot detect unknown unknowns (DNNs are trained to provide an output to a given input).

CHALLENGE 2

During training, DNNs are presented high confidence labels and tend to replicate this high confidence even in unclear situations.

CHALLENGE 3

DNNs do not necessarily base decisions on semantically meaningful features.

CHALLENGE 4

The design of the DNN assumes a certain statistical distribution of input features (defined through the training dataset). Therefore, the performance of the DNNs tends to change even under minor changes to the input distribution (distributional shift).

To ensure the safety of deployed systems, the above challenges will have to be considered and measures should be taken accordingly. This can be achieved via the runtime monitoring of:

THE INPUT DATA

- Checking the operational domain for distributional shifts (i.e. a significant drift within the feature distribution relative to the training dataset). This addresses Challenge 4.
- Checking for new concepts (e.g. new objects, different behavior, new rules, etc.) as these would be unknown unknowns from the perspective of the initial training. Naturally, the output of a trained model on such new concepts should be considered less dependable. This addresses Challenges 1 and 2.
- Changes occurring in the world (domain drifts, new objects, changing rules) necessitate ongoing data measurement campaigns. The additional data needs to be selected with the goal of maximizing the safety of the intended functionality, considering the existing dataset and the behavior of the model. This poses a difficulty with respect to the correct amplification of rare cases during the overall training process. This addresses Challenges 1 and 4.

THE MODEL BEHAVIOR

- Using the observers from Section 6.3.2 at runtime can help when interpreting the output of the network. This addresses Challenge 3.
- In order to prevent systematic DNN insufficiencies due to undetected and unwanted correlations in the test data used for the evaluation and the safety argument, methods of bias detection on the fleet of deployed automated vehicles can be used. This addresses Challenges 2 and 3.

Runtime monitoring analysis may necessitate measures to ensure the ongoing safety of the deployed system. These measures may include:

- Developing a new safety mechanism or improving an existing safety mechanism
- Iterating the nominal function
- Updating the ODD
- Updating the dataset attributes
- Updating the output of the runtime monitoring mechanisms

The actions will result in a configuration and change control process trigger used to identify the need of new software versions (including retraining the DNN) and/or to revalidate the safety of the deployed system. The following artifacts are expected from this phase of development:

- KPI monitoring reports
- Corner case monitoring
- Distributional shift monitoring (ODD)

6.6 DNN Safety Artifacts

Artifacts from the above sections play a central role when building the safety argument for the safety case of the automated driving product. When developing a DNN, the question of which other artifacts are required in addition to the ones that are generated during traditional model-based development should be considered. To do so, one could consider the complete development pipeline of the DNN as described earlier in the publication. Along that pipeline, one could identify the following additional artifacts:

Example Safety Artifacts for DNN Development Steps			
Define	Specify	Develop & Evaluate	Deploy & Monitor
Dataset specification (specification of the global dataset attributes)	Refined dataset specification	Refined labeled dataset	KPI monitoring
Labeling specification (specification of the classes, boundaries, labeling guidelines)	Refined labeling specification	Chosen hyperparameters (architectural decision points, random seeds ...)	Corner case monitoring
DNN requirements specification (specification of the ODD, functional objective requirements, technical safety requirements, etc.)	Labeling quality report	Training baseline (model parameters, hyperparameters, data points for each training step)	Distributional shift monitoring (ODD)
KPI specification (measurables such as dataset coverage, algorithm robustness, dataset quality, etc.)	Labeled dataset (representative global dataset, including the data splits for training, validation and testing datasets)	Intermediate validation and testing report	
	KPI report dataset (measurables such as dataset coverage, algorithm robustness, dataset quality, etc.)	Code and evaluation report for DNN observers	
	Scripting tools (dataset creation, labeling, measurement of KPIs, etc.)	Chosen optimization and compression methods, parameters and baselines	
	Architecture specification (specification defining the chosen DNN design architecture to solve the objective defined for the system)		
	Code and objective of uncertainty and observers		
	Report on additional mechanisms to reach safety requirements		

