TECHNICAL REPORT

ISO/TR 4804

First edition
2020-12

Road vehicles — Safety and cybersecurity for automated driving systems — Design, verification and validation

Véhicules routiers — Sécurité et cybersécurité pour les systèmes de conduite automatisée — Conception, vérification et validation

Reference number
ISO/TR 4804:2020(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 22, *Road Vehicles*.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

Automated driving is one of the key modern technologies. In addition to offering broader access to mobility, it may also help to reduce the number of road traffic related accidents and crashes. When doing so, the safe operation of automated driving vehicles is one of the most important factors. Designed to supplement existing standards and publications on various aspects of safety, this document presents a more technical overview of the recommendations, guidance and methods to achieve a positive risk balance and to avoid unreasonable risk and cybersecurity related threats, emphasizing the importance of safety by design. This document closes the loop to provide a discussion with recommendations and methods on the verification and validation of automated driving systems.

Set forth are a proposed framework and guidelines focused on the safety and cybersecurity during the development, verification, validation, production and operation of automated driving systems for all stakeholders in the automotive and mobility world – from technology start-ups through to established OEMs and the tiered suppliers of key technologies.

Road vehicles — Safety and cybersecurity for automated driving systems — Design, verification and validation

1 Scope

This document describes steps for developing and validating automated driving systems based on basic safety principles derived from worldwide applicable publications. It considers safety- and cybersecurity-by-design, as well as verification and validation methods for automated driving systems focused on vehicles with level 3 and level 4 features according to SAE J3016:2018. In addition, it outlines cybersecurity considerations intersecting with objectives for safety of automated driving systems.

2 Normative references

There are no normative references in this document

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
automated driving system
ADS
set of *elements* (3.14) that offer a specific conditional or higher automated driving *use case* (3.63) in or for a specific *ODD* (3.37)

3.2
automated vehicle
AV
vehicle equipped with at least one conditional (SAE level 3) or higher (SAE level 4/level 5) *automated driving system* (3.1)

3.3
availability
capability (3.4) of a product to provide a stated function if demanded, under given conditions over its defined lifetime

Note 1 to entry: In the context of this document the product is the *automated driving system* (3.1).

Note 2 to entry: In the context of this document "availability" is defined solely referring to the automated driving system aspects and does not include human factor aspects.

[SOURCE: ISO 26262-1:2018, 3.7]

3.4
capability
ability of a product to deliver a function, feature or service

Note 1 to entry: In the context of this document the product is the *automated driving system* (3.1).

3.5
conventional driver
driver (3.11) who manually exercises in-vehicle braking, accelerating, steering and transmission gear selection input devices in order to operate the vehicle

[SOURCE: SAE J3016:2018, 3.29.1.1]

3.6
corner case
scenario (3.53) in which two or more parameter values are each within the *capabilities* (3.4) of the system, but together constitute a rare condition that challenges its capabilities

Note 1 to entry: In the context of this document the system is the *automated driving system* (3.1).

[SOURCE: ISO/PAS 21448:2019, Table 11]

3.7
crash
undesirable, unplanned event that leads to an unrecoverable loss due to unfavourable external conditions (e.g. human error), typically involving material damage, financial loss or human injuries and/or fatalities

3.8
cybersecurity
condition in which assets are sufficiently protected against threat *scenarios* (3.53) to electrical or electronic components of road vehicles and their functions

[SOURCE: ISO/SAE 21434]

3.9
degradation
state or transition to a state of the *item* (3.26) or *element* (3.14) with reduced functionality, performance, or both

Note 1 to entry: In the context of this document the item is the *automated driving system* (3.1).

[SOURCE: ISO 26262-1:2018, 3.28, modified — Note 1 to entry added.]

3.10
dependability
ability of a system to provide a service or function regarding the attributes of *reliability* (3.44), *availability* (3.3), maintainability, *safety* (3.51) and security (RAMSS)

Note 1 to entry: In the context of this document the system is the *automated driving system* (3.1).

3.11
driver
user (3.64) who performs in real-time part or all of the *DDT* (3.13) and/or DDT fallback for a particular vehicle

[SOURCE: SAE J3016:2018, 3.29.1, modified — The word "human" was removed from the term and the note was deleted.]

3.12
driver in the loop
DiL
execution of the target software on prototype or target hardware in the target vehicle or a mock-up, in which the environment is modified with virtual stimuli, and the driver's reaction influences the vehicle's behaviour

EXAMPLE Driving simulator or vehicle in the loop (ViL) (augmented reality for safety-related manoeuvres in real vehicles).

3.13
dynamic driving task
DDT
all of the real-time operational and tactical functions required to operate a vehicle in on-road traffic

Note 1 to entry: This excludes the strategic functions such as trip scheduling and selecting destinations and waypoints, and includes without limitation:

- 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);
- manoeuvre planning (tactical); and
- enhancing conspicuity via lighting, signalling or gesturing, etc. (tactical).

[SOURCE: SAE J3016:2018, 3.13, modified — Note 1 to entry was previously part of the definition, the notes, figure and additional information were removed.]

3.14
element
at least first-level decomposition of *capabilities* (3.4) to a logical system architecture

Note 1 to entry: One or more elements realize one or more capabilities.

3.15
equivalence class
class being identified based on the division of inputs and outputs, such that a representative test value can be selected for each class

Note 1 to entry: See ISO 26262-6:2018, Table 8.

3.16
fail-degraded
property of the *item* (3.26) to operate with reduced functionality in the presence of a *fault* (3.20)

Note 1 to entry: This property can be realized as fail-degraded *capability* (3.4) of fail-degraded mode.

Note 2 to entry: In the context of this document the item is the *automated driving system* (3.1).

Note 3 to entry: This means that the item is fault-tolerant for a subset of its intended functionality.

Note 4 to entry: The absence of *unreasonable risk* (3.62) can require the duration of the presence of the fault to be time limited and/or system maintenance in a limited time frame.

Note 5 to entry: The absence of unreasonable risk in the presence of the fault can require limitations of the item behaviour.

3.17
fail-operational
property of the *item* (3.26) to maintain its full intended functionality in the presence of a *fault* (3.20)

Note 1 to entry: In the context of this document the item is the *automated driving system* (see 3.1).

Note 2 to entry: This means that the item is fault-tolerant for its intended functionality.

Note 3 to entry: The absence of *unreasonable risk* (3.62) can require the duration of the presence of the fault to be time limited and/or system maintenance in a limited time frame.

3.18
fail-safe
property of an *automated driving system* (3.1) to achieve a *minimal risk condition* (3.29) and to achieve a *safe state* (3.50) in the event of a *failure* (3.19)

Note 1 to entry: A fail-safe condition is to be reached for example, by means of: demanding the vehicle control to driver/vehicle *operator* (3.39) and/or switching off the automated driving function.

3.19
failure
termination of an intended behaviour of an *element* (3.14) or the *automated driving systems* (3.1) due to a *fault* (3.20) manifestation

[SOURCE: ISO 26262-1:2018, 3.50, modified — The term "automated driving system" replaces "item" and Note 1 to entry is not included here.]

3.20
fault
abnormal condition that can cause an *element* (3.14) or the *automated driving system* (3.1) to fail

[SOURCE: ISO 26262-1:2018, 3.54, modified — The term "automated driving system" replaces "item" and Notes to entry are not included here.]

3.21
field operational testing
FOT
use of large-scale testing programs aimed at generating a comprehensive assessment of the efficiency, quality, robustness and acceptance of transport solutions

3.22
high definition map
HD map
maps with high level precision mostly used in the context of *automated driving system* (3.1) to give the vehicle precise information about the road environment

3.23
hardware in the closed loop
HiL
execution of target software on target hardware, whereby the hardware outputs influence the hardware inputs

Note 1 to entry: HiL executes the target software in real time.

EXAMPLE AUTOSAR stack on radar with no frontend.

3.24
hardware open loop
HoL
execution of target software on target hardware, whereby the hardware outputs do not influence the hardware inputs

EXAMPLE Monitor hardware testbench.

3.25
human-machine interaction
interdisciplinary interaction between a human and an *automated vehicle* (3.2), considering the human-machine interface (HMI) with the aim to develop a user interface that satisfies requirements regarding mental, cognitive and manual abilities of the *user* (3.64)

3.26
item
system or combination of systems, that implements a function or part of a function at the vehicle level

[SOURCE: ISO 26262-1:2018, 3.84, modified — The phrase "to which ISO 26262 is applied" and the Note 1 to entry were deleted.]

3.27
lagging measure
metrics that are assessed after deployment of an *automated driving system* (3.1) and provide confirmation that the *positive risk balance* (3.42) as well as the conformance with the safety-by-design techniques have been achieved

EXAMPLE Statistics for *crashes* (3.7) or other *safety* (3.51) events.

Note 1 to entry: See Reference [1].

3.28
leading measure
metrics that are derived from data that is assessed prior to deployment of an *automated driving system* (3.1) indicating that the automated driving system conforms with safety-by-design techniques to achieve a *positive risk balance* (3.42) and avoidance of *unreasonable risk* (3.62)

EXAMPLE A design *verification* (3.67) that the HMI guidelines were incorporated into the vehicle's design.

Note 1 to entry: See Reference [1].

3.29
minimal risk condition
MRC
condition to which a *user* (3.64) or an *automated driving system* (3.1) may bring a vehicle after performing the *minimal risk manoeuvre* (3.30) in order to reduce the risk of a *crash* (3.7) when a given trip cannot be completed

Note 1 to entry: The minimal risk condition integrates the meaning of avoidance of *unreasonable risk* (3.62), according to the ISO 26262:2018 series. They can be combined but they never exclude one each other.

[SOURCE: SAE J3016:2018, 3.17, modified — The term "minimal risk manoeuvre" replaces "DDT fallback", the notes and examples were deleted and the Note 1 to entry was added.]

3.30
minimal risk manoeuvre
MRM
automated driving system's (3.1) *capability* (3.4) of transitioning the vehicle between nominal and *minimal risk conditions* (3.29)

3.31
operating mode awareness
driver's (3.11) *capability* (3.4) to identify the current automation mode and his/her driving responsibility

3.32
naturalistic driving study
NDS
driving study where research subjects are recruited to drive on public roads (not in a simulator or on a test track), where there is no in-vehicle experimenter or confederate vehicles, and where driving conditions are not experimentally controlled or manipulated

Note 1 to entry: Subjects are not instructed to drive differently than they normally would and the data collection instrumentation is unobtrusive.

Note 2 to entry: Typically, these studies last a minimum of several weeks for each subject and can go much longer.

Note 3 to entry: An approach during which 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.

3.33
nominal performance
performance of the system free from *fault* (3.20) and that meets its defined performance criteria

3.34
non-vulnerable road user
protected *road users* (3.46) such as *users* (3.64) in other vehicles, trucks, construction and agricultural machines

3.35
object under test
OuT
item (3.26) or *element* (3.14) to be tested as planned and specified

Note 1 to entry: Similar usage as ISO 16750 for device under test.

3.36
open road testing
execution of target software on target hardware in the target vehicle with a *driver* (3.11), whereby the driving environment is real, road infrastructures are public and can be only partially controlled

EXAMPLE *Field operational testing* (3.21) or *naturalistic driving studies* (3.32), testing in the development vehicles.

3.37
operational design domain
ODD
operating conditions under which a given *automated driving system* (3.1) or feature thereof is specifically designed to function, including, but not limited to, environmental, geographical, and time-of-day restrictions, and/or the requisite presence or absence of certain traffic or roadway characteristics

Note 1 to entry: These limitations, as from constraints as specified in operating conditions, reflect the technological *capability* (3.4) of the automated driving system.

[SOURCE: SAE J3016:2018, 3.22, modified — Note 1 to entry is added, and the note and examples are not included here.]

3.38
ODD functional adaptation
operational design domain functional adaptation
property of a system to operate safely with reduced performance in the case of detected functional insufficiencies inside the *ODD* (3.37)

EXAMPLE Speed adaptation because of low fuel level, or dense fog, or sensor performance insufficiencies.

3.39
operator
designated person, appropriately trained and authorized, to operate the vehicle

3.40
other road user
vulnerable road users (3.68) and *non-vulnerable road users* (3.34) with no role in the ego *automated vehicle* (3.2)

3.41
passenger
user (3.64) in a vehicle who has no role in the operation of that vehicle

3.42
positive risk balance
benefit of sufficiently mitigating residual risk of traffic participation due to *automated vehicles* (3.2)

Note 1 to entry: This includes the expectation that automated vehicles cause less *crashes* (3.7) on average compared to those made by drivers.

Note 2 to entry: Positive risk balance is one of the concepts that can be considered when defining the acceptance criteria of ISO/PAS 21448:2019.

3.43
proving ground testing
execution of target software on target hardware in the target vehicle in a realistic but controlled and private driving environment

Note 1 to entry: The driver can be real or a robot.

EXAMPLE Emergency braking assistant tests on soft crash target.

3.44
reliability
ability of a system to continuously provide correct service

3.45
reprocessing
replay of time stamped, recorded data to provide input for the *object under test* (3.35)

Note 1 to entry: The time stamp needs a sufficient time accuracy.

3.46
road user
anyone who uses a road including sidewalk and other adjacent spaces

Note 1 to entry: Relationship between the human related terms are shown in Figure 1.

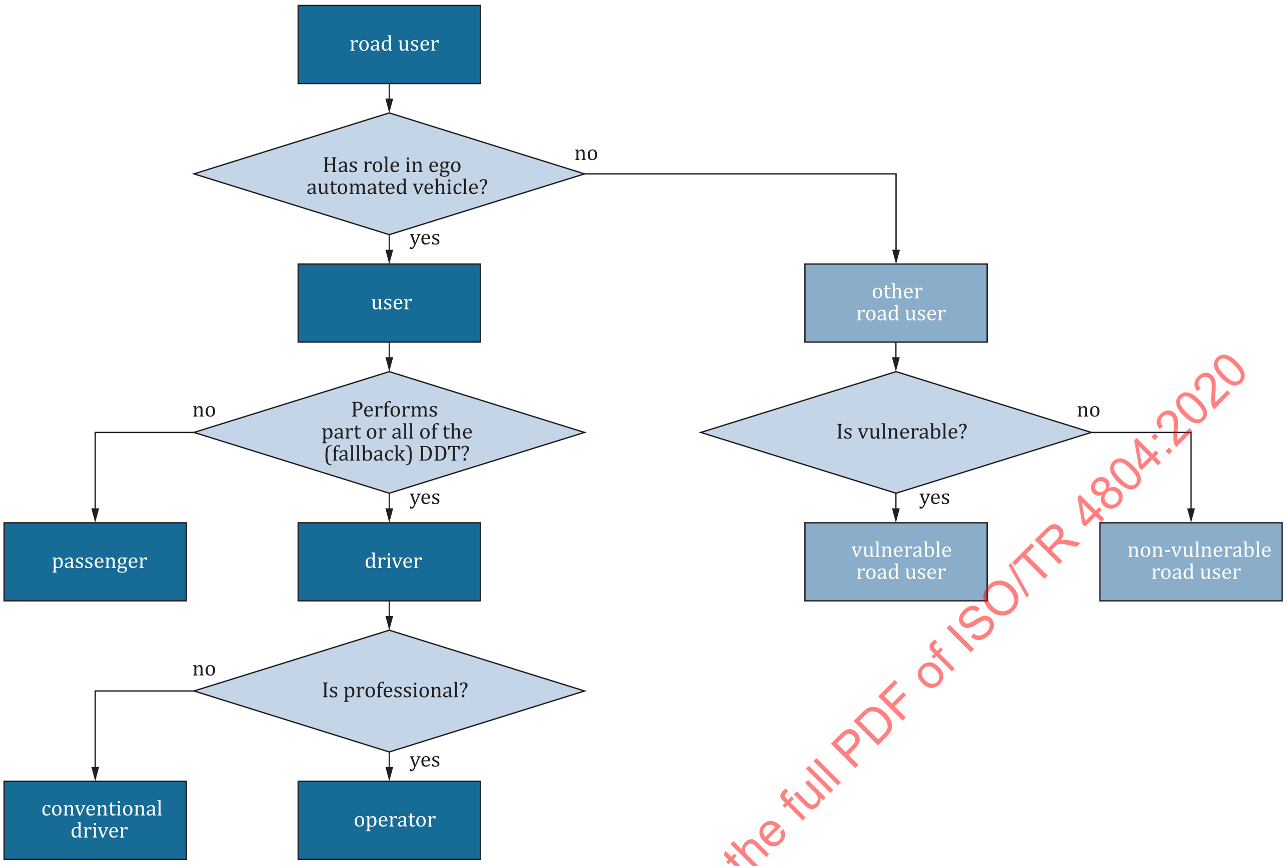

Figure 1 — Decision tree for terms related to automated driving systems

3.47
remote driver
driver (3.11) who is not seated in a position to manually exercise in-vehicle braking, accelerating, steering and transmission gear selection input devices (if any) but is able to operate the vehicle

[SOURCE: SAE J3016:2018, 3.29.1.2, modified — The notes and examples were removed.]

3.48
remote operator
operator (3.39) who is not seated in a position to manually exercise in-vehicle braking, accelerating, steering and transmission gear selection input devices (if any) but is able to operate the vehicle with or without direct vision

3.49
SAE levels of driving automation
levels of driving automation as defined in SAE J3016

Note 1 to entry: Six levels of driving automation, ranging from no driving automation (level 0) to full driving automation (level 5), in the context of motor vehicles and their operation on roadways. See Table 1.

Table 1 — Level of automation SAE level

Level	Level of automation
Driver performs part or all of the DDT	
0	No driving automation
1	Driver assistance
2	Partial driving automation

Table 1 *(continued)*

Level	Level of automation
ADS performs the entire DDT (while engaged)	
3	Conditional driving automation
4	High driving automation
5	Full driving automation

3.50
safe state
operating mode that is reasonably safe

Note 1 to entry: The safe state is the state in which both *fail-safe* (3.18) and *fail-degraded* (3.16) systems will provide a solution (technically provided by an alternative functionality) to avoid risk, in an acceptable criterion, to any *road user* (3.46).

3.51
safety
absence of *unreasonable risk* (3.62)

[SOURCE: ISO 26262-1:2018, 3.132]

3.52
safety of the intended functionality
SOTIF
absence of *unreasonable risk* (3.62) due to hazards resulting from functional insufficiencies of the intended functionality or by reasonably foreseeable misuse by *road users* (3.46)

[SOURCE: ISO/PAS 21448:2019, modified — The term "road users" replaces “persons”.]

3.53
scenario
description of the consecutive time series of activities integrating the subject vehicle, all its external environment and their interactions in the process of performing a certain driving task

3.54
scene
snapshot of the *scenario* (3.53) at a given point of time

Note 1 to entry: Only a simulated scene can be all-embracing (i.e. objective, otherwise known as ground truth), whereby a real-world scene is incomplete, afflicted with *faults* (3.20) and uncertainties, and observed from a subjective perspective.

3.55
scenery
part of the environment that remains unchanged

3.56
simulation
approximated imitation of selected behavioural characteristics of one physical or abstract system by a static or dynamic model

Note 1 to entry: The simulation represents the behaviour over time in which the system or parts of it are replaced by the model. It includes *SiL* (3.58), *SoL* (3.57), *HiL* (3.23), *HoL* (3.24) and *DiL* (3.12).

3.57
software reprocessing open loop
SoL
execution of target software on hardware, whereby the software decisions have no influence on the stimulus

3.58
software in the closed loop
SiL
execution of partial target software on hardware, whereby the software decisions influence the virtually generated stimulus

EXAMPLE MATLAB Simulink model, AUTOSAR Stack, C++ dynamic link library.

3.59
system limit
limit of the operation as stated in the *ODD* (3.37) for the specific system of interests

3.60
takeover
transfer of the driving task between the *automated driving system* (3.1) and the *driver* (3.11)

Note 1 to entry: The term handover is not used in this document. In this context the terms handover and takeover are regarded as synonyms.

3.61
tele operator
remote operator (3.48) without direct vision, but with tele-transmitted information (e.g. by cameras)

3.62
unreasonable risk
risk judged to be unacceptable in a certain context according to valid societal moral concepts

[SOURCE: ISO 26262-1:2018, 3.176]

3.63
use case
specification of a generalized field of application, possibly entailing information of the system on one or several *scenarios* (3.53), the functional range, the desired behaviour and the *system limits* (3.59)

Note 1 to entry: 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.

3.64
user
general term referencing the human role in driving automation

[SOURCE: SAE J3016:2018, 3.29, modified — The word "human" was removed from the term and the notes were deleted.]

3.65
vehicle-to-everything
V2X
technology that allows a vehicle to exchange additional information with infrastructure, other vehicles and *other road users* (3.40)

Note 1 to entry: 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* (3.2) in relevant *scenarios* (3.53).

3.66
validation
confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled

[SOURCE: ISO/IEC/IEEE 15288:2015, 4.1.53, modified — Note 1 to entry was deleted.]

3.67
verification
confirmation, through the provision of objective evidence, that specified requirements have been fulfilled

Note 1 to entry: Typically used to obtain fast feedback during development.

[SOURCE: ISO/IEC/IEEE 15288:2015, 4.1.54, modified — Note 1 to entry was replaced.]

3.68
vulnerable road user
VRU
non-protected *road user* (3.46) such as motorcyclists, cyclists, pedestrians and persons with disabilities or reduced mobility and orientation

4 General approach and overview

4.1 Introduction and motivation

Automated driving is one of the key modern technologies. In addition to offering broader access to mobility, it may also help to reduce the number of driving-related crashes. When doing so, the safe operation of automated vehicles is one of the most important factors. Designed to supplement existing publications and standards (see Annex D) on various aspects related to safety and cybersecurity of automated driving, this document presents a more technical overview of the specification for the development to avoid safety-related hazards, emphasizing the importance of safety by design. Furthermore, this document aims to provide a sound discussion of the verification and validation of such systems.

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

4.2 Overview of this document

The goal of this document is to provide an overview of and guidance about the general steps for developing, verifying and validating automated driving system safety. The starting point for this is the extraction of principles taken from different regulatory publications (various legal frameworks from around the world, ethics reports, etc.). These principles are the foundation of this document, forming the basis from which the safety by design methods and verification and validation 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 cybersecurity in conjunction with safety, whereby cybersecurity evaluation will necessitate the use of additional analysis and technical mechanisms which in turn impacts safety. Thus, safety and cybersecurity need work together, and 5.2.4 explores this concept in detail.

This document further aims to develop guidance to tackle the risks introduced by automated driving system. It reflects the state-of-the-art automated driving technology and capabilities at the time of publication. As such, it is not a complete work and its contents will be revisited and revised as advances are made in the areas of social acceptance, technology, legal and legislation.

Devising an explicit technical solution or minimum or maximum standard is not included in the scope of this document, as several possibilities exist regarding the definition of the automated driving system, its operational design domain and technical advancements, and unique technical design elements that vary among manufacturers and that affect system capabilities, etc. Due to its focus, this document does not address topics such as abuse, data privacy and advanced driver assistance systems. 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 explicitly considered in the scope of this document.

4.3 Structure and development examples used in this document

This document is structured as interconnected topics which build upon one another to achieve an overall safety vision.

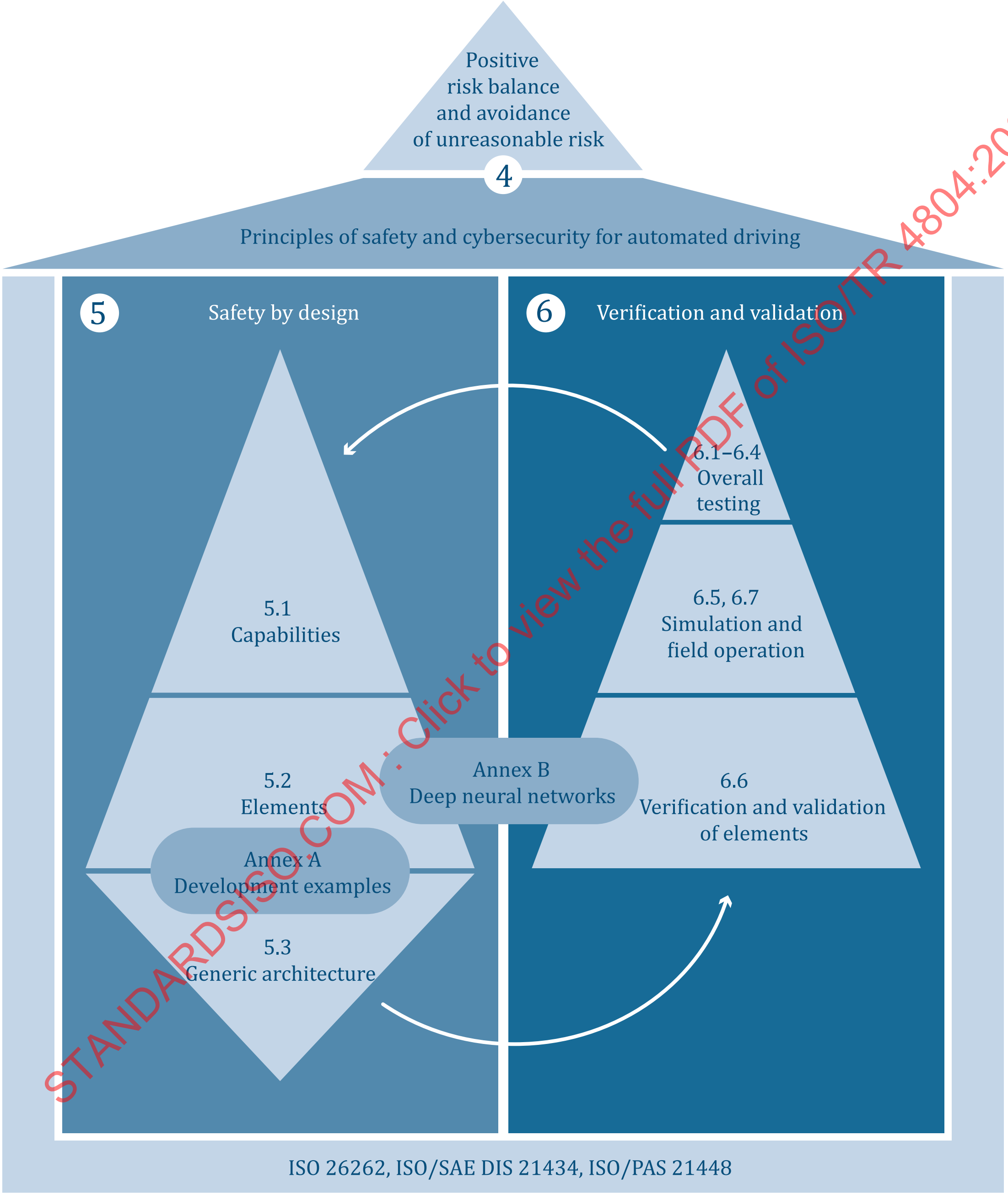
Positive risk balance and avoidance of unreasonable risk
4
Principles of safety and cybersecurity for automated driving
5
Safety by design
6
Verification and validation
5.1 Capabilities
5.2 Elements
Annex A Development examples
5.3 Generic architecture
Annex B Deep neural networks
6.1–6.4 Overall testing
6.5, 6.7 Simulation and field operation
6.6 Verification and validation of elements
ISO 26262, ISO/SAE DIS 21434, ISO/PAS 21448

Figure 2 — Structural overview

The structure of this document is summarized in Figure 2. Clause 3 describes the safety and cybersecurity principles recommended to achieve a positive risk balance and to avoid unreasonable risk. Clause 4 describes how measures taken during the system design support the argument for safety and cybersecurity and Clause 5 describes the contribution of evidence provided by the verification and validation activities. The first pillar introduces the three domains for automated driving: safety of the intended functionality (5.2.2, based on ISO/PAS 21448), functional safety (5.2.3, based on the

ISO 26262:2018 series) and automotive cybersecurity (5.2.4, based on ISO/SAE 21434[1]). Capabilities for automated driving are then derived from the principles of safety and cybersecurity for automated driving systems and the three dependability domains. The architecture of the development examples (described in Annex A) forms the last element of this pillar.

The second pillar begins by introducing the approach in 6.2 to 6.4 before discussing the quantity of testing (6.5) and simulation (6.6). 6.7 presents the verification and validation of the elements introduced in 5.3. The final block of the second pillar comprises the discussion relating to field operation in 6.8. Both pillars are linked together via the verification and validation approach outlined throughout Clause 6, which combines safety by design and testing with the main strategies applied in verification and validation to solve the challenges discussed throughout this document. Finally, Annex B discusses the use of deep neural networks (DNNs) to realize safety-relevant elements for automated driving.

Various methods are used to aid the reader of this document. This document also uses the following four development examples and their pictograms throughout for further clarity.

- L3 traffic jam chauffeur system (TJCS) as an option for vehicle customers: vigilant conventional driver with driver's license, driving only on structurally separated roads, typically less pedestrians or cyclists compared to other roads, 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 motorway chauffeur system (MCS) as an option for vehicle customers: vigilant conventional 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 chauffeur system (UCS) in fleet operation in urban areas: non-vigilant conventional 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 teleoperation if necessary.

- L4 automated valet parking system (AVPS) as an option for vehicle customers and in fleet operation: driverless movement within certified parking structures or areas (no vigilant conventional 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).

4.4 Safety vision

4.4.1 Background

According to the German traffic accident statistics published by the Federal Statistical Office of Germany[2] over 98 % of traffic crash are caused, at least in part, by humans, see Reference [2]. Similarly, the US National Highway Traffic Safety Administration (NHTSA) reports that up to 94 % of serious vehicle crashes in the US are caused at least in part by human error, see Reference [3]. Therefore, introducing automated driving poses great potential for reducing the rates of crashes and accidents. However, there are also major challenges in realizing the full safety benefit of automated driving in order to make automated driving systems reasonably safe and achieve the target of the avoidance of unreasonable risk as recommended by ISO/PAS 21448, and a "positive risk balance compared to human driving performance", as one of the recommendations of the German Ethics Commission in June, 2017[4] focusing on level 4 and level 5 systems without excluding level 3 systems.

Taking a deeper look at the statistics published by Reference [2] which also serve as an indicator of human driving performance, it can be argued that human beings are a reasonable factor for traffic safety. 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.

[1] Under preparation. Stage at the time of publication: ISO/SAE DIS 21434:2020.

4.4.2 Positive risk balance and avoidance of unreasonable risk

Automated driving can improve performance in most situations compared to that of conventional drivers. However, it cannot completely eliminate the risk of accidents or crashes.

Avoidance of unreasonable risk is a major measure to claim an acceptable level of safety. Its evidence is based on the application of a proactive and reactive driving behaviour, avoidance of crashes as much as "practically possible" while reasonably safe and the avoidance of discrimination on basis of any road user-related characteristics. These judgements are typically made on basis of a combination of qualitative and quantitative assessments, and also on an understanding of good engineering practice and existing standards. It can be argued that applying existing standards and engineering state-of-the-art will result in safer products.

A positive risk balance is a second major measure of an ethically acceptable level of safety. The evidence used for a positive risk balance can be derived from traffic accidents statistics to define acceptance criteria. It is important that validation target values are based on conservative traffic accident statistics of the intended target market, covering dense traffic and challenging traffic scenarios. Different validation target values might apply for different markets. In the long term state-of-the-art automated driving systems will be reflected in the traffic accident statistics and will continuously increase the validation target values of new automated driving systems.

The goal of this document is to present a general approach for tackling the risks introduced by automated vehicles. While this general approach is to be interpreted as a baseline for safe automated driving, it does not define a specific product that is complete and safe.

4.4.3 Principles of safety and cybersecurity for automated driving

The general approach of this document is based on the principles of safety and cybersecurity (PSC) for automated driving systems. They are derived from the considerations of the positive risk balance and the avoidance of unreasonable risk as presented above. Additionally, they are comprising a collection of publications and recommendations from the main public authorities or consumer associations (see Annex C and References [3]-[8]). These principles provide a foundation for deriving a baseline for the overall safety requirements and activities necessary for the different automated driving functions.

The principles of safety and cybersecurity for automated driving are clustered in three groups, each with a dedicated common area of influence (see Figure 3).

The first group named "automated vehicle and related aspects" is addressing overarching aspects on vehicle level including the automated driving system and human factors. Nevertheless, these aspects are essential to achieve overall safety and cybersecurity. This covers technical features as well as process related aspects.

The principles of the second group named "automated driving system" focus on the main technical aspects of the system. These principles constitute functionality related to safety and cybersecurity of the automated driving system itself.

The third group is named "human factors" and is addressing all safety and secure interaction aspects of the ADS and the user. This includes the role of user, clear repartition of role between the user and the ADS as well as takeover scenarios in both directions.