Table 12: Example Safety Artifacts for DNN Development Steps

Chapter

07

GLOSSARY

7 Glossary

(VULNERABLE) ROAD USER	<p>A (vulnerable) road user is anyone who uses a road (including sidewalk and other adjacent spaces).</p> <p>Alternatively, (vulnerable) road users are defined as non-motorized (vulnerable) road users, such as pedestrians and cyclists as well as motorcyclists and persons with disabilities or reduced mobility and orientation.</p>
ACCEPTABLE RISK	<p>This refers to the remaining risk of a developed system that is argued to be acceptable by the developing company and also acceptable with respect to legal and social acceptance criteria.</p>
ACCIDENT	<p>An accident is an undesirable, unplanned event that leads to an unrecoverable loss of service due to unfavorable external conditions, typically involving material damage, financial loss and (lethally) injured humans.</p>
ARTIFICIAL MARKERS	<p>An artificial marker is an object or painting introduced into a scene with the purpose of marking positions in a three-dimensional space. A marker typically an easily recognizable shape such as a high contrast disk, square or another simple geometric object. In addition, a marker could also carry coded information that can be extracted and decoded by the automated driving system to assist in parking lot localization and other related features.</p>
AUTOMATED DRIVING SYSTEM (ADS)	<p>An automated driving system comprises a set of elements that offer a specific conditional or higher automated driving use case in or for a specific ODD.</p>
AUTOMATED VEHICLE (AV)	<p>Automated vehicles are vehicles equipped with at least one conditional (SAE L3) or higher (SAE L4/L5) automated driving system that enables them to provide an automated dynamic driving task.</p>
AVAILABILITY	<p>Availability is a system state with the ability to readily provide correct service.</p>
CAPABILITY	<p>A capability is the ability of a product to deliver a function, feature or service.</p>
CRASH	<p>A crash is an undesirable, unplanned event that leads to an unrecoverable loss of service caused by scientifically explainable unfavorable external conditions (e.g. human error), typically involving material damage, financial loss and severe or fatally injured humans.</p>
DEGRADATION	<p>Degradation is the reduced performance of the system or function, but which still provides safe operations/service in the presence of hazardous events.</p>
DEPENDABILITY	<p>Dependability is the ability to provide Reliability, Availability, Maintainability, Safety and Security (RAMSS).</p>

DETERMINISM	Determinism refers to the concept that an output is directly defined by its defined input. In this context, noise is defined as an undefined input that accompanies a defined input.
DRIVER-IN-THE-LOOP (DIL)	Target software is executed on prototypical or target hardware in the target vehicle or a mockup, and the environment is modified with virtual stimuli, whereas the driver's reaction influences the vehicle's behavior. E.g.: driving simulator or ViL (augmented reality for safety-related maneuvers).
DYNAMIC DRIVING TASK (DDT): [SAE J3016]	Dynamic driving tasks comprise all the real-time operational and tactical functions required to operate a vehicle in on-road traffic, excluding the strategic functions such as trip scheduling and selecting destinations and waypoints, and including without limitation: <ul style="list-style-type: none"> ○ Lateral vehicle motion control via steering (operational) ○ Longitudinal vehicle motion control via acceleration and deceleration (operational) ○ Monitoring the driving environment via object and event detection, recognition, classification, and response preparation (operational and tactical) ○ Object and event response execution (operational and tactical) ○ Maneuver planning (tactical) ○ Enhancing conspicuity via lighting, signaling and gesturing, etc. (tactical)
ELEMENT	Elements result from a first-level decomposition of capabilities to a logical system architecture. One or more elements realize one or more capabilities.
FAIL-DEGRADED	This means that the system is still able to operate safely when degraded.
FAIL-OPERATIONAL	This refers to full & safe operations/service in the presence of hazardous events. The loss of safety-related functions or system components shall not lead to a hazard.
FAIL-SAFE	This means that the system still operates in a safe state in the event of a failure.
FAILURE [ISO 26262]	A failure is the termination of an intended behavior of an element or an item due to a fault manifestation.
FAULT [ISO 26262]	A fault is an abnormal condition that can cause an element or an item to fail.
FAULT TOLERANCE	This refers to the ability to deliver a specified functionality in the presence of one or more specified faults.
FIELD OF VIEW (FOV)	Field of view describes the angle through which a sensor or device can pick up electromagnetic radiation. [HTTPS://WHATIS.TECHTARGET.COM/DEFINITION/FIELD-OF-VIEW-FOV]

FIELD OPERATIONAL TESTING (FOT)	<p>Field operational testing refers to use of large-scale testing programs aimed at generating a comprehensive assessment of the efficiency, quality, robustness and acceptance of transport solutions.</p> <p>[HTTP://WIKI.FOT-NET.EU/INDEX.PHP/WHAT_ARE_FIELD_OPERATIONAL_TESTS%3F]</p>
HD MAP	<p>High Definition (HD) maps are very detailed maps with high level precision mostly used in the context of automated driving systems to give the vehicle precise information about the road environment to maneuver effectively and safely.</p>
HARDWARE-IN-THE-CLOSED-LOOP (HIL)	<p>Target software is executed on target hardware, whereas the hardware outputs influence the hardware inputs.</p> <p>E.g. AUTOSAR Stack on Radar with no frontend</p>
HARDWARE REPROCESSING (OPEN LOOP)	<p>Target software is executed on target hardware, whereas the hardware outputs do not influence the hardware inputs</p> <p>E.g. monitor hardware testbench</p>
HUMAN-MACHINE INTERACTION	<p>Human-machine interaction focuses on the interdisciplinary interaction between a human and computer and considers the human-machine interface (HMI). The aim is to develop an ideal user interface that satisfies the requirements regarding the mental, cognitive and manual abilities of the user.</p> <p>[HTTPS://WWW.ITWISSEN.INFO/HMI-HUMAN-MACHINE-INTERACTION-MENSCH-MASCHINE-INTERAKTION.HTML]</p>
INCIDENT	<p>An incident is an undesirable, unplanned event that leads to a recoverable loss of service due to favorable external conditions, typically sparing any material damage, financial loss and (lethally) injured humans.</p>
ITEM DEFINITION (REFERENCE: ISO26262, P.16)	<p>System or combination of systems, to which ISO 26262 is applied, that implements a function or part of a function at the vehicle level.</p>
MINIMAL RISK CONDITION [SAE J3016]	<p>A condition to which a user or an ADS may bring a vehicle after performing the Minimal Risk Maneuver in order to reduce the risk of a crash when a given trip cannot or should not be completed.</p>
MINIMAL RISK MANEUVER [SAE J3016]	<p>Minimal risk maneuver refers to a procedure aimed at minimizing risks in traffic and which is automatically performed by the system, e.g. when the driver does not respond to a takeover request.</p>
MODE AWARENESS	<p>Mode awareness refers to the driver's capability to identify the current automation mode and their driving responsibility.</p>

NATURALISTIC DRIVING STUDIES (NDS)	<p>Naturalistic driving study concerns studies carried out using unobtrusive observation during driving in natural settings. A new approach, the driver becomes unaware of observation as data is collected as discreetly as possible. This data is then used to examine the relationship between the driver, vehicle and/or environment.</p> <p>[HTTP://WIKI.FOT-NET.EU/INDEX.PHP/WHAT_IS_THE_DIFFERENCE_BETWEEN_AN_FOT/_/PILOT/_/NATURALISTIC_DRIVING_STUDY_(NDS)%3F]</p>
NOMINAL PERFORMANCE	<p>Nominal performance of the system is defined as a system free from fault and one that meets or exceeds its defined performance metrics.</p>
OBJECT UNDER TEST (OUT)	<p>Similar usage as ISO 16750 for Device Under Test: The item or object which is to be tested as planned and specified.</p>
OPEN ROAD (OR)	<p>Target software is executed on target hardware in the target vehicle with a human driver, whereas the driving environment is real and can be only partially controlled. E.g. field operational test or naturalistic driving studies, testing in the development vehicles.</p>
OPERATIONAL DESIGN DOMAIN (ODD) [SAE J3016]	<p>The ODD refers to the operating conditions under which a given automated driving system or feature thereof is specifically designed to function. “These limitations reflect the technological capability of the automated driving system.”</p>
POSITIVE RISK BALANCE	<p>In the sense of: Positive Risk Balance is the result of a risk benefit evaluation with a lower remaining risk of traffic participation due to automated vehicles. This includes the fact that automated vehicles causes less crashes on average compared to the average human driver.</p>
PROVING GROUND (PG)	<p>Target software is executed on target hardware in the target vehicle, whereas the driving environment is real and largely controlled. The driver can be real or a robot. E.g. EBA tests on soft crash target.</p>
RELIABILITY	<p>This refers to the ability of a system to continuously provide correct service.</p>
REPROCESSING	<p>Reprocessing is the generic activity which is done with SoL and HoL. Reprocessing is a replay of time stamped recorded data with a sufficient time accuracy to provide input for the OuT.</p>

SAE LEVELS OF DRIVING AUTOMATION
[SAE J3016]