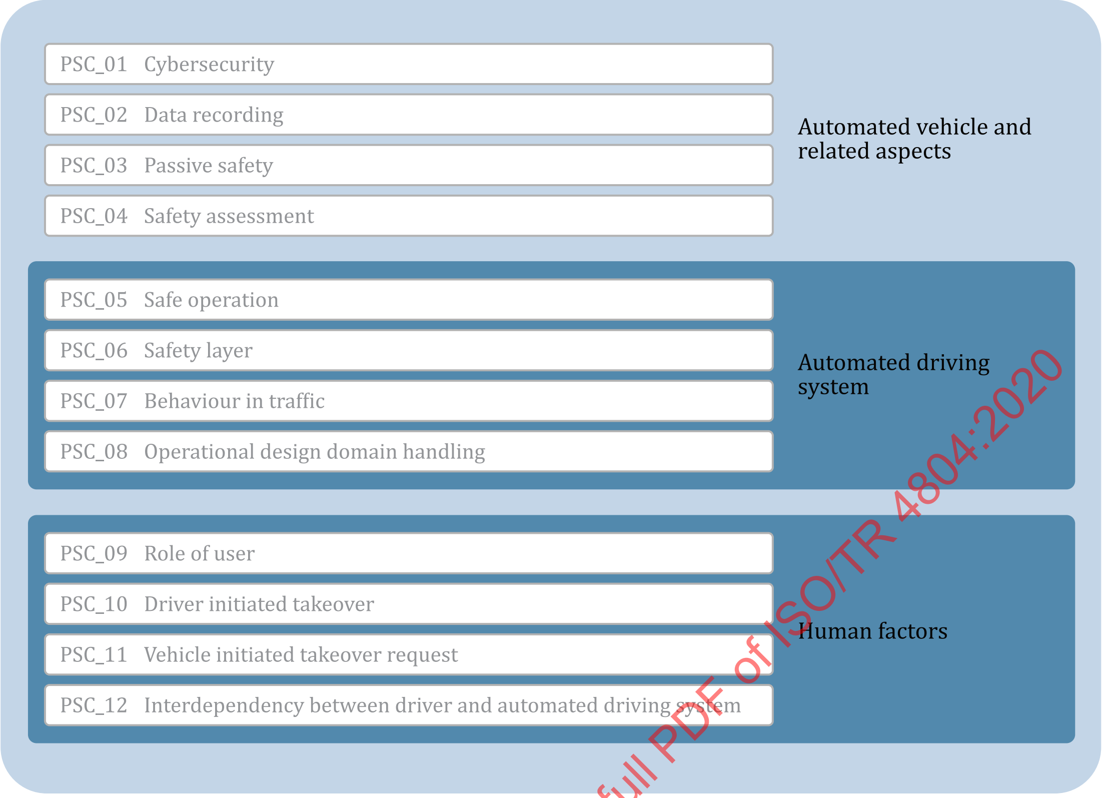
PSC_01 Cybersecurity
PSC_02 Data recording
PSC_03 Passive safety
PSC_04 Safety assessment
Automated vehicle and related aspects
PSC_05 Safe operation
PSC_06 Safety layer
PSC_07 Behaviour in traffic
PSC_08 Operational design domain handling
Automated driving system
PSC_09 Role of user
PSC_10 Driver initiated takeover
PSC_11 Vehicle initiated takeover request
PSC_12 Interdependency between driver and automated driving system
Human factors

Figure 3 — Groups of principles of safety and cybersecurity for automated driving

The purpose of this document is to highlight safety and cybersecurity-relevant aspects of developing, producing, operating and maintaining automated driving systems the combination of which leads to a safe product on the road. The aspects brought forward contribute toward a foundation for the safety and cybersecurity of automated vehicles.

4.4.3.1 PSC_01: Cybersecurity

When providing an automated driving system, steps are taken to protect the automated driving system from cybersecurity threats.

4.4.3.2 PSC_02: Data recording

Automated vehicles record the relevant data about the status of the automated driving system when a crash is recognized in a manner that complies with the applicable data privacy laws. Data recording must be sufficient to support forensic determination of crash root cause.

4.4.3.3 PSC_03: Passive safety

Crash scenarios

If existing, the vehicle layout considers collision scenarios resulting from vehicle automation.

Alternative seating positions

Occupant protection is considered even when the customer has new uses for the interior that are made possible through the automated vehicle.

NOTE Passive safety is not further detailed in this document as it is fully covered by other standards and regulations.

4.4.3.4 PSC_04: Safety assessment

Verification and validation ensure that the safety requirements (ISO 26262 series and ISO/PAS 21448) and safety relevant security requirements (ISO/SAE 21434) are met.

4.4.3.5 PSC_05: Safe operation

Dealing with degradation

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

- is capable of compensating and transferring the automated driving system to a safe condition/state (without unreasonable risk); and
- ensures a safe transition of control to the driver.

Fail-operational (limited to the safety-related function or component)

The loss of safety-related functions or system components do not lead to an unsafe situation.

4.4.3.6 PSC_06: Safety layer

The automated driving system recognizes system limits, especially those that do not allow a safe transition of control to the driver, preventing him or her from reacting to and minimizing the risk. The recognition is performed at least before any driving-cycle or activation of the automated driving system functionality.

4.4.3.7 PSC_07: Behaviour in traffic

Manners on the road

It is important that the behaviour of the automated driving function is be easy-to-understand for surrounding other road users, and also predictable and manageable.

Conforming to rules

It is important that the automated driving system complies with all applicable traffic rules. Crash avoidance manoeuvres can be prioritized over traffic rules if these are not endangering other road users. The principle PSC_09 role of user describes the remaining user responsibilities.

4.4.3.8 PSC_08: Operational design domain handling

ODD determination

As soon as the automated driving system recognizes that system limits, that restrict the safe functionality of the automated driving system are approached or reached, the automated driving system is reacting to compensate or to issue a driver takeover request with a time frame sufficient for takeover in reasonably predictable situations and contexts.

Manage typical situations

The automated driving system takes the relevant situations into account that can reasonably be expected in the ODD and address possible risks including other road users.

4.4.3.9 PSC_09: Role of user

To promote safety, it is important that the driver's state (i.e. state of attentiveness, see Reference [9] and 6.4.1) is suitable for a responsible takeover procedure. The automated driving system keeps the user informed about his responsibilities concerning the required tasks. It also informs the respective driver about safety-relevant driving situations in unmanned driving services.

Responsibilities

It is important that the aspects of the driving task which remain under the driver's responsibility are clear to the driver before engaging, while operating and disengaging the automated driving system.

Operating mode awareness

It is important that the automated function ensures that the currently active driving mode can be clearly seen or heard at any time. In addition, a change in driving mode is clearly apparent to the user as well.

4.4.3.10 PSC_10: Driver initiated takeover

Engaging and disengaging the automated driving system requires an explicit interaction from the driver, indicating a high confidence of intent.

NOTE The driver takeover is prioritized against the automated driving system.

4.4.3.11 PSC_11: Vehicle initiated takeover request

Minimal risk condition

If the driver does not accept a takeover request, the automated driving system performs a manoeuvre to minimize risk, resulting in a minimal risk condition. This manoeuvre depends on the situation and the current performance of the automated driving system.

Takeover requests

It is important that vehicle-initiated takeovers are clearly understandable and manageable for the driver.

4.4.3.12 PSC_12: Interdependency between driver and automated driving system

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

5 Systematically developing dependability to support safety by design

5.1 General

This clause describes how the three dependability domains, safety of the intended functionality, functional safety and cybersecurity, work together and how to combine them to create a dependable system. The clause 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 4).

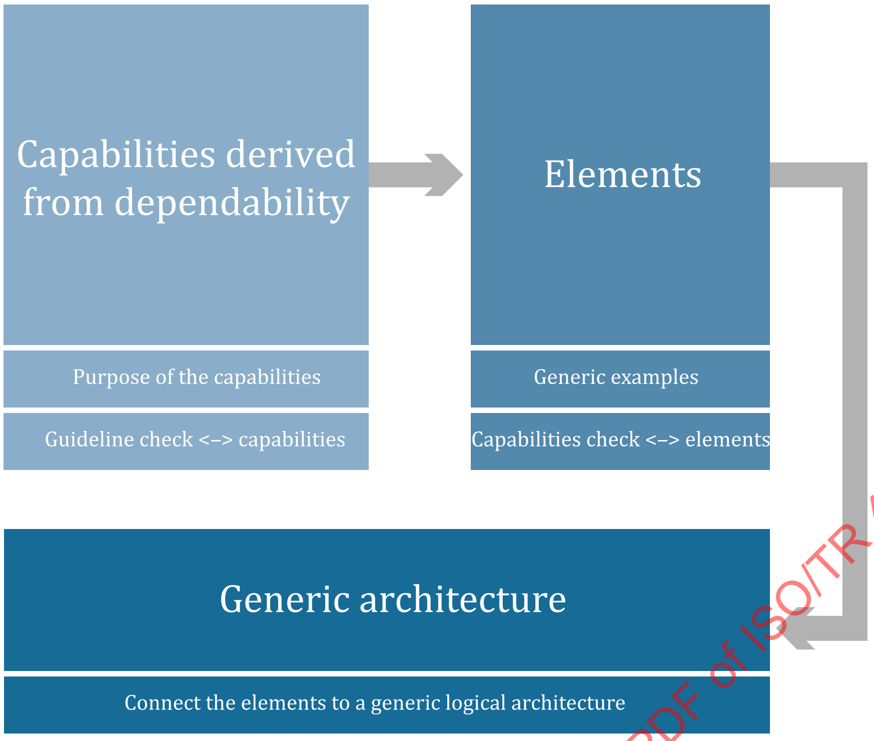
Capabilities derived from dependability
Purpose of the capabilities
Guideline check <–> capabilities
Elements
Generic examples
Capabilities check <–> elements
Generic architecture
Connect the elements to a generic logical architecture

Figure 4 — Systematic development of dependability

5.2 Deriving capabilities of automated driving from dependability domains

Deriving capabilities from dependability domains begins with different worldwide regulatory publications for automated driving systems to identify the requirements that capabilities need to cover in addition to the principles of safety and cybersecurity for automated driving systems. The capabilities cover SOTIF, which includes human factors, functional safety and cybersecurity, which both address the logical and technical architecture. Input requirements are provided from each domain to the others. This subclause provides advice on cybersecurity approaches and measures.

5.2.1 Applying the related safety standards

To ensure a necessary degree of safety, it is important that the system controls safety relevant use cases that result either from the intended use or from unlikely electrical and/or electronic (E/E) faults. The ISO 26262:2018 series and ISO/PAS 21448 use the term foreseeable misuse in the hazard analysis phase of the product development. While performing the foreseeable misuse analysis various actor types are considered which may include both compliant and malevolent vehicle occupants and external actors. The malevolent actor type is one that aligns with both ISO/SAE 21434 and allows for the consideration of intentional misuse or damage to the automated driving system. This is specifically related to systems which relieve the conventional driver from the driving task due to higher automation (SAE L2 and higher). This perspective is partially covered in well-established development standards like the ISO 26262 series and ISO/PAS 21448. This document has the aim for more guidance specifically for the application of the related safety standards for automated driving systems.

Existing standards do not present detailed solutions to some of the most relevant 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 Annex B), human factors and psychology, and the technological capability of the sensory devices used as inputs to the automated driving system. Nevertheless, it is important that safety-related use cases are analysed 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.

According to ISO/PAS 21448, initially a safe functionality needs to 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 series, whereby danger as a result of deliberate manipulation is assessed from an ISO/SAE 21434 cybersecurity point of view. Implementing the safety standards ISO/PAS 21448, the ISO 26262 series and ISO/SAE 21434 would allow 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 driving system functions, the risks based on the functional and system limits are evaluated. Performance limitations (e.g. based on sensor limitations) could potentially introduce hazardous situations. Development standards will aid developers in managing the complexity of the systems, estimating possible risks and addressing adequate measures.

Finally, availability of safety-related capabilities of a system is an important topic, which is partially covered by the ISO 26262 series.

As stated in ISO/PAS 21448, it becomes increasingly important to approach the problem from an analytical perspective, addressing the safety of the system by designing scenario-based system behaviours and addressing the technological capability including availability aspects through the analysis of use cases and scenarios to design a robust and safe system. This document defines a typical set of arguments and technological capabilities as the concept of safety by design, and this is the approach that is detailed throughout this document.

To achieve the balance between safety with respect to fail-safe behaviour and availability of customer functionality, the design is analysed 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.

5.2.2 ISO/PAS 21448 - Safety of the intended functionality

The basic concept of the ISO/PAS 21448 - Safety of the Intended Functionality (SOTIF) approach is to introduce an iterative function development process (see Figure 26) that includes verification and validation. This approach leads to an intended function which can be declared safe by applying progressive improvement during development. Several activities will be derived based on an approach that argues that these activities are adequate for developing a safe automated function. This approach assumes that there is an area of known scenarios with defined behaviour and an unknown area with potential harm. The automotive development goal is to reduce the impact of known potentially harmful scenarios and the amount of unknown scenarios to an acceptable level of residual risk.

Clause 5 can be referred to at this point since the main evidence to demonstrate that the system is reasonably safe enough for road users is provided via verification and validation activities.

5.2.3 ISO 26262 series - Functional safety

In the ISO 26262 series the risk is determined and communicated or mapped using automotive safety integrity levels (ASIL). These levels are representing related requirements. This document 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 subclauses below.

A starting point in the ISO 26262 series is the risk determination based on a defined function including a first preliminary architecture. Therefore, the approach of ISO/PAS 21448 SOTIF can be used to support – applying functional safety in accordance with the ISO 26262 series – the creation of the item definition. This item definition per the ISO 26262 series needs to 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 goals for the function and its related system. The next technical steps are developing functional and technical safety concepts.

The first edition of the ISO 26262:2011 series 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 completely address very complex, distributed systems and how to deal with availability requirements. The second edition resolves some of these issues, but further interpretations are needed.

Meeting all challenges will result in the definition of the safe intended functionality. In other words, weaknesses of the specification or the technologies have been considered (SOTIF) and possible E/E faults can be controlled by the system or by other measures (the ISO 26262 series). Consequently, it will be possible to declare the automated system safe without considering cybersecurity issues, which are currently not covered by ISO/PAS 21448 or the ISO 26262:2018 series.

5.2.4 ISO/SAE 21434 - Automotive cybersecurity

Cybersecurity focuses on risks presented by active adversaries in the form of creative, determined and malicious entities acting intentionally. This leads cybersecurity to utilize additional analysis tools and technical mechanisms that nevertheless also affect safety.

The ISO/SAE 21434 describes the tasks for a secure development process, with a special focus on risk analysis. For automated driving systems the ISO/SAE 21434 is therefore fully applicable. However, while using a certain risk approach, it is important to think about specifics of automated driving.

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. Connectivity additions include new interfaces between the control functions of connected vehicles, IT backend systems, and other external information sources (see Figure 5). This rich attack surface creates considerable interest for malicious actors with various goals. As the connectivity and level of automation increase, the likelihood of an attack will increase. Therefore, the cybersecurity risk will be higher without proper mitigation.

The other dimension of a cybersecurity risk is the impact of an attack. Here, there are also differences between human controlled driver assistance functions and system controlled automated driving functions. As the driver is out of the loop, they are not able to recognize an attack and to react to it. So, the impact is probably higher than in actual released systems. There are approaches to use the safety analysis results as input for cybersecurity analysis, see ISO/SAE 21434:—, Clause 8, RQ-08-07.

Another interesting overlap is between the disciplines of cybersecurity and safety of the intended function. There are some real-world attacks focus on vehicles but without directly attacking them. For example, if the lane marking on the road is changed, or if a misleading traffic sign is projected on a house wall, the vehicle may be deceived into leaving its lane. It is acknowledged that these kinds of attacks are neither solely a cybersecurity nor a SOTIF topic. It is rather a joint effort to account for these topics, probably with additional domains. Thereto, one might use methods from both domains as follows. The risk on objects outside the vehicle is considered by the different standards. It is important that mitigations are evaluated, both inside and outside the vehicle [see ISO/SAE 21434:—, 8.9, RQ-08-12 (risk transfer)]. In the end, it may not be the task for road vehicle cybersecurity to mitigate these risks with typical cybersecurity measures like integrity checks or proofing the authenticity as the attacked environment is not cyber-physically connected to the considered system. Also, for SOTIF, it is hard to cover these scenarios as it deals mainly with incorrect usage of the system itself, instead of the environment. Nevertheless, these attacks and risks are considered when defining the ODD. Resulting measures to cover these risks have then also to be covered by means of validation.

In short, we have advanced to a level where vehicles cannot maintain a safe state unless they also operate securely. Most importantly, cybersecurity processes of ISO/SAE 21434 are applied as a starting point to ensure that for example 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.

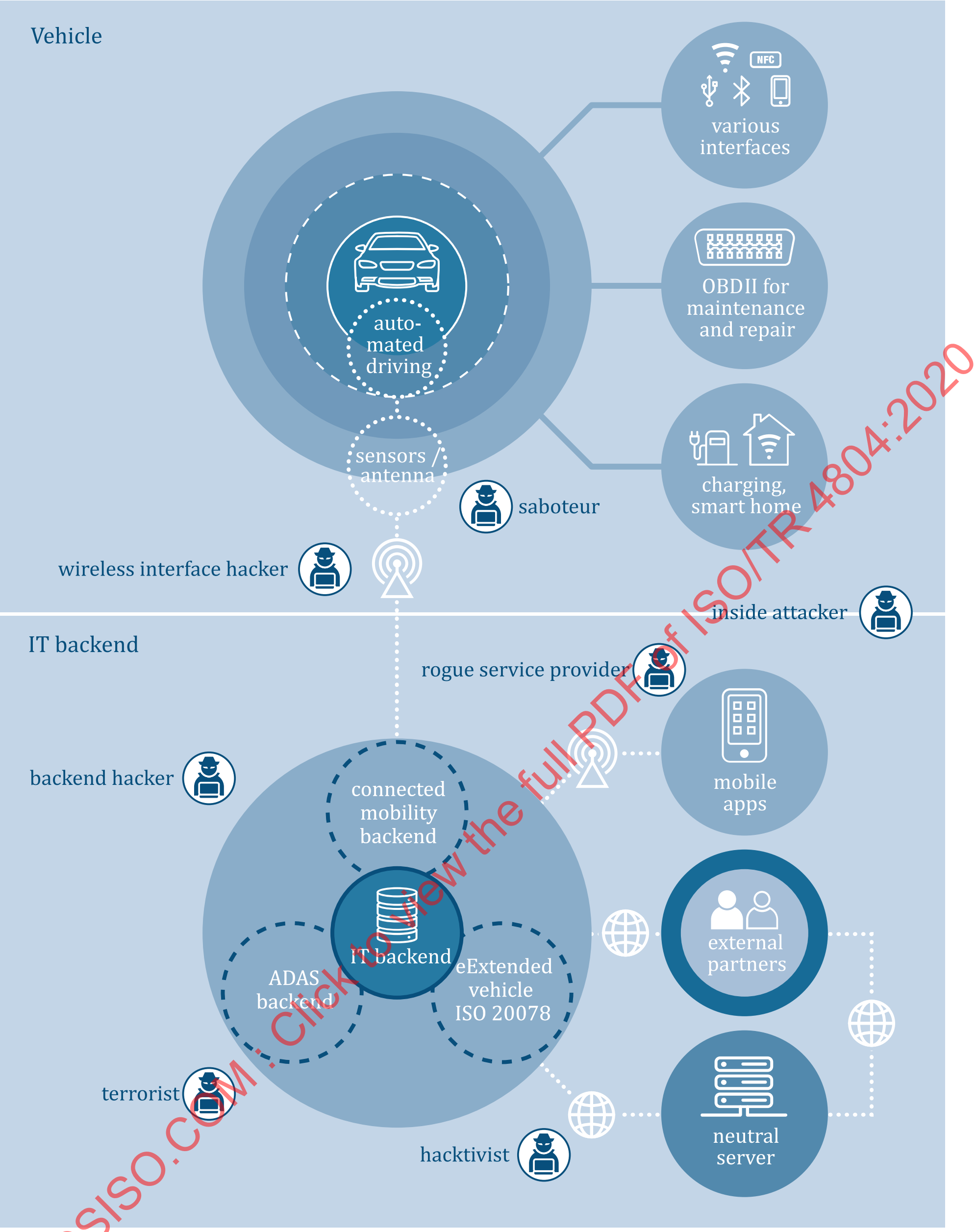

Figure 5 — Automotive cybersecurity

5.2.5 Capabilities of automated driving

5.2.5.1 Initial derivation of capabilities

An automated driving system that complies with the principles outlined in 4.4.3 has a basic set of system properties that are specified here as capabilities. The following will discuss the capabilities present when the overall system is considered safe.

The capabilities are divided into fail-safe capabilities (FS) and fail-degraded capabilities (FD). Fail-safe capabilities provide and enable customer value, but they can also bring the system to achieve a minimal risk condition in the event of failure. Fail-safe capabilities can be discontinued, because the safety relevance of their unavailability is low enough or is covered by the fail-degraded capabilities.

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 and perception (including localization), planning and control, and actuation and stability provide a general, implementation-independent view of the automated driving system. Figure 6 illustrates this general model:

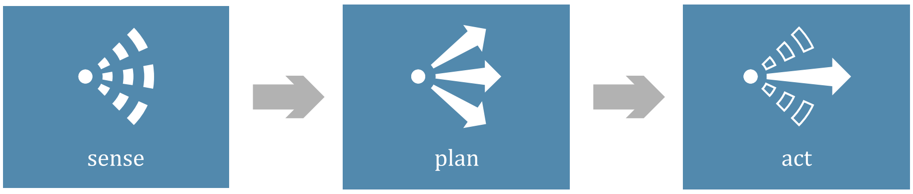

Figure 6 — Sense – plan – act design paradigm

Based on the allocation of capabilities to the basic functions for sense – plan – act, it is possible to allocate requirements for elements that the automated vehicle is reasonably safe as depicted in Figure 7.

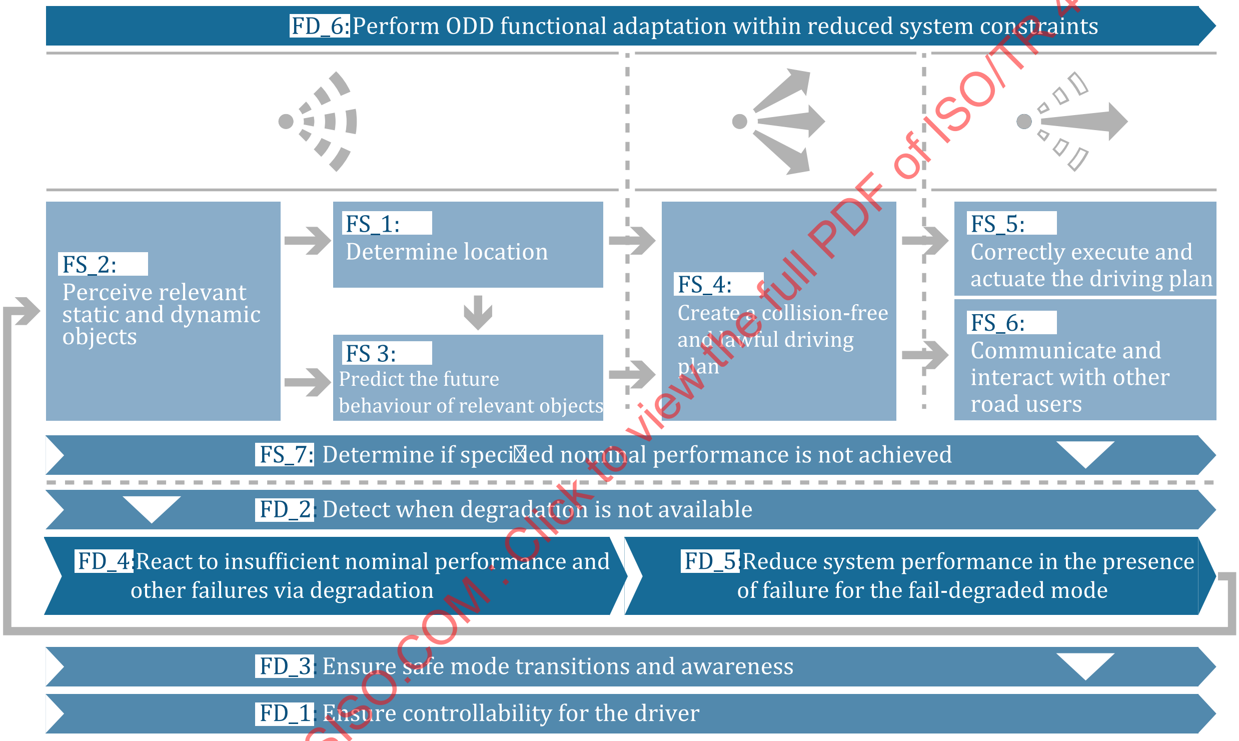
FD_6: Perform ODD functional adaptation within reduced system constraints
FS_2: Perceive relevant static and dynamic objects
FS_1: Determine location
FS 3: Predict the future behaviour of relevant objects
FS_4: Create a collision-free and lawful driving plan
FS_5: Correctly execute and actuate the driving plan
FS_6: Communicate and interact with other road users
FS_7: Determine if speciﬁed nominal performance is not achieved
FD_2: Detect when degradation is not available
FD_4: React to insufficient nominal performance and other failures via degradation
FD_5: Reduce system performance in the presence of failure for the fail-degraded mode
FD_3: Ensure safe mode transitions and awareness
FD_1: Ensure controllability for the driver

Figure 7 — Realizing fail-safe and fail-degraded capabilities

5.2.5.2 Overview of the capabilities

FS_1: Determine location

The system is able to determine its location in relation to the ODD. The vehicle is 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.

NOTE The location is stored as initial situation of any realization of capabilities, including the relevant perceived proximity information.

FS_2: Perceive relevant static and dynamic objects

All entities that an automated driving system requires for its functional behaviour are perceived, optionally pre-processed, and provided correctly. The highest priorities are placed on entities with an associated risk of collision, also confirming their plausibility. Sample entities include dynamic objects (e.g. other road users and characteristics of the respective movement), static instances (e.g. road boundaries, traffic guidance and communication signals) and obstacles.

FS_3: Predict the future behaviour of relevant objects

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

FS_4: Create a collision-free and lawful driving plan

To ensure a collision-free and lawful driving policy, the following is respected.

Maintaining a safe lateral and longitudinal distance to other objects:

- complying with all applicable traffic rules whenever and wherever the automated vehicle is driving;
- considering potential areas where objects, other road users, or animals may be occluded;
- in unclear situations the right of way is given, not taken;
- if a crash can be avoided without endangering other road users, traffic rules may be compromised if necessary, to avoid injury or preserve life;
- considering the inadequacy and uncertainty of other road users and automated vehicles.

FS_5: Correctly execute and actuate the driving plan

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

FS_6: Communicate and interact with other road users

For safety-related use cases and depending on the ODD, automated vehicles communicate and interact with other road users.

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 behaviour. Therefore, the adverse nominal performance of the system or violation of ODD conditions are detected by appropriate mechanisms. FD_4 covers the reaction to detected adverse behaviour.

Typical aspects for influencing the nominal performance are:

- unwanted human factors, including foreseeable misuse and manipulations;
- deviation of the intended functionality;
- technological limitations;
- environmental conditions;
- systematic and random failures.

Capabilities for recovering to nominal performance, as specifically defined herein, are possible but are not considered further in this document, 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 driver

The vehicle operator’s level of control varies depending on the automation level as per SAE J3016:2018 and the use case definition and therefore is ensured.

FD_2: Detect when degradation is not available

It is ensured that a possible unavailability of the fail-degraded capability is detected. If the degradation strategies depend on the degradation reason, the degradation reason is identified.

FD_3: Ensure safe mode transitions and operating mode awareness

Transitions of driving mode are performed correctly and controlled by the affected vehicle operator if necessary. It is important that the affected driver is aware of the current driving 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 with the vehicle operator taking control again or misusing the system and it will be deactivated prior to leaving the ODD.

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), it is important that the system degrades within a well-defined amount of time.

FD_5: Reduce system performance in the presence of failure for the fail-degraded mode

The reaction in case of failures during fail-degraded mode is defined.

FD_6: Perform ODD functional adaptation within reduced system constraints

Automated driving system operation with ODD functional adaptation is actuated as nominal capabilities with new limits. Multiple functional adaptations are possible. The new limits are defined such that the functional adaptations are safe. If suitable, a safe functional adaptation can be a non-permanent operation with a defined time frame for an additional required reaction.

The selection matrix in Table 2 is an example to demonstrates the state of completeness of the derived capabilities to provide evidence of traceability to the principles from 4.4.3. Such kind of table needs to be adapted for a concrete development project.

Table 2 — Example of a selection matrix for the traceability of the capabilities

ID	**PSC_01** Cybersecurity	**PSC_02** Data recording	**PCS_03** Passive safety	**PSC_04** Safety assessment	**PSC_05** Safe operation	**PSC_06** Safety layer	**PSC_07** Behaviour in traffic	**PSC_08** Operational design domain handling	**PSC_09** Role of user	**PSC_10** Driver initiated takeover	**PSC_11** Vehicle initiated takeover request	**PSC_12** Interdependency between driver and ADS
FS_1 Determine location	×			×			×	×				
FS_2 Perceive relevant static and dynamic objects	×			×			×					
FS_3 Predict the future behaviour of relevant objects	×			×			×					
FS_4 Create a collision-free and lawful driving plan	×			×			×					
FS_5 Correctly execute and actuate the driving plan	×			×			×					
FS_6 Communicate and interact with other road users	×			×			×					
FS_7 Determine if specified nominal performance is not achieved	×			×		×		×				
FD_1 Ensure controllability for the driver	×			×	×				×	×	×	×
FD_2 Detect when degradation is not available	×			×	×							
FD_3 Ensure safe mode transitions and operating mode awareness	×			×	×	×			×	×	×	×
FD_4 React to insufficient nominal performance and other failures via degradation	×			×	×	×						
FD_5 Reduce system performance in the presence of failure for the fail-degraded mode	×			×	×	×						
FD_6 Perform ODD functional adaptation within reduced system constraints	×			×	×	×		×			×	

As shown in Table 2, 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 subclauses where the full logic and rationale surrounding the methodology for validating an automated driving system is developed.

5.2.6 Minimal risk conditions and minimal risk manoeuvres

A minimal risk manoeuvre (MRM) is the system's capability of transitioning the vehicle between minimal risk conditions (MRC). The concept of MRCs and MRMs are derived considering safety at vehicle level and are defined in accordance with the principles of the ISO 26262 series as an operating mode (in the case of a failure) of an item with a tolerable level of risk. In terms of ISO 26262-1:2018, an MRM is an emergency operation to reach an MRC – referred to as a safe state. This document expands the definition to also include fail-degraded mode 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 8 visualizes this general principle.

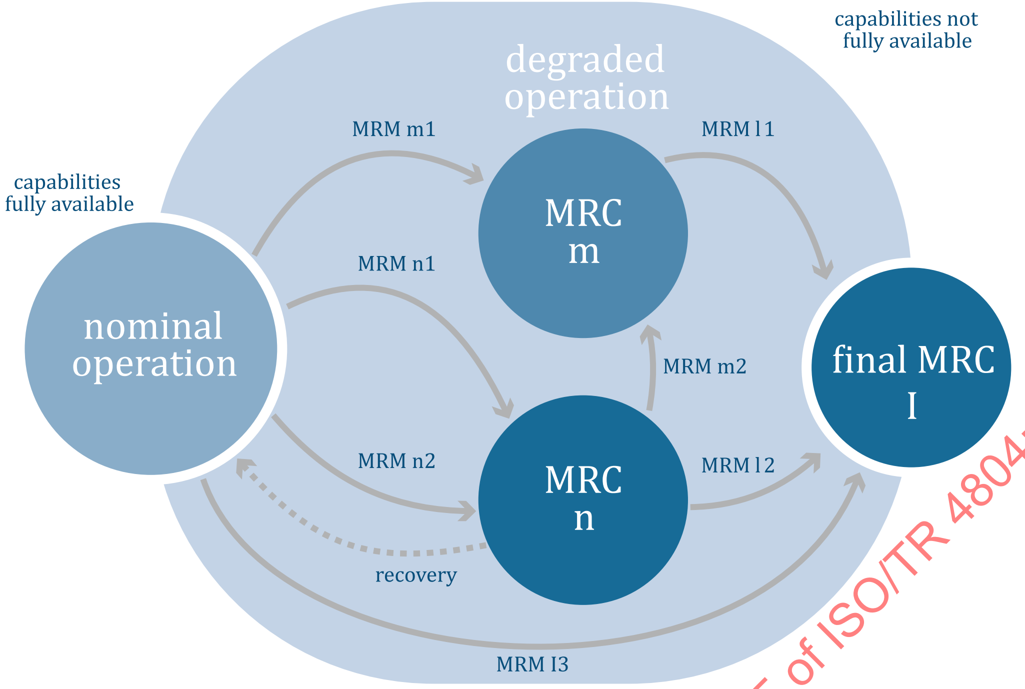

Figure 8 — Minimal risk manoeuvre and minimal risk condition

The purpose of the MRM is to transfer the vehicle to a minimal risk condition. 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 fail-degraded mode and the remaining fail-degraded capabilities will reach a minimal risk condition by executing an appropriate minimal risk manoeuvre. The fail-degraded mode is a time-limited operational domain in which the frequency of its occurrence needs to be reduced whenever possible. The acceptable time for the fail-degraded mode depends on the remaining capabilities in the current system. One principle concept of the ISO 26262 series is that every order of magnitude in reducing the frequency of possible harm lowers the required safety integrity level of the automated driving system. Table 3 defines the conditions between which the MRM will allow for safe transition. Here, “Nominal Operation” is handled like an MRC. Specific examples are outlined in Annex A.