SAE J3016™ Levels of Driving Automation						
	SAE Level 0	SAE Level 1	SAE Level 2	SAE Level 3	SAE Level 4	SAE Level 5
What does the human in the driver's seat have to do?	You are driving whenever these driver support features are engaged – even if your feet are off the pedals, and you are not steering			You are not driving when these automated driving features are engaged – even if you are seated in “the driver's seat”		
	You must constantly supervise these support features; you must steer, brake or accelerate as needed to uphold safety			When the feature requests, you must drive	These automated driving features will not require you to take over driving	
	These are driver support features			These are automated driving features		
What do these features do?	These features are limited to providing warnings and momentary assistance	These features provide steering OR brake/acceleration support to the driver	These features provide steering AND brake/acceleration support to the driver	These features can drive the vehicle under limited conditions and will not operate unless all required conditions are met		This feature can drive the vehicle under all conditions
Example features	<ul style="list-style-type: none"> ▪ Automatic emergency braking ▪ Blind spot warning ▪ Lane departure warning 	<ul style="list-style-type: none"> ▪ Lane centering OR ▪ Adaptive cruise control 	<ul style="list-style-type: none"> ▪ Lane centering AND ▪ Adaptive cruise control at the same time 	<ul style="list-style-type: none"> ▪ Traffic jam chauffeur 	<ul style="list-style-type: none"> ▪ Local driverless taxi ▪ Pedals/steering wheel may or may not be installed 	<ul style="list-style-type: none"> ▪ Same as Level 4, but feature can drive everywhere in all conditions

SAFE STATE

Safe state is an operating mode without an unreasonable level of risk.

SAFE(TY)

This refers to the absence of unreasonable risk due to hazards.

SAFETY OF THE INTENDED FUNCTIONALITY (SOTIF)

“The absence of unreasonable risk due to hazards resulting from functional insufficiencies of the intended functionality or by reasonably foreseeable misuse by persons is referred to as the Safety Of The Intended Functionality (SOTIF).”

[[HTTPS://WWW.ISO.ORG/STANDARD/70939.HTML](https://www.iso.org/standard/70939.html)]

SCENARIO

A scenario is a temporal sequence of scenes and covers a certain time span.

SCENE

A scene describes a snapshot of an environment that describes the scenery, dynamic elements, and the self-representation of all actors and observers as well as their connection. Only a simulated scene can be all-embracing (i.e. objective, otherwise known as ground truth), whereas a real-world scene is incomplete, afflicted with faults and uncertainties, and observed from a subjective perspective.

SCENERY

The scenery includes all spatial stationary elements: The lane network (lanes, lane markings, etc.), stationary elements (obstacles, curbs, traffic signs, traffic lights, etc.), vertical elevation, and environmental conditions.

SECURITY

Security is the protection against intentional subversion or forced failure.

SIMULATION	The approximated imitation of selected behavioral characteristics of one physical or abstract system by a static or dynamic model [according to ISO 2382/1]. The simulation represents the behavior over time in which the system or parts of it are replaced by the model. It includes SiL, SoL, HiL, HoL and DiL.																																			
SOFTWARE REPROCESSING (OPEN LOOP)	Target software is executed on prototypical hardware, whereas the software decisions have no influence on the stimulus. E.g. replay or synthetic data to simulate a CEM																																			
SOFTWARE-IN-THE-CLOSED-LOOP (ABBREV.: SIL)	Partial target software is executed on prototypical hardware, whereas the software decisions influence the virtually generated stimulus. E.g. MATLAB Simulink model, AUTOSAR Stack, C++ DLL																																			
SYSTEM LIMITS	The defined limits of the operation as stated in the ODD for the specific system of interests.																																			
TAKEOVER	Transfer of responsibility for the driving task from the automated vehicle to the operator.																																			
UNREASONABLE RISK	This refers to a risk that is judged to be unacceptable in a certain context according to valid societal moral concepts																																			
USE CASE	<p>This is the specification of a generalized field of application, possibly entailing the following information about the system:</p> <ul style="list-style-type: none"> ○ one or several scenarios; ○ the functional range; ○ the desired behavior; and ○ the system boundaries <p>Note: The use case description typically does not include a detailed list of all relevant scenarios for this use case. Instead a more abstract description of these scenarios is used.</p>																																			
USER [SAE J3016]	<table border="1" data-bbox="552 1480 1374 1809"> <thead> <tr> <th colspan="7">User Roles in the Automated Driving System</th> </tr> <tr> <th rowspan="2"></th> <th rowspan="2">No Driving Automation 0</th> <th colspan="5">Engaged Level of Driving Automation</th> </tr> <tr> <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th> </tr> </thead> <tbody> <tr> <td>In-Vehicle User</td> <td>Driver</td> <td colspan="3"></td> <td>DDT fall-back-ready user</td> <td colspan="2">Passenger</td> </tr> <tr> <td>Remote User</td> <td>Remote driver</td> <td colspan="3"></td> <td>DDT fall-back-ready user</td> <td colspan="2">Driverless operation dispatcher</td> </tr> </tbody> </table> <p>A driver is a human being who is using a vehicle. This human being takes over different tasks, depending on the level of automation. A user is a general term that refers to the human role in driving automation (SAE): A passenger is a user in a vehicle, who has no role in operating the vehicle.</p>	User Roles in the Automated Driving System								No Driving Automation 0	Engaged Level of Driving Automation					1	2	3	4	5	In-Vehicle User	Driver				DDT fall-back-ready user	Passenger		Remote User	Remote driver				DDT fall-back-ready user	Driverless operation dispatcher	
User Roles in the Automated Driving System																																				
	No Driving Automation 0	Engaged Level of Driving Automation																																		
		1	2	3	4	5																														
In-Vehicle User	Driver				DDT fall-back-ready user	Passenger																														
Remote User	Remote driver				DDT fall-back-ready user	Driverless operation dispatcher																														