Table 3 — Minimal risk conditions

ID	MRC	Definition	Possible reasons
MRC_1	Takeover by the driver	The driver has completely taken over the driving task.	— Known future limitation in ODD — Limitations or the driver (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.	— Deviations from nominal state, reduced capabilities
MRC_3	End of operation	This condition describes a vehicle state that allows safe deactivation of the function.	— Severe system failures, loss of capabilities, missing driver takeover

The following MRMs are proposed:

Table 4 — Proposed MRMs

ID	MRM	Definition	Target condition
DTO	Driver takeover request	Request takeover by the driver.	MRC_1 Takeover by the driver
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 failure, next possible stop when suitable according to driving situation and surroundings to end operation.	MRC_3 End of operation
MRM_5	Emergency stop	Due to severe failure, immediate stop is necessary to avoid unreasonable risk.	MRC_3 End of operation
Recovery	Recovery	Limitations of capabilities are resolved and therefore nominal state is reached again.	Nominal state

Using the list of MRMs in Table 4, the potential failure modes are reflected within the overall system. It is important to apply several analysis methods from related safety standards (e.g. failure analysis techniques such as FMEA or FTA, analysis for the intended use of the system and 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.

5.3 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 Annex A can be considered possible real-world implementations that fulfil the capabilities introduced in 5.2.2. This subclause provides guidance about fulfilling the fail-safe and fail-degraded capabilities with real-world elements generically, but in a manner to ensure a reasonable level of safety. 5.3.1 discusses possible implementations of each capability by introducing elements. 5.3.2 then discusses each element in detail.

5.3.1 Implementing the capabilities

The following describes the capabilities in relation to their associated elements. Elements in light blue are for the sake of clarity displayed without further inputs. Each of those elements is represented as a whole with all relevant inputs in one figure only.

5.3.1.1 FS_1: Determine location

Location information can be used to select correct lane (e.g. for making turns) or to know which local traffic rules apply. In addition to that it is ensured that the automated driving system operates only within specified system limits if the intended use of the automated driving system is restricted. Therefore, it is important that the automated driving vehicle is located adequately, using fused environmental sensor information from sensor fusion algorithms (see 5.3.1.4). 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 vehicle is about to exceed an ODD limit. This is a precondition for activating the automated driving system, which prevents operators from misusing the automated driving system outside the ODD (see Figure 9).

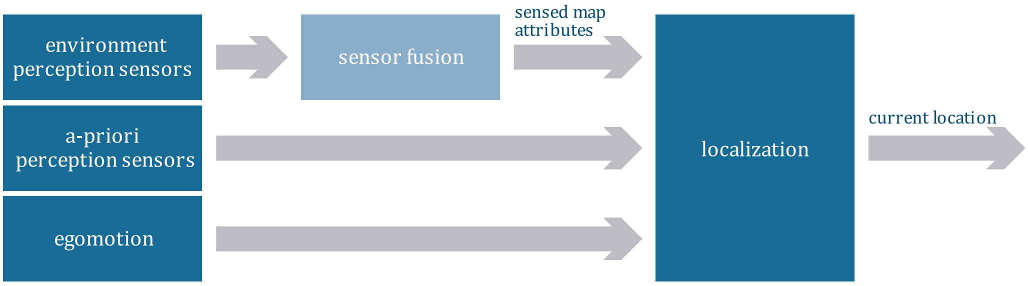

Figure 9 — FS_1: Determine location

5.3.1.2 FS_2: Perceive relevant static and dynamic objects

All entities that an automated driving system requires to account for its functional behaviour are perceived, pre-processed and provided safely. The highest priority is placed on entities with the highest associated risk of collision. Example entities include dynamic instances (e.g. other 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 (see Figure 10).

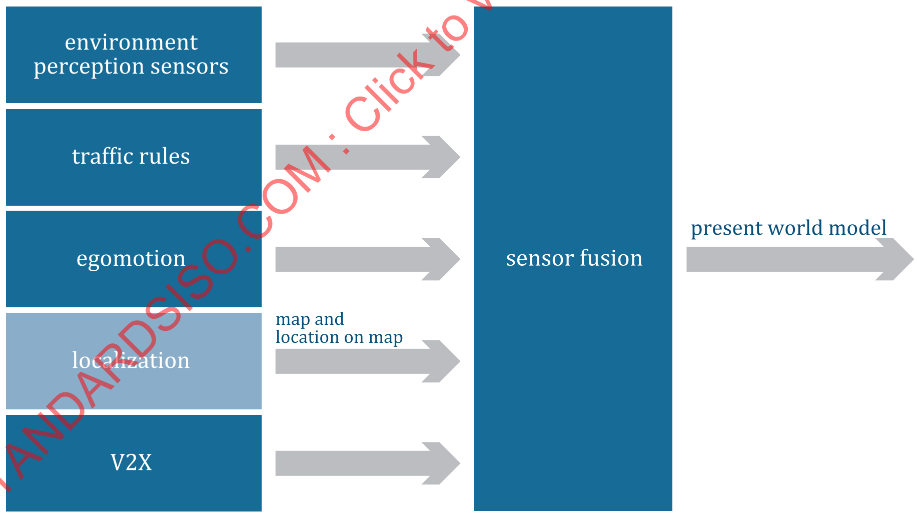

Figure 10 — FS_2: Perceive relevant static and dynamic objects

5.3.1.3 FS_3: Predict the future behaviour of relevant objects

The present world model recorded as the output of FS_2 (5.3.1.2) may not suffice as an input for the safe and lawful creation of a driving plan FS_4 (5.3.1.4). Therefore, it is important to extended it 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” (see Figure 11). It also considers the intention of other dynamic objects and obstacles including those that are partially occluded as the basis for predicting future motions. Additionally, current environment conditions such as low road friction and reduced sensor performance (e.g. fog, mist, heavy rain) are taken into consideration for the prediction. Depending on the item definition even V2X can be used to get information about the relevant surrounding objects (e.g. valet parking systems).

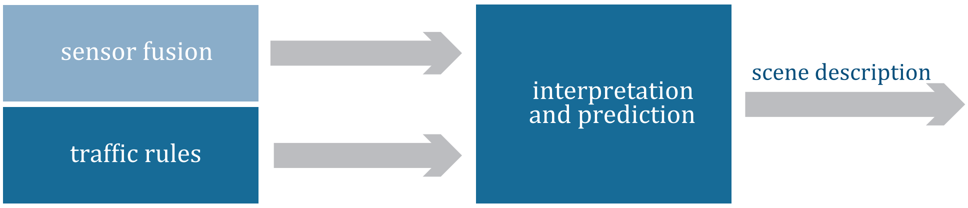

Figure 11 — FS_3: Predict the future behaviour of relevant objects

5.3.1.4 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 first accurately senses its environment (not shown in Figure 12) and performs localization before any driving plan can be created. With localization performed and an accurate world model provided, the vehicle considers safety-relevant other road users in the world model and what their tracked motion suggests from an interpretation and prediction standpoint. This provides a baseline for possibly reasonable assumptions that can be considered regarding other road users that have been detected.

The automated driving system obeys traffic rules for the drive planning element to produce a lawful driving plan, unless crash avoidance manoeuvres can be prioritized over traffic rules to prevent collisions.

The automated driving system's egomotion may also be considered as physical properties limit the set of possible manoeuvres (physical properties not needed to explain). Finally, the ADS mode manager is considered so that drive planning is aware of whether it is in normal operation mode or a fail-degraded/ minimal risk condition mode. Figure 12 illustrates the elements required.

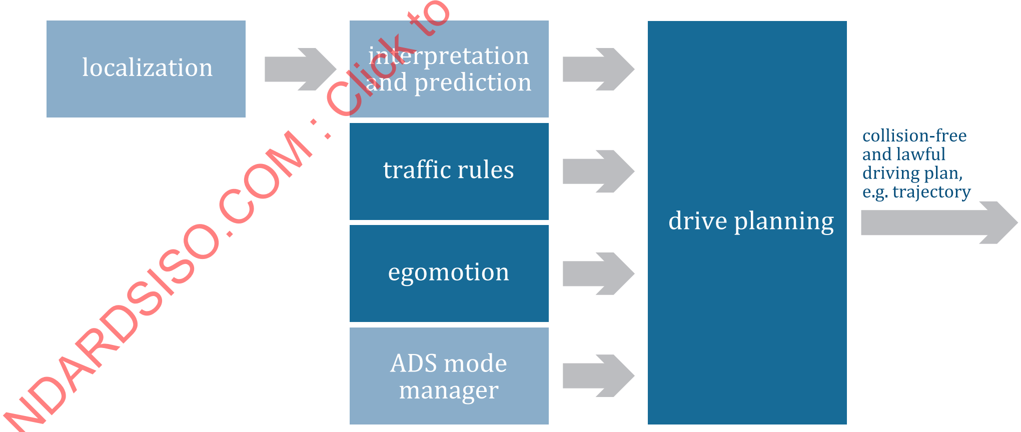
localization
interpretation and prediction
traffic rules
egomotion
ADS mode manager
drive planning
collision-free and lawful driving plan, e.g. trajectory

Figure 12 — FS_4: Create a collision-free and lawful driving plan

5.3.1.5 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 trajectory requested by the drive planning element's requested trajectory into the physical motion for the vehicle's motion actuators (e.g. steering, braking or powertrain) (see Figure 13).

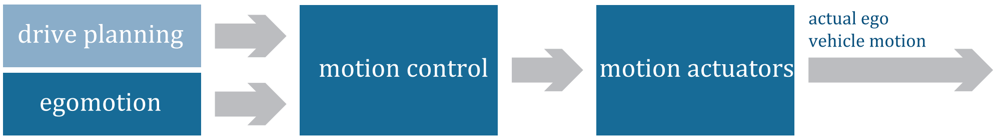

Figure 13 — FS_5: Correctly execute and actuate the driving plan

5.3.1.6 FS_6: Communicate and interact with other road users

The automated driving vehicle drives in a predictable manner (e.g. no lane change without prior turn indicator activation, no incautious lane change, foreseeable behaviour while approaching a lane merge). Similar to manual driving, means of communication include visual and sometimes acoustic indicators. V2X or other types of interaction could be a means of communication as well (see Figure 14).

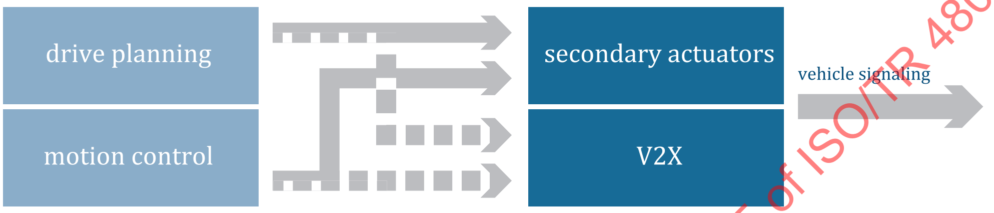

Figure 14 — FS_6: Communicate and interact with other road users

5.3.1.7 FS_7: Determine if specified nominal performance is not achieved

Monitors are in place to detect defined nominal performance limits at the element or system level with sufficient time to ensure a safe reaction (see Figure 15). The limits are derived by following the iterative approach described in Figure 4.

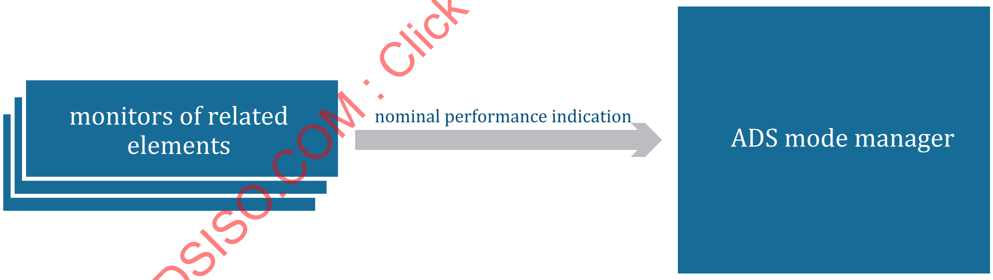

Figure 15 — FS_7: Determine if specified nominal performance is not achieved

5.3.1.8 FD_1: Ensure controllability for the driver

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 driver 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 so that he or she can concentrate on the driving task and regain situational awareness when a system takeover request is imminent. Therefore, the automated driving system continuously monitors the vehicle operator for possible distraction or mode confusion. Monitoring for possible mode confusion is also needed 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 realizes that the ODD is being exited. The driver may be an entity outside of the vehicle in question in case where the system foresees remote control. To ensure controllability for a local or remote driver, the ADS mode manager senses the vehicle operator's (conventional driver or a remote operator) wish to control the vehicle and reacts to this (see Figure 16).

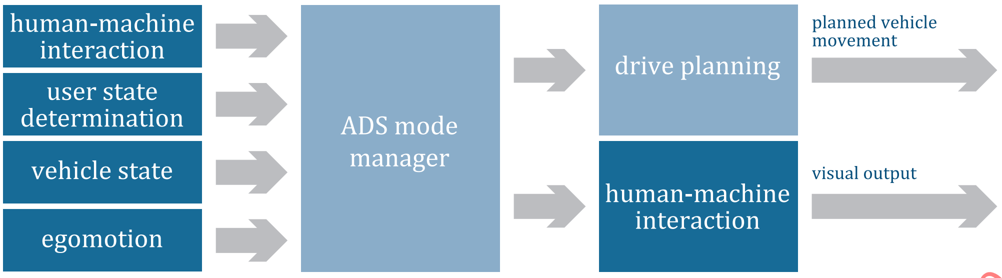

Figure 16 — FD_1: Ensure controllability for the driver

5.3.1.9 FD_2: Detect when degradation is not available

A monitor checks whether fail-degraded mode is available even if it is not actively being used in the nominal drive mode. Thus, the monitor continuously checks the availability of the fail-degraded mode that can be implemented in several ways (see Figure 17).

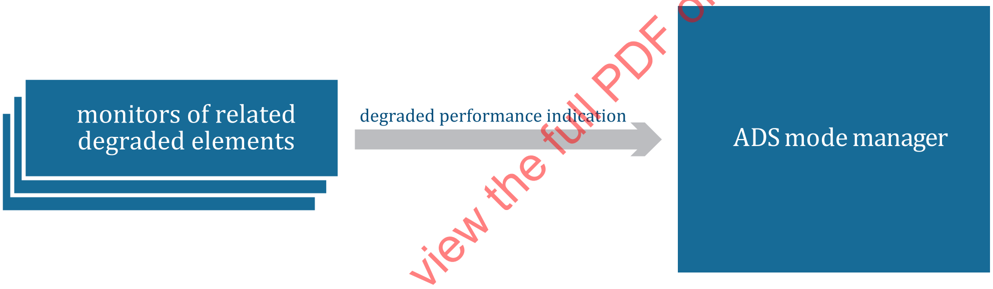
monitors of related degraded elements
degraded performance indication
ADS mode manager

Figure 17 — FD_2: Detect when degradation is not available

5.3.1.10 FD_3: Ensure safe mode transitions and operating mode 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 electrical, electronic and software 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 of ADS is allowed only if the automated driving system detects that the driver is back in the loop for controlling all functions of the vehicle or if the vehicle is in a safe state (see Figure 18).

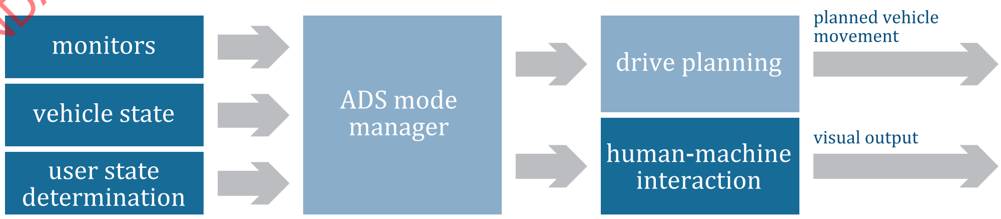

Figure 18 — FD_3: Ensure safe mode transitions and awareness

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

The system's reaction to insufficient nominal performance (see FS_7, 5.3.1.7) is defined, and the system reacts 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 (see Figure 19).

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

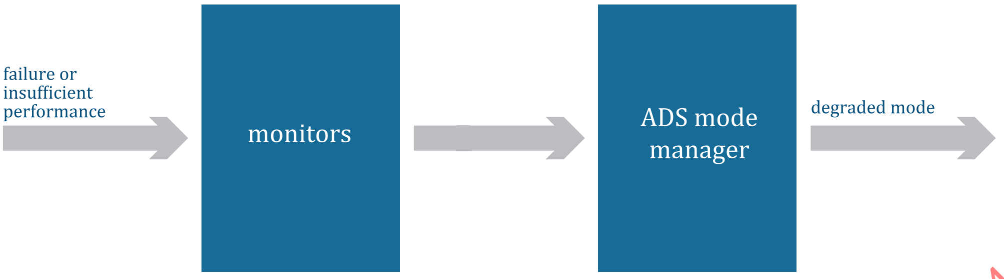

Figure 19 — FD_4: React to insufficient nominal performance and other failures via degradation

5.3.1.12 FD_5: Reduce system performance in the presence of failure for the fail-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 is 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 (see Figure 20). 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.

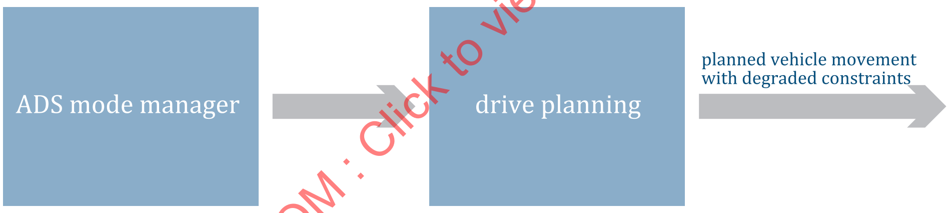

Figure 20 — FD_5: Reduce system performance in the presence of failure for the fail-degraded mode

5.3.1.13 FD_6: Perform ODD functional adaptation within reduced system constraints

Automated driving system operation with functional adaptation is operating with new defined limits for the nominal capabilities. If needed an MRM is carefully defined to achieve the MRC. In this case, the automated driving system is able to perform the DDT within a well-defined time frame (see Figure 21).

The elements needed for functional adaptation depend on the item definition. Depending on the ODD functional adaptation, adequate HMI operations are implemented.

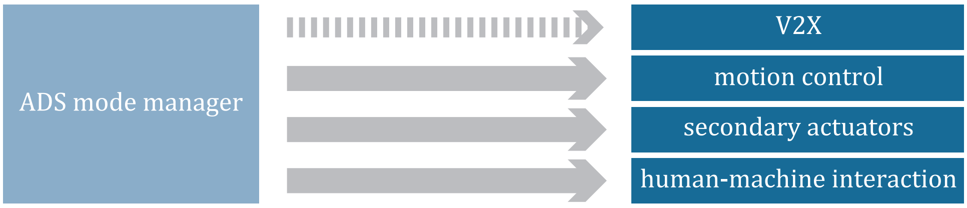

Figure 21 — FD_6: Perform ODD functional adaptation within reduced system constraints

5.3.2 Elements

Fail-degraded and ODD functional adaptation capabilities are implemented by the elements described in detail below. The general capability regarding cybersecurity as described in 5.2.4 is considered for each element. However, the specific cybersecurity measures are not described in each element, as this depends on cybersecurity concept and architectural design.

5.3.2.1 Environment perception sensors

The environment perception sensors cluster captures 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, other road users, obstacles, traffic signs and acoustic signals.

Sensor types: As of today, a single sensor in an ADS 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 may provide suitable coverage for the given specific product:

Camera

Digitally encoded representation of the visible environment similar to human perception. Main sensor for object/feature type classification. High sensitivity to weather conditions.

LIDAR

High-precision measurement of structured and unstructured elements. High sensitivity to material and colour of objects. Medium sensitivity to environment conditions. Insensitive to small targets.

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, emergency vehicles, and law enforcement. Thus, devices capturing acoustic signals are important for automation levels where the systems need to react to these.

It is important that sensor sets are 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 covers 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 design of the sensor cluster needs to detect and classify objects that may be reasonably expected within its ODD and safety relevant. Recognizing traffic signs also is taken into consideration. The detectability of external entities strongly depends on the material these are made of. This document 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 enhance the performance of individual sensor performance.

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

5.3.2.2 A-priori perception sensors

5.3.2.2.1 HD map

HD map as a reliable sensor

An in-vehicle map has not previously played a safety-related role comparable to what 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 driving system in certain situations. An HD map is, therefore, necessary as a reliable off-board sensor. It contains carefully processed a-priori information, which is useful for "detecting" features that are not easily detectable by on-board sensors. Also, an HD map provides a redundant source of information for on-board sensors, including location-based ODD determination, environment modelling in adverse conditions, and precise semantic understanding 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 (RMAs) 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 for different reasons including occlusion, abrasion or frequent changes. Therefore, reliable map attributes are detected correctly in safety-relevant use cases in order to meet the requirement on location-based ODD determination with a low false positive rate.

RMAs have the following properties:

- fused with on-board sensor inputs, a combination of RMAs is a sufficient condition to infer that the automated vehicle is reaching the boundary of the ODD;
- RMAs are reliably detectable by onboard sensors within the ODD;
- RMAs are observed with a relatively low real-world rate of change, so that the RMA failure rate can be controlled to a level to avoid unreasonable risk;
- the quality and freshness of RMAs are 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 4.3).

RMA deviations, 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 having a different 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, which are a new fault type because they neither can be treated as a systematic fault, nor as a random fault. Such deviations can possibly arise from:

- map inaccuracy;
- map data errors introduced during source data collection, map production, map creation and distribution processes, in-vehicle storage and usage;
- deviations and failures 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);

- map update delays;
- localization errors.

The above deviations and failures are addressed appropriately to ensure that the automated driving system is able to stay at or below 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, ISO/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, road construction and maintenance plans are fully transparent and easily accessible as indispensable public information, by all map providers. Errors as a result of real-world changes are difficult to monitor and control, thus they are carefully analysed. Changes of RMAs can be divided into two categories based on the impact that they have for the use case of an automated driving 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) and/ or a hazardous behaviour (e.g. missing traffic sign like a stop).

Therefore, deviations and failures of RMAs due to major changes are considered to have an impact on the failure rate of location-based ODD determination primarily, but also on environment modelling (e.g. intersection and traffic light identification). Several measures can be implemented to mitigate deviations and random failures. First, RMAs are 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 an important factor. 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 road changes and reduce the risk of unwanted RMA deviations and failures.

Other safety considerations:

Map modification after initial creation is a mandatory map processing step in certain regions of the world, see Reference [10]. Safety-relevant content has not to 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, it is important to prevent malicious changes to map content. As an HD map is an off-board sensor, cybersecurity is considered from creation to storage and distribution. This is discussed in greater detail in 5.2.4.

5.3.2.2.2 Global navigation satellite system (GNSS)

Absolute GNSS position contributes to the automated driving system safety. Consequently, not only accurate but also trustful absolute GNSS positions are necessary for location-based ODD determination. A time window of GNSS position validity with integrity is defined, as various levels of accuracy, integrity and availability will be in place while the automated driving system 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 important 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. It is important that these services implement fast processing, frequent updates and dedicated correction sets to support the best possible GNSS positioning algorithm.

A further aspect to cover is the assessment of new signals with respect to interferences in aeronautical radio navigation service (ARNS)/ radio navigation satellite system (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.

5.3.2.3 V2X

V2X can be considered a wireless environment sensing sensor, which allows vehicles to share information through communications channels. It can detect hidden threats and expand the range of the automated vehicle sensing.

V2X may provide valuable information to the automated driving system. However, safety and cybersecurity aspects are considered to ensure sufficient integrity and availability. Depending on the ODD, the automated driving system may also need to operate safe in conditions where V2X is not available.

There are many advanced applications such as vehicle platooning, remote driving, and cooperative automated valet parking system where V2X communication is essential. V2X has the potential to inform the ego-vehicle about the status of a traffic light or other vehicles, weather conditions, crashes on the road and construction on the road, especially during severe weather conditions and in complex traffic scenarios.

5.3.2.4 Sensor fusion

Sensor fusion is a process of combining multiple sensors and database data to improve information. It is a multi-level process, which can deal with the relationship and correlation between data and combine the data. Compared with using a single data source, it can get cheaper higher quality and more relevant information.

There are 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, are 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 is 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.

5.3.2.5 Interpretation and prediction

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

Interpreting the current environment enables the prediction of other 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 one or more of the following properties.

- Predict only a short time into the future. The likelihood of an accurate prediction is inversely 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 other 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 considers the compliance of other road users with traffic rules to a valid extent. For example, the automated vehicle at cross intersections with green traffic lights without stopping needs to rely on other road users to follow the rule of stopping at red lights. In addition to this, foreseeable non-compliant behaviour to traffic rules (e.g. pedestrians crossing red lights in urban areas) needs to be considered, supported by defensive drive planning.
- Situation prediction to further increase the likelihood of other road user prediction being correct. For example, the future behaviour of other road users when driving in a traffic jam differs greatly to their behaviour in flowing traffic.

The interpretation and prediction system understands not only the worst-case behaviour of other road users, but their worst-case reasonable behaviour. This allows the interpretation and prediction system to make reasonable and physically possible assumptions about other road users. The automated driving system makes a naturalistic assumption, just as humans do, about the reasonable behaviour of others. These assumptions are adaptable to local requirements so that they meet locally different “driving cultures”.

5.3.2.6 Localization

An automated driving system needs to 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 involved sensors, or sensor processing chains (or by real failures in either of these), or by multiple elements involved in the process.

Localization is implemented such that it 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). Therefore, a sound safety analysis of involved inputs with relevant failure modes, performance limitations, availabilities, and respective effects on the position estimation is 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.

5.3.2.7 ADS mode manager

The ADS mode manager fulfils 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). 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 gets 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 is targeted (see 5.2.6).

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 seatbelt status or vehicle operator attention. Based on the information from one or more monitors, the ADS mode manager decides whether to initiate an ODD functional adaptation 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 error types that may occur in the event of erroneous detection:

Table 5 — Determining the vehicle’s location

		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)

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 behaviour and consequences of automated driving operation outside of the ODD are, by definition, not safe enough as neither designed nor verified and validated for it. Therefore, it is important that appropriate safety and cybersecurity measures 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.

5.3.2.8 Egomotion

Egomotion describes the actual state of the vehicle 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 is fail-degraded to fulfil the capabilities. There are numerous ways to achieve this, so implementations will vary considerably.

5.3.2.9 Drive planning

Creating a driving policy that can drive in a collision-free manner without compromising comfort or traffic flow is the task of the element driving planning. A promising solution lies in defining formal rules, such as the examples of a theoretical approach[11] and hierarchical sets of rules[12]. There are many driving policy implementations available for driving planning, e.g. rule-based decision making, cost-based decision making[13], game theory[14] or Markov-decision-process-based decision making[15]. These theoretical rules are still applied to the complexities of real-world mixed traffic, and the resulting evaluation of the effect on traffic still takes place.

However, driving policies that cannot be proven to be safe by design should not be relied upon alone for decision making, and need to be complemented with a formal model that can help to ensure safety when probabilistic or theoretic approaches fail.

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

Explicit traffic rules

— The automated vehicle conforms to all applicable traffic rules within the ODD that it is operating in, taking regional differences in traffic rules into special consideration. Roads, signalling 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

— Confirmation to 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 road users with traffic rules are expected and dealt with defensively.

— Be cautious in areas where other 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 driving system is prepared for the possible sudden appearance of other road users.

— If it is possible to perform a legal and safety-assured manoeuvre to evade a potentially unsafe situation, then the automated driving system is doing so.

If it is not possible to evade an unsafe situation without prioritizing traffic rules, then it may be possible for the automated driving system to prioritize traffic rules while making a safety-assured manoeuvre.

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 is noted that safe driving is inherently based on assumptions about other road users (e.g. maximum deceleration). This is particularly important for occlusion scenarios. On the other hand, driving too defensively may confuse other 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, other 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 to escape a dangerous situation. If the automated driving system implements a proper response to a dangerous situation, at the danger threshold, then the automated driving system's own actions do not cause a collision and often avoid collisions caused by others who were not driving safely.

5.3.2.10 Traffic rules

Traffic rules are an important part of the behaviour of any vehicle on the road. All automated vehicles 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[16] states that the vehicle has not to drive faster than is safe for current conditions. However, "safe for current conditions" is not explicitly defined and so could be subject to interpretation. For these subjective traffic rules, an explicit definition of the expected behaviour (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) is highly desired.

5.3.2.11 Motion control

To implement the desired vehicle motion, precise actuator commands are derived from the given trajectory that is the output of the drive planning element (see 5.3.2.9). Therefore, a motion controller is necessary for generating lateral and longitudinal commands. It is important that the respective closed control loops are 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.

5.3.2.12 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 meet safety requirements. Depending on the item definition, different sets of manoeuvres can be derived that are still actuated and accordingly require different fail-degraded capabilities of the different primary actuator systems.

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

5.3.2.12.1 Steering system

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

To fulfil the capabilities, particularly being able to fail-degraded, there are now fail-degraded 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 differential braking.

5.3.2.12.2 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, any possible impact due to the automated driving functionality to the brake functions are considered, e.g. different usage profiles and possible increased availability.

5.3.2.12.3 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 based on the definitions of MRM.

5.3.2.13 Secondary actuators

The role of body control for automated driving is mainly to communicate planned driving manoeuvres 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 egomotion of the vehicle.

The following describes examples of potential impacts that are controlled by the automated driving system.

- External lights illuminate with the correct intensity to ensure adequate visibility to surrounding other road users and to provide a bright illumination for optical sensors (e.g. a camera sensor). The automated driving system ensures this operation when in automated driving mode, as the driver may be performing other tasks.

- Warning or indicator lights work correctly, as they may confuse other road users (e.g. by unintended activation or indication of wrong direction). Additional communication systems may be needed depending on the item definition.

- In order to ensure a clean field of view for the driver to take over the DDT, clean windshields (and rear mirrors) are desirable. Thus, cleaning, air conditioning and heating systems 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 document. Nonetheless, the impact automated driving has on these components are considered.

5.3.2.14 Human-machine interaction

Human-machine interaction is considered a crucial element for the safe operation of vehicles with SAE L3, L4 or L5 features. The human-machine interface 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, audible and haptic stimulus to support the user with relevant information, and it can offer different types of interfaces to receive input from the user. HMI is 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), a critically important and challenging goal particularly 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 time interval.

Regarding the different levels of automation (SAE L0-L5), 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 J3016:2018, as depicted by Figure 22).

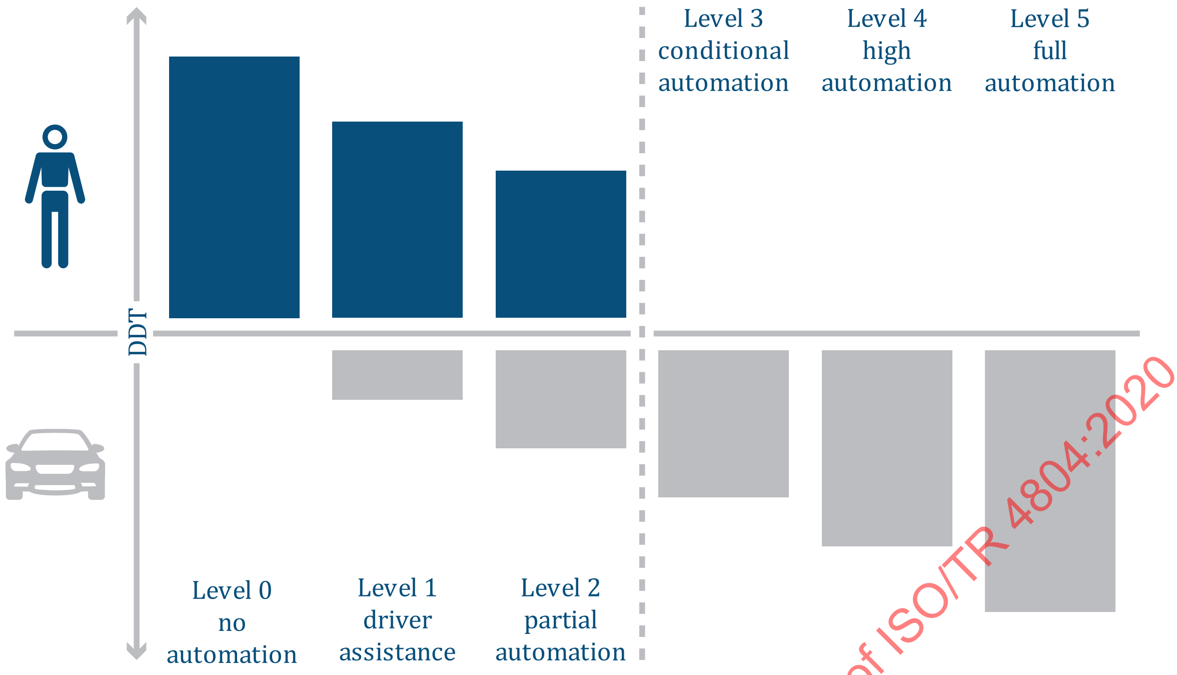

Figure 22 — Roles of the user and automated driving system by level of driving automation

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 driver experiences several systems under the similar environment but with different tasks. Mode confusion and an inappropriate transfer 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 Figure 22).