V2X	Vehicle-to-Everything (V2X) is an emerging technology that augments an automated vehicle to receive additional information from infrastructure or other vehicles or vice versa send information. V2X can provide a growing number of helpful information such as parking space availability, upcoming road hazards and map updates, or support tele-operation of the automated vehicle in relevant scenarios. A direct communication to human (vulnerable) road users is handled in human-machine interaction.
VALIDATION [ISO 15288]	“Confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled” [ISO/IEC15288]. Takes place during validation testing to determine if an outcome is best for the end customer. Typically done at a later development stage with much slower feedback, as validation is normally performed via statistical methods with high number of tests.
VEHICLE OPERATOR	The person who operates the vehicle either in the vehicle itself behind the steering wheel (L3 or L4) or via teleoperation (L4).
VERIFICATION [ISO 15288]	“Confirmation, through the provision of objective evidence, that specified requirements have been fulfilled” [ISO 15288]. Typically used to obtain fast feedback during development.

Chapter

08

REFERENCES

8 References

- ABI & THATCHAM RESEARCH.** (2017). *Regulating Automated Driving – The UK Insurer View*. UK.
- ARNOLD, E., AL-JARRAH, O. Y., DIANATI, M., FALLAH, S., OXTOBY, D., & MOUZAKITIS, A.** (2019). A Survey on 3D Object Detection Methods for Autonomous Driving Applications. *IEEE Transactions on Intelligent Transportation Systems*, 1-14. doi:10.1109/TITS.2019.2892405
- BAINBRIDGE, L.** (1983). Ironies of Automation. In *Analysis, Design and Evaluation of Man–Machine Systems. Proceedings of the IFAC/IFIP/IFORS/IEA Conference, Baden-Baden, Federal Republic of Germany, 27–29 September 1982* (pp. 129-135). UK: Pergamon.
- BANSAL, M., KRIZHEVSKY, A., & OGALE, A.** (2018). *ChauffeurNet: Learning to Drive by Imitating the Best and Synthesizing the Worst*. Waymo Research. Retrieved from <https://arxiv.org/pdf/1812.03079.pdf>
- BOJARSKI, M., DEL TESTA, D., DWORAKOWSKI, D., FIRNER, B., FLEPP, B., GOYAL, P., & ZIEBA, K.** (2016). End to End Learning for Self-Driving Cars. *arXiv:1604.07316*. Retrieved from <https://arxiv.org/abs/1604.07316>
- CALIFORNIA VEHICLE CODE 22350 VC.** (1959). *Division 11: Rules of the Road. Chapter 7: Speed Laws*. California. Retrieved from https://leginfo.legislature.ca.gov/faces/codes_displaySection.xhtml?lawCode=VEH§ionNum=22350
- CENSI, A., SLUTSKY, K., WONGPIROMSARN, T., YERSHOV, D. S., PENDLETON, S., FU, J., & FRAZZOLI, E.** (2019). Liability, Ethics, and Culture-Aware Behavior Specification using Rulebooks. *CoRR, abs/1902.09355*. Retrieved from <http://arxiv.org/abs/1902.09355>
- ECE/TRANS/WP.1/165.** (2018). *Report of the Global Forum for Road Traffic Safety on its Seventy-Seventh Session – Annex 1*. Geneva. Retrieved from <http://www.unece.org/fileadmin/DAM/trans/doc/2018/wp1/ECE-TRANS-WP1-165e.pdf>
- ECLIPSE FOUNDATION.** (2019). *OpenPASS Working Group*. Retrieved from www.openpass.eclipse.org: <https://openpass.eclipse.org/>
- ESSER, P., SUTTER, E., & OMMER, B.** (2018). A Variational U-Net for Conditional Appearance and Shape Generation. *arXiv:1804.04694*. Retrieved from <https://arxiv.org/abs/1804.04694>
- EUCAR.** (2018). *Self-Driving Vehicles: European Automotive R&D Leading the Global Race*. Retrieved from <https://www.eucar.be/self-driving-vehicles-european-automotive-rd-leading-the-global-race/>
- FAHRENKROG, F.** (2016). *Wirksamkeitsanalyse von Fahrerassistenzsystemen in Bezug auf die Verkehrssicherheit*. RWTH Aachen University. Aachen: fka GmbH.