Thus, the HMI design in vehicles with different levels of automation is safety relevant and is carefully designed to allow the driver to discriminate between different modes. Therefore, the HMI design performs the following.

— Reliably detects intended driver behaviour 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.

— Points out the actual driving mode and the driver's responsibility in an unambiguous and easily understandable way.

— Promotes an appropriate trust in automation for the actual driving mode.

— Issues a request to intervene that is prominent, easily understandable, and gives the vehicle operator enough time to regain manual control and situational awareness.

A first approach for defining evaluation criteria was also released by the National Highway Safety Administration (NHTSA, 2017). Following NHTSA's guidance[17], HMI of an automated driving system has at minimum "to 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”.

Even though a driver using a L4 automated driving system has a further reduced responsibility of the driving task, such that the minimal risk manoeuvre 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 operating 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 other 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 and cargo 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.

5.3.2.15 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 other 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).

Table 6 — Example of available technologies and potential use of sensor data to measure or detect occupant attributes and to inform

HMI technologies 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			++					
NOTE A higher number of + indicates a greater potential of sensor data being an indicator for the occupant attribute.								

Like HMI, the sensor set to determine occupant attributes such as driver distraction or mode confusion is 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.

5.3.2.16 Vehicle state

Beneath the obvious driving task that will shift from the user's responsibility to the automated driving system, monitoring and maintenance duties are also in charge of the automated driving system while it is in automated driving mode. The vehicle state informs 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 has then to be able to carry out his mission.

The monitoring of the vehicle state includes:

- status of the energy storage system such as the fuel or the electric battery's state of charge,
- tire pressures,
- oil temperature and level,
- door status.

5.3.2.17 Monitors (all modes)

Monitors are essential for ensuring safe operation of the system by monitoring the state and behaviour of system elements in terms of performance, cybersecurity 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 nominal performance [FS_7 (5.3.1.7)],
- monitoring the availability of fail-degraded mode [FD_2 (5.3.1.9)],
- monitoring cybersecurity events.

If a monitor detects a lack of performance, one or more failures, or a cybersecurity event, this information is sent to the ADS Mode Manager, where appropriate measures are taken. In the case of failure impacting AD mode, an appropriate degradation concept inhibits AD mode reactivation until next vehicle switch off and vehicle proper operation has been verified either by self-diagnostic or maintenance.

5.3.2.18 Processing unit

It is important that the processing unit is developed in accordance with the ISO 26262 series and under consideration of the safety goals of the automated driving system. It offers enough integrity, performance, calculating power, availability, cybersecurity support, real-time support, automotive bus interfaces, high speed data interfaces, digital and analogue pins and low-power mode.

5.3.2.19 Power supply

The power supply provides 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 affects both power supplies at once. The power supply assures the operation of the automated driving system in all modes.

5.3.2.20 Communication network

The communication network provides 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.

5.3.3 Generic logical architecture

5.2.5.1 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. 5.3.2 introduced the logical building blocks (elements) of a system for implementing automated driving functionalities.

5.2.3 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 5.2.3, 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 document omits their implementation. Annex 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 5.3. The resulting logical architecture provides a complete view of the connection and signal flow among the different elements. The functional architecture of the intended functionality of an automated driving system is shown in Figure 23 and Figure 24.

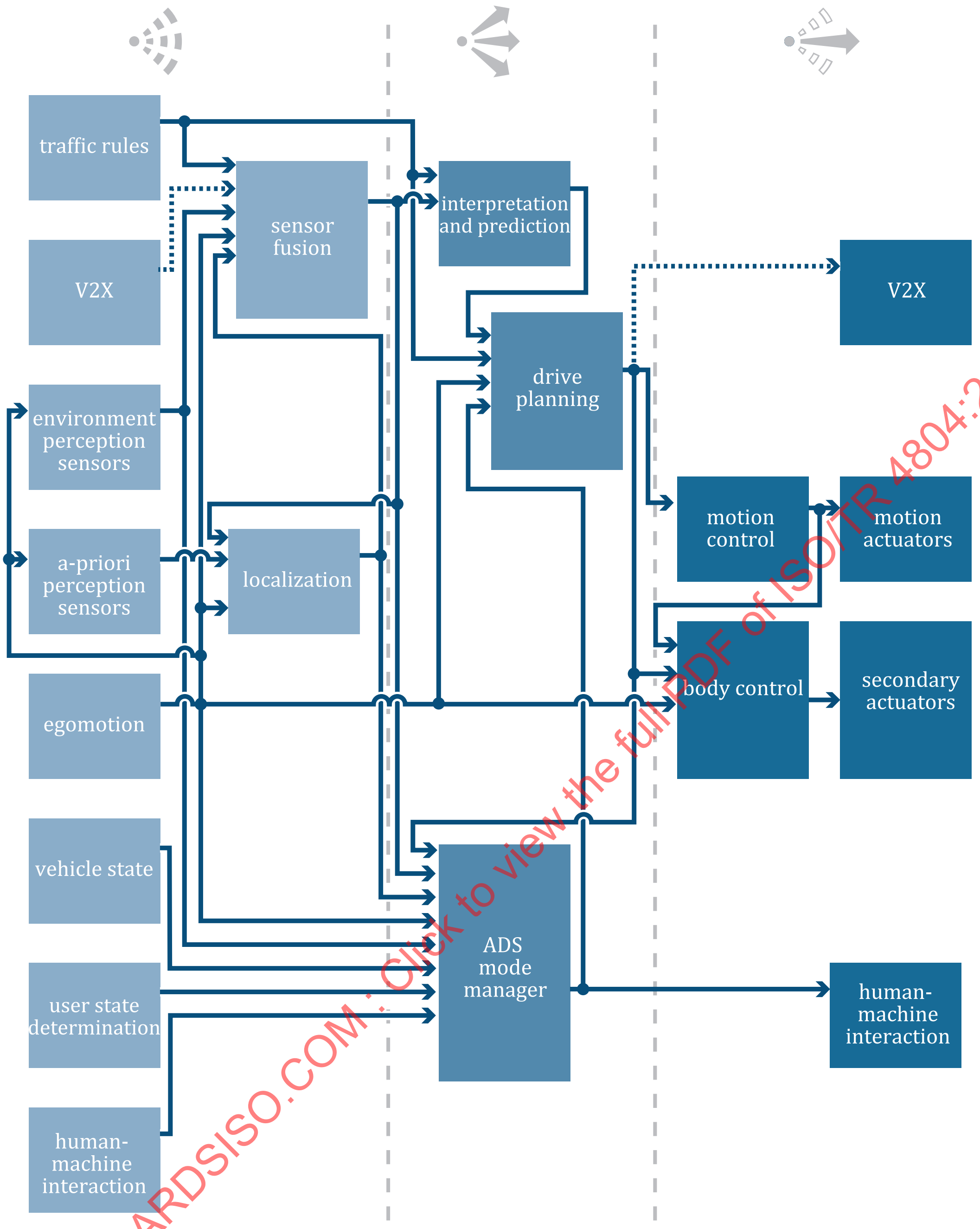

Figure 23 — Example architecture of the intended functionality

In addition to the elements described in 5.3, a set of additional requirements is also fulfilled in order to represent the current state-of-the-art. Each element needs 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 is observed and reported to the system monitor.

Redundant elements avoid dependent failures, so they are decoupled by design. Furthermore, redundant elements avoid common cause failures, so it is important to consider diversity during the design phase. Lastly, the system maintains at least fail-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.

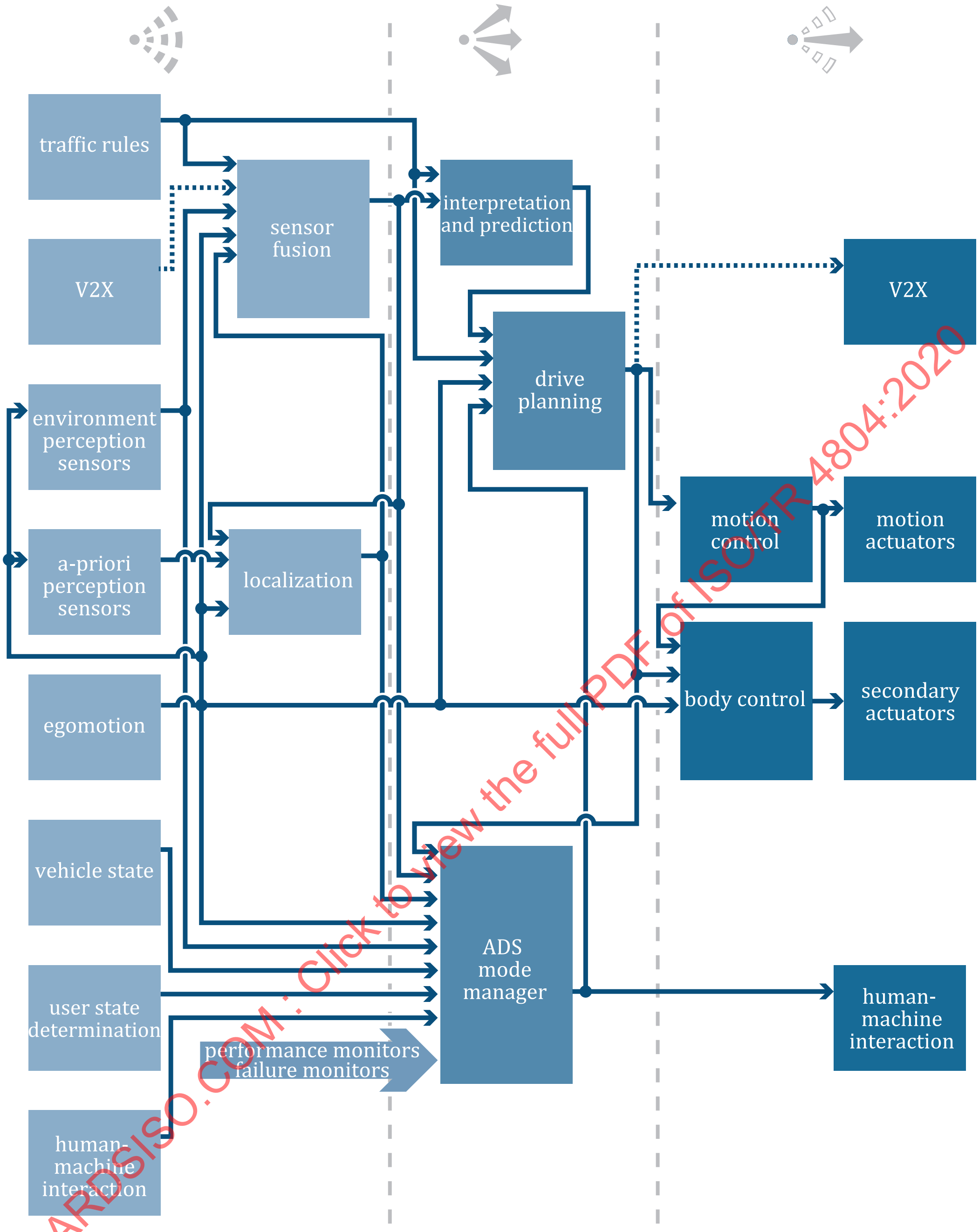

Figure 24 — Example architecture of the intended functionality, including monitors

Finally, all the capabilities are identified and assigned to the elements as the result of safety design and analysis. Figure 25 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 25, 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.

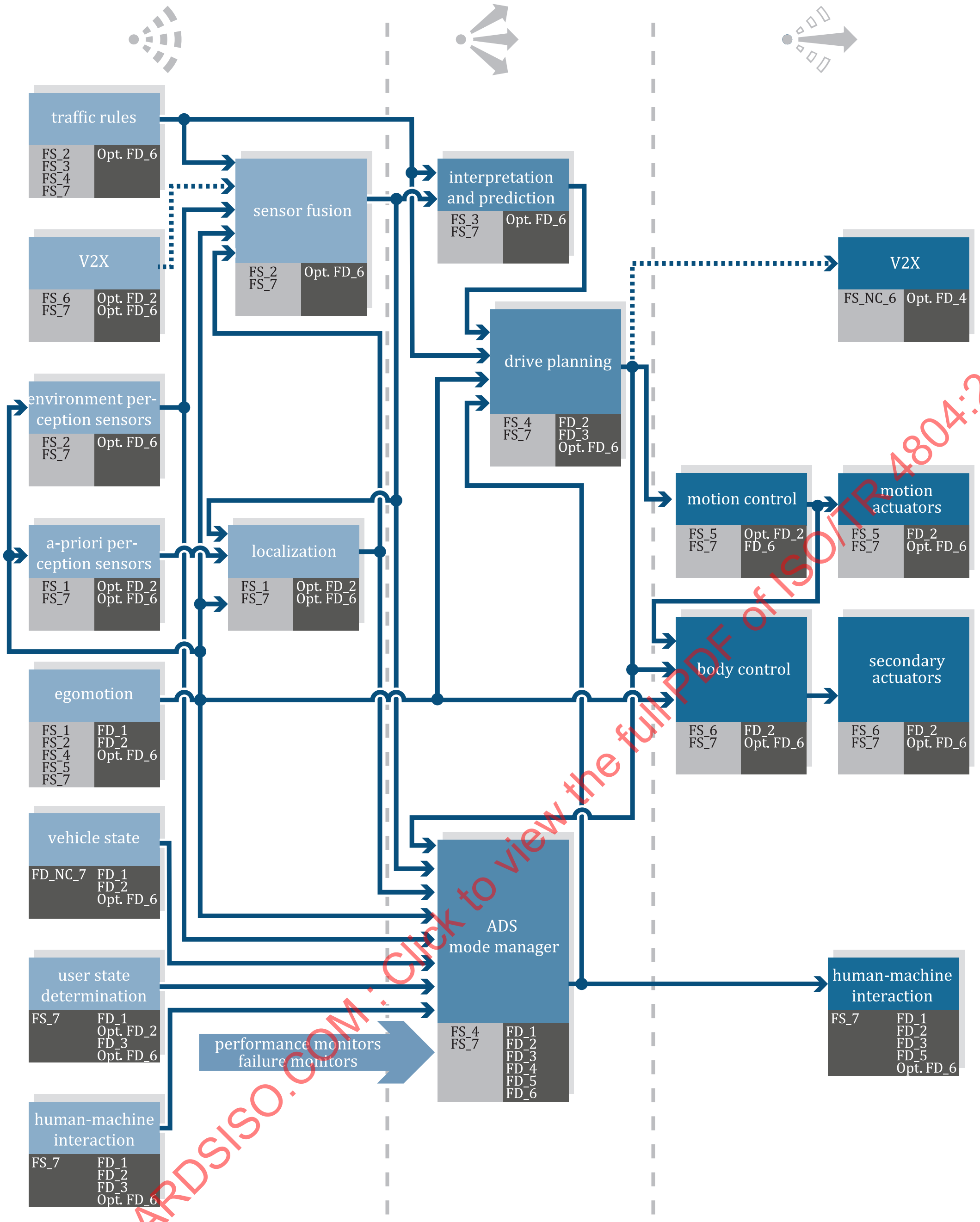

Figure 25 — Fail-safe and fail-degraded capabilities assigned to example architecture with monitors and fail-safe/fail-degraded elements

6 Verification and validation

6.1 General

This clause addresses the verification and validation of automated driving systems, including field monitoring and updates. 6.2 introduces the main steps and general approach and defines the scope of this clause. 6.3 lists the five key challenges that are unique to the verification and validation of L3 and higher automated driving systems. 6.4 proposes solutions for each of the challenges and includes a discussion of the various test platforms involved. 6.5 discusses the quantity and quality of real-world driving required, while 6.6 reviews the use of simulation for verification and validation. Finally, 6.7 focuses on specific verification and validation considerations for individual elements of an automated driving system. Although this document recognizes the possibility that validation testing may trigger functional design changes, most of this clause focuses on validating a stable system in a fixed ODD. However, 6.8 discusses post-deployment field operations, including the monitoring and management of configuration and ODD changes and updates.

6.2 The scope and main steps of verification and validation for automated driving systems

This clause focuses on verification and validation as it relates to the safety validation of SAE L3 and L4 automated driving systems. Thus, its scope excludes the verification and validation of product requirements not related to safety, such as comfort and efficiency. It also excludes standard verification and validation 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 the ISO 26262 series). This document assumes that SAE L3 and L4 automated driving systems will conform to these same standards, which cover many of the verification testing procedures. For example, the verification and validation methods and processes for cybersecurity are the same as for SAE L0 – L2, L3 and L4 automated driving systems and are stated in ISO/SAE 21434. Furthermore, it also excludes verification and validation activities in the production line (e.g. sensor adjustment, security in the production line), which can also follow standard SAE L0 – L2 procedures.

5.2 outlines the general concept of the systematic safety system development approach for an automated driving system. Figure 26 shows the main verification and validation activities used to evaluate the deployment and continued operation of SAE L3 or higher automated driving systems.

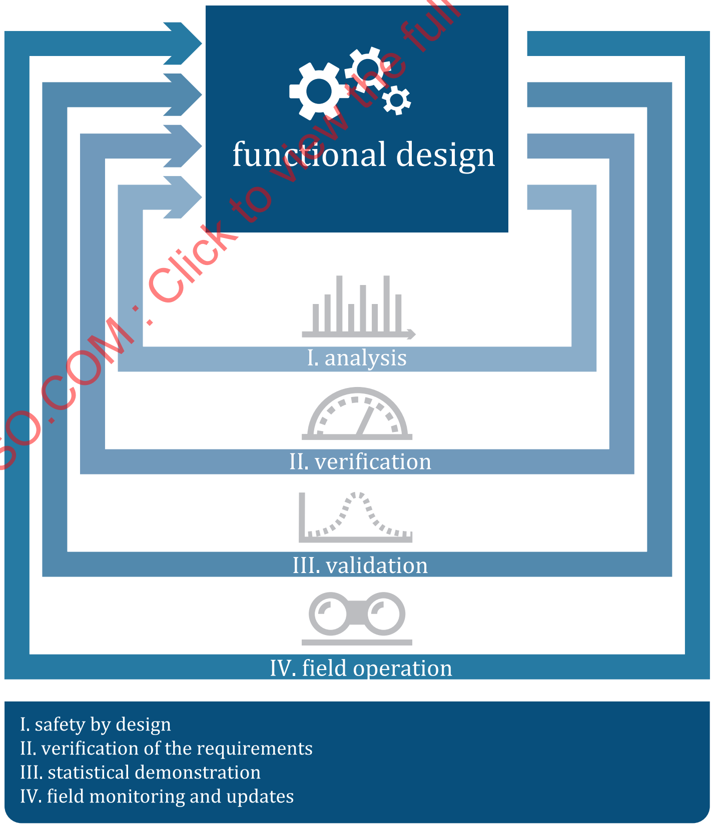
functional design
I. analysis
II. verification
III. validation
IV. field operation
I. safety by design
II. verification of the requirements
III. statistical demonstration
IV. field monitoring and updates

Figure 26 — Process of functional design in acc. with ISO/PAS 21448 SOTIF

As required by the ISO 26262 series, full system safety validation consists not only of testing, but also efforts such as quality audits of the development as well as verification and validation processes and tools, 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 verification and validation on the analysis side is the same as for SAE L0 – L2 systems (see the ISO 26262 series), but with increased complexity, this clause focuses on the testing side of verification and validation.

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 26, 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 reducing their presence.

While the principle of safety by design is fundamental to the safety approach, it serves only as a starting point for safety demonstration of automated driving systems, because of the existence of unknown unsafe scenarios that cannot be explicitly designed for or verified (see Clause 5). 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 4.4, validation aims to build the statistical argument as described in 6.4.4 to confirm the safety across both known and unknown scenarios with enough confidence. This represents the second step in the verification and validation 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 (e.g. 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 functional design. However, a valid statistical claim requires that the system under test be stable, and thus any iteration on the functional design will require repeating any validation tests that may have been affected by changes. Given the considerable extent of testing likely required for validation, this document recommends minimizing the degree of iteration on the functional design once validation has started.

The third step (loosely included in the broader verification and validation process) consists of post-deployment observation. This includes the field monitoring of the safety performance and cybersecurity 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 6.8).

6.3 Key challenges for verification and validation of SAE L3 and SAE L4 automated driving systems

This subclause decomposes the challenge of verification and validation for SAE L3 and L4 automated driving systems into five separate challenges, and 6.4 outlines viable solutions that cope with these challenges. The overall challenge is to increase confidence by using an acceptance criteria such as validation targets that the automated driving system has achieved a positive risk balance compared to the applicable traffic accident statistics (see 4.4.2). Therefore, possible driving scenarios occurring with a noticeable exposure are considered. As complete testing of every single scenario is neither appropriate nor technically possible, a viable method to statistically demonstrate system safety is defined further in 6.5.

6.3.1 Challenge 1: Statistical demonstration of avoidance of unreasonable risk 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 primarily focuses on the safety of actuators and electro-mechanical systems, and mostly considers only selected worst-case scenarios. In L3 and L4 automated driving 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 to be tested and implies to include statistical considerations in the overall safety argumentation, particularly to define ISO/PAS 21448 validation stop criteria.

6.3.2 Challenge 2: System safety with driver interaction (especially in takeover manoeuvres)

In the verification and validation of SAE L3 (and to a minor extent also SAE L4) automated driving systems, takeover scenarios also are assessed, as these impact the safety of automated driving systems. The driver as the DDT fallback ready driver maintains operating mode awareness and receives an unambiguous indication of any mode transitions. Likewise, the system supports 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 are analysed carefully and considered in the overall verification and validation and safety impact analysis.

6.3.3 Challenge 3: Consideration of scenarios currently not known

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 when 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 are considered in the overall validation method.

6.3.4 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 automated driving systems is expected to exceed the number of configurations for L2 systems. This topic is covered by verifying and validating each system configuration.

6.3.5 Challenge 5: Validation of (sub)systems that are based on machine learning

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

6.4 Verification and validation approach for automated driving systems

The following approach to verify and validate the safety of automated driving systems addresses the key challenges mentioned above. This document emphasizes that verification and validation processes include both testing and other verification activities to ensure rigorous development and system design implementation (see 6.2). Within the verification process, testing evaluates whether the specified requirement is met. Within the validation process, testing evaluates whether the automated driving system fulfils the intended use cases. However, complete test coverage is not achievable to fully validate the safety of SAE L3 and SAE L4 automated driving 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[18,19]. Only by fully answering these seven questions, the specific test case can be generated. The questions "when" and "who" are answered in the same way as for the development of L0–L2 systems. This subclause outlines the general approach to validating system safety, responding to the key challenges by explicitly describing:

1. why and how well: test goals including the scope, completion criteria and metrics (see 6.4.1);
2. how: test techniques (see 6.4.2);
3. where: test platforms or test environments (see 6.4.3);
4. what: test elements or object under test (OuT) (see 6.7).

Combining these characteristics establishes a specific test strategy (see 6.4.4 and 6.5) for L3 and L4 automated driving 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 6.5), the simulation environment and the number of tests (see 6.6). 6.7 details the examples for verifying and validating specific elements and capabilities.

6.4.1 Defining test goals and objectives (why and 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 are achieved. For every single test case, the test goal is quantified (e.g. test completion, stopping and resumption criteria) in accordance with the safety argumentation. Furthermore, objective metrics for test completion and test quality are defined for the entire test strategy. These are already used in L2 systems and so are not detailed here. For L3 and L4 automated driving systems, the metrics for the test coverage of large parameter distributions may be defined carefully. Some of the most important considerations when defining the test goals include the principles described in 4.4.3 and their related capabilities discussed in 5.2.5.

6.4.2 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 automated driving systems. Typically, these test techniques are classified either based on the testers' viewpoint ("box approach") or on how the test inputs are derived (ISO/IEC/IEEE 29119 series). All classification approaches have in common that, implicitly, the test techniques are classified by the knowledge about the OuT. In a very generic manner, one can distinguish between the very extreme test techniques, being completely undirected testing (i.e. random), completely directed testing (i.e. based on schemes), or a combination of these extremes (i.e. randomly searching along the schemes).

The following test design techniques for L3 and L4 automated driving systems possibly involve greater effort (e.g. depending on the ODD and the development example) than for L0–L2 systems:

- scheme-based test design techniques:
  - equivalence classes test design techniques (see 6.5.1);
  - boundary value test design techniques;
- search-based test design techniques:
  - design of experiments;

— mutation test design technique;

— reactive test design technique.

6.4.3 Test platforms (where)

Test platforms that have been adapted to the respective OuT and different test goals are used. 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. In Clause 3, the available test environments are defined including 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 7 classifies these stimuli and test items for each test environment.

Table 7 — Test platforms and test items

Test item Test platform	Test SW (code)	Target HW (ECU)	Vehicle	Driver	Driving environment
Simulation in the closed loop	virtual	virtual	virtual	virtual	virtual
	real			none	
Software reprocessing	virtual	virtual	none	none	virtual
	real				
Hardware in the closed loop	real	real	virtual	virtual	virtual
				none	
Hardware reprocessing	real	real	none	none	virtual
Driver in the loop	real	virtual	virtual	real	virtual
			real		
		none	none		real
Proving ground	real	real	real	real	real
				robot	
Open road	real	real	real	real	real

6.4.4 Test strategies in response to the key challenges

To address the key challenges discussed in 6.3, test goals, objects under test, test techniques and test platforms can be combined to define a viable test strategy for L3 and L4 automated driving systems.

6.4.4.1 Solution for challenge 1: Statistical demonstration of system safety and positive safety impact without driver/ operator interaction of ego-vehicle

To respond to this challenge, the automated driving system without a driver (sense – plan – act as discussed in 5.2.5) is covered. The following three strategies are combined to implement a general test approach for statistically demonstrating system safety.

a) Use of statistical grey box testing in real-world driving tests to cover the essential 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 to cover the variety of real-world driving scenarios to develop, as detailed in 6.5.

— Validation of the complete closed-loop system in read-world driving conditions.

— Identification of driving scenarios available in the ODD as a basis for addressing unknown unsafe test scenarios as described in 6.3.3.

b) Implementation of scenario-based testing for the complete driving system as well as for specific elements in dedicated test platforms using useful test techniques.

- Software/hardware reprocessing: validation of perception and sensor fusion (reprocessing of field measurements with new software releases).
- SiL: validation of trajectory-planning and control algorithms in simulations with sensor models covering a wide range of variations in the scenarios and scenery.
- 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) Ensuring field monitoring of the system over its lifetime.

- 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 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 6.5 discusses in greater detail.

The disadvantage of the dedicated testing of specific scenarios as outlined in 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 c) enhances the coverage of scenario-based testing for a sufficiently validated automated driving system.

6.4.4.2 Solution for challenge 2: Assessment of human driving performance (especially in takeover manoeuvres)

The safety of the human driver and automated driving 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 are passed sequentially. The following is an example of such a sequence and the steps involved:

1. simulation to find worst-case scenarios (in the sense of controllability by the driver in take over manoeuvres) for DiL using a basic driver model;
2. DiL testing of driver performance in combination with HMI;
3. real-vehicle testing on the proving ground with a closed-loop L3 or L4 automated driving system with a safety driver;
4. real-world testing on the proving ground with a closed-loop L3 or L4 automated driving system with expert drivers (no safety driver);
5. real-world testing on the proving ground with a closed-loop L3 or L4 automated driving 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).

Long term behaviour of the driver is an additional point to be considered in some steps. The analysis considers foreseeable misuse 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 behaviour of the customer. The safety goals determine the scope of the verification and validation of foreseeable misuse and abuse.

Customer case studies involving heterogeneous and large enough groups of participants is an additional point to be considered 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 automated driving systems. Vehicle-level and DiL tests are also necessary for validation, (so that safety-related testing does not jeopardize the safety of the test drivers).

6.4.4.3 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 is examined. Essentially, the test strategy rests on the following test platforms:

— simulation with bidirectional interaction of a fleet of automated vehicles (SAE L3 – L4) and multiple other, non-automated road users (including SAE L0 – L2), e.g. openPASS[20], including simulation of new combinations of scenario parameters as a result of the augmentation of existing ones;

— DiL and open-road testing to assess unknown scenarios resulting from the interaction of drivers in SAE L0 – L2 vehicles with automated driving systems.

In both test platforms, a broad range of possible system implementations is considered with regard to the behaviour of the simulated automated driving systems. 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.

6.4.4.4 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. for every change in one line of code, 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.

6.4.4.5 Solution for challenge 5: Validation of (sub)systems that are based on machine learning

For the verification and validation of safety-related machine learning algorithms, it is crucial to define a safe design process for these algorithms in addition to testing them. Annex B (particularly B.4 to B.7) examines machine learning in greater detail and describes the basic requirements for the verification and validation of machine learning algorithms.

Summary of the test strategy

In conclusion, a viable test strategy responds to the key challenges in the verification and validation 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, Table 8 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 Table 8, the example of the open-road test platform is used to test different test goals, focusing on the entire system. The entire system can only be tested in proving ground or open road as the sensor and actuator hardware (except the hardware controller) are not included in the SiL or HiL test platforms.

Table 8 — Summary of the test strategy

	SiL/SW reprocessing	HiL/HW reprocessing	DiL	Proving ground	Open road
Components	◆ 🚶 ◓ 🔒 🖥	◆ ◓ 🖥			
Sensor fusion, localization, perception	◆ ◓ 🔒 🖥	◆ ◓ 🖥		◆ ◓ 🖥	◆ 🚶 ◓ 🔒 🖥
System without sensors, prediction (drive planning)	◆ ◓ 🖥	◆ 🚶 ◓ 🖥	◆ ◓ 🖥		
Motion control, egomotion	◆ ◓ 🖥	◆ ◓ 🔒 🖥	◆ ◓ 🖥		
HMI, user state determination, ADS mode manager		◆ 🚶 🖥	◆ 🚶 ◓ 🖥		
Automated vehicle				◆ 🚶 ◓ 🔒 🖥	◆ 🚶 ◓ 🔒 🖥

NOTE Test goals:

◆	Technical aspects of SOTIF	🔒	Security and penetration testing
🚶	Human factor aspects of SOTIF	🖥	Validation on virtual test platforms
◓	Functional safety		

The time-consuming nature of some of the steps in the final safety validation testing process (see 6.5) prolongs the overall validation process, particularly as the steps cannot all be carried out in parallel (see 6.4.4.2). As discussed in 5.3.1, it is important that automated vehicles are aware of regulations specifying design, construction, performance and durability requirements. Documentation includes the verification and validation process. In addition, the necessary homologation process starts 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 may jointly define which kinds of tests are relevant for fulfilling the framework of homologation, e.g. depending on the countries, ODD and/or development example.

6.5 Quantity and quality of testing

As discussed in 6.3, one of the key challenges in validating automated driving systems is to statistically demonstrate avoidance of unreasonable risk and a positive risk balance based on the validation stop criteria (e.g. statistical validation target). 6.2 defines the main safety objective to be demonstrated. In a purely statistical, black box approach, i.e. using the automated driving system as a customer, "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", see Reference [21]. 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, see References [22]-[24].

To address this challenge, the test strategy proposed in 6.4.4 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, see References [1],[21]. 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 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 5.3 and testing these elements as discussed in 6.7.

— Combining different test platforms and test design techniques (e.g. stochastic variation in SiL) to increase test coverage as discussed in 6.4.4.

— Leveraging existing knowledge on relevant scenarios (e.g. from crash 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 considers the following.

— System design:

  — definition of the ODD (see also the development examples in 4.3), as the reference for the safety impact analysis and the test space are affected;

  — robustness of the elements (e.g. via redundancy of each element within systems like sensors, logics, and/or actuators), which require evidence that the elements are reliable (e.g. reasonable evidence of independence is presented for redundancy).

— Definition of the reference:

  — acceptance criteria (e.g. validation target, human driver in the ODD), similar to L3/L4;

  — confidence level to be achieved;

  — assumed statistical distribution.

Currently, this document roughly outlines the methods. For L3 automated driving systems, at least a similar amount of real-world test driving as for state-of-the-art L2 systems seems appropriate. Consequently, several million kilometres are driven in the real world for the development and validation of the motorway chauffeur system (see 4.3). This may then differ when the new methods outlined above are used. For other systems such as the traffic jam chauffeur system, urban chauffeur system and automated valet parking systems, other metrics adapted to the systems' ODD may be applied, and so the comparison of driven distance may not be reasonable. Validation is conducted continuously until the required confidence level is achieved.

6.5.1 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 safety of the system in the whole class. The initially defined equivalence classes are derived from the system design and analysis and are adapted in the verification and validation step, e.g. by the exposure, severity and controllability levels (as defined in the ISO 26262:2018 series) as a representative sample of operational scenarios to be grouped in equivalent classes. All influencing factors are modelled in the SiL/software reprocessing test platforms. Consequently, each equivalence class can be covered precisely during testing in simulation. When interpreting the simulation results, modelling inaccuracies and approximations are 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 are 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 can be split into six layers of a scenario, see Reference [25].

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 temporary construction sites

Layer 4: Road users and objects, including interactions based on manoeuvres

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, see Reference [26]. Consequently, scenarios and the scenario parameters at the different layers provide a holistic concept for defining equivalence classes. The inputs to get the influencing parameters are described in 6.6.2.

6.6 Simulation

In its broadest sense, simulation can help us to understand the possible behaviours 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 a 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 behaviour of the system of interest, for example. Models are abstractions from the physical reality and rely almost 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, see References [27],[28]. The level of accuracy required for a simulation model depends on the test goal