- FEDERAL MINISTER OF TRANSPORT AND DIGITAL INFRASTRUCTURE (BMVI).** (June, 2017). *Ethics Commission – Automated and Connected Driving*. Berlin. Retrieved from https://www.bmvi.de/SharedDocs/EN/publications/report-ethics-commission-automated-and-connected-driving.pdf?__blob=publicationFile
- FELDHÜTTER, A., SEGLER, C., & BENGLER, K.** (2018). Does Shifting Between Conditionally and Partially Automated Driving Lead to a Loss of Mode Awareness? In N. Stanton (Ed.), *Advances in Human Aspects of Transportation. AHFE 2017. Advances in Intelligent Systems and Computing* (597, pp. 730-741). Springer, Cham. doi:https://doi.org/10.1007/978-3-319-60441-1_70
- FORM, T.** (2018). PEGASUS Method for Assessment of Highly Automated Driving Function. *SIP-Adus Workshop 2018, 13-15 November, 2018*. Tokyo International Exchange Center, Tokyo, Japan. Retrieved from http://en.sip-adus.go.jp/evt/workshop2018/file/PEGASUS_SIP-adus_Thomas_Form.pdf
- FRAADE-BLANAR, L., BLUMENTHAL, M., ANDERSON, J., & KALRA, N.** (2018). *Measuring Automated Vehicle Safety – Forging a Framework*. Santa Monica, CA: RAND Corporation. Retrieved from https://www.rand.org/pubs/research_reports/RR2662.html
- GAL, Y.** (2016). *Uncertainty in Deep Learning*. Dissertation, University of Cambridge, UK. Retrieved from <http://mlg.eng.cam.ac.uk/yarin/thesis/thesis.pdf>
- GESAMTVERBAND DER DEUTSCHEN VERSICHERUNGSWIRTSCHAFT E.V.** (2018). Technische Aspekte des automatisierten Fahrens und Verkehrssicherheit. (84). Germany. Retrieved from <https://udv.de/download/file/11115>
- GOLD, C. G.** (2016). *Modeling of Take-Over Performance in Highly Automated Vehicle Guidance*. Dissertation, Technische Universität München. Retrieved from <https://mediatum.ub.tum.de/doc/1296132/document.pdf>
- GOLD, C., DAMBÖCK, D., LORENZ, L., & BENGLER, K.** (2013). “Take Over!” How Long Does It Take To Get The Driver Back Into The Loop? *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 57(1), 1938-1942. doi:<https://doi.org/10.1177/1541931213571433>
- HANKE, T., HIRSENKORN, N., VAN-DRIESTEN, C., GARCIA-RAMOS, P., SCHIEMENTZ, M., SCHNEIDER, S., & BIEBL, E.** (2017). *Open Simulation Interface: A Generic Interface for the Environment Perception of Automated Driving Functions in Virtual Scenarios*. Retrieved from [www.github.com: http://www.hot.ei.tum.de/forschung/automotive-veroeffentlichungen](http://www.hot.ei.tum.de/forschung/automotive-veroeffentlichungen)
- HERE.** (2019). *HERE Hazard Warnings – Keeping Vehicles, and Their Drivers, Perfectly Alert*. Retrieved June 18th, 2019, from [www.here.com: https://www.here.com/products/automotive/hazard-warnings](https://www.here.com/products/automotive/hazard-warnings)
- HINTON, J., & SEJNOWSKI, T.** (1999). *Unsupervised Learning: Foundations of Neural Computation*. Cambridge, MA: MIT Press.

- IEC 61508.** (2010). *2nd Ed. Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems (all parts)*.
- INFORMAL WORKING GROUP ON INTELLIGENT TRANSPORT SYSTEMS/AUTOMATED DRIVING (IWG).** (2017). *Proposal for the Definitions of Automated Driving under WP.29 and the General Principles for Developing a UN Regulation on Automated Vehicles*. Geneva: UNECE.
- ISO 26262.** (2018). *Road Vehicles – Functional Safety*.
- ISO/IEC 15288.** (2015). *Systems and Software Engineering – System Life Cycle Processes*.
- ISO/PAS 21448.** (2018). *Road Vehicles – Safety of The Intended Functionality*.
- ISO/SAE CD 21434.** (Under Development). *Road Vehicles – Cybersecurity Engineering*.
- KALRA, N., & PADDOCK, S.** (2016). *Driving to Safety – How Many Miles of Driving Would It Take to Demonstrate Autonomous Vehicle Reliability?* Santa Monica, CA: RAND Corporation. Retrieved from https://www.rand.org/pubs/research_reports/RR1478.html
- KERSCHBAUM, P., LORENZ, L., & BENGLER, K.** (2014). *Highly Automated Driving with a Decoupled Steering Wheel. Proceedings of the Human Factors and Ergonomics Society 58th Annual Meeting*, 58(1), 1686-1690. Los Angeles, CA.
- KNAPP, A., NEUMANN, M., BROCKMANN, M., WALZ, R., & WINKLE, T.** (2009). *PREVENT RESPONSE III: Code of Practice for the Design and Evaluation of ADAS*. Retrieved from https://www.acea.be/uploads/publications/20090831_Code_of_Practice_ADAS.pdf
- KOOPMAN, P., & WAGNER, M.** (2018). *Toward a Framework for Highly Automated Vehicle Safety Validation*. SAE Technical Paper 2018-01-1071. doi:<https://doi.org/10.4271/2018-01-1071>
- LORENZ, L., KERSCHBAUM, P., & SCHUHMANN, J.** (2014). *Designing Take Over Scenarios for Automated Driving: How Does Augmented Reality Support the Driver to get Back into the Loop?* *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 58(1), 1681-1685. doi:<https://doi.org/10.1177/1541931214581351>
- MENZEL, T., BAGSCHIK, G., & MAURER, M.** (2018). *Scenarios for Development, Test and Validation of Automated Vehicles*. In *2018 IEEE Intelligent Vehicles Symposium IV 2018, Changshu, Suzhou, China, June 26-30, 2018*. Retrieved from <https://arxiv.org/pdf/1801.08598.pdf>
- MICROSOFT.** (2019). *Microsoft Security Development Lifecycle (SDL)*. Retrieved from www.microsoft.com: <https://www.microsoft.com/en-us/securityengineering/sdl/>

- NATIONAL HIGHWAY TRANSPORTATION SAFETY ADMINISTRATION (NHTSA).** (2017). *Automated Driving Systems 2.0 – A Vision for Safety*. Retrieved from https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf
- NATIONAL TRANSPORTATION SAFETY BOARD (NTSB).** (2017). Collision Between a Car Operating With Automated Vehicle Control Systems and a Tractor-Semitrailer Truck Near Williston, Florida, May 7, 2016. *NTSB/HAR-17/02*. Retrieved from <https://www.nts.gov/investigations/AccidentReports/Reports/HAR1702.pdf>
- NHTSA'S NATIONAL CENTER FOR STATISTICS AND ANALYSIS (NCSA).** (February, 2015). *TRAFFIC SAFETY FACTS. Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey*. USA.
- NPC.** (2017). *Surveying and Mapping Law of the People's Republic of China*. Retrieved from http://www.npc.gov.cn/englishnpc/Law/2007-12/12/content_1383865.htm
- OHNO, T.** (1988). *Toyota Production System: Beyond Large-Scale Production*. Portland, Oregon: Productivity Press.
- OpenDRIVE.** (2018). *OpenDRIVE Standard*. Retrieved from <http://www.opendrive.org/project.html>
- OpenSCENARIO.** (2017). *OpenSCENARIO Specification Rev. 0.9.1*. Retrieved from <http://www.openscenario.org/download.html>
- PEGASUS.** (2019). *PEGASUS Method – An Overview*. Retrieved from <https://www.pegasusprojekt.de/files/tmp/PEGASUS-Abschlussveranstaltung/PEGASUS-Gesamtmethode.pdf>
- PETERMANN, I., & SCHLAG, B.** (2010). Auswirkungen der Synthese von Assistenz und Automation auf das Fahrer-Fahrzeug System. *11. Symposium AAET 2010 – Automatisierungs-, Assistenzsysteme und eingebettete Systeme für Transportmittel*, 257-266. Germany.
- SAE J3016.** (2018). *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. Document J3016_201806*. SAE International. doi:https://doi.org/10.4271/J3016_201806.
- SALAY, R., & CZARNECKI, K.** (2018). *Using Machine Learning Safely in Automotive Software: An Assessment and Adaption of Software Process Requirements in ISO 26262*. Waterloo Intelligent Systems Engineering (WISE) Lab. University of Waterloo, Canada. Retrieved from <https://arxiv.org/ftp/arxiv/papers/1808/1808.01614.pdf>
- SCHMIDHUBER, J.** (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61, 85–117. doi:<https://doi.org/10.1016/j.neunet.2014.09.003>

- SHALEV-SCHWARTZ, S., SHAMMAH, S., & SHASHUA, A.** (2018). On a Formal Model of Safe and Scalable Self-driving Cars. *CoRR*, *abs/1708.06374*. Retrieved from <http://arxiv.org/abs/1708.06374>
- STATISTISCHES BUNDESAMT (DESTATIS).** (2018). *Fachserie 8, Reihe 7, Verkehr, Verkehrsunfälle 2017*. Germany.
- STRASSENVERKEHRSGESETZ (STVG).** (2018). §1a Kraftfahrzeuge mit hoch- oder vollautomatisierter Fahrfunktion. *BGBI. I* [in English: *German Road Traffic Act, §1a Motor Vehicles with Highly or Fully Automated Driving Functions (Federal Law Gazette I)*], 310,919. Germany.
- SUTTON, R., & BARTO, A.** (1998). *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press.
- TAGUE, N.** (2005). *The Quality Toolbox* (2nd ed.). Wisconsin: Quality Press.
- U.S. DEPARTMENT OF TRANSPORTATION (U.S. DOT).** (2016). *Federal Automated Vehicles Policy – September 2016*. Retrieved from <https://www.transportation.gov/AV/federal-automated-vehicles-policy-september-2016>
- U.S. DEPARTMENT OF TRANSPORTATION (U.S. DOT).** (2018). *Preparing for the Future of Transportation; Automated Vehicles 3.0*. USA. Retrieved from <https://www.transportation.gov/sites/dot.gov/files/docs/policy-initiatives/automated-vehicles/320711/preparing-future-transportation-automated-vehicle-30.pdf>
- UNITED NATIONS.** (1969). Vienna Convention on the Law of Treaties. 1155, 331. Retrieved from <https://www.refworld.org/docid/3ae6b3a10.html>
- VAN DEN BEUKEL, A. P., VAN DER VOORT, M. C., & EDGER, A. O.** (2016). Supporting the Changing Driver's Task: Exploration of Interface Designs for Supervision and Intervention in Automated Driving. *Transportation Research Part F: Traffic Psychology and Behaviour*, 43, 279-301. doi:<https://doi.org/10.1016/j.trf.2016.09.009>
- WACHENFELD, W.** (2017). *How Stochastic can Help to Introduce Automated Driving*. Dissertation, Technische Universität Darmstadt. Retrieved from http://tuprints.ulb.tu-darmstadt.de/5949/7/Diss_Wf_2017_02_04_Ver%C3%B6ffentlichung.pdf
- WINNER, H., HAKULI, H., LOTZ, F., & SINGER, C. (EDS.).** (2016). *Handbook of Driver Assistance Systems* (1 ed.). Switzerland: Springer International Publishing. doi:10.1007/978-3-319-12352-3
- ZEEB, K., BUCHNER, A., & SCHRAUF, M.** (2016). Is Take-Over Time All That Matters? The Impact of Visual-Cognitive Load on Driver Take-Over Quality after Conditionally Automated Driving. *Accident Analysis & Prevention*, 92, 230-239. <https://doi.org/10.1016/j.aap.2016.04.002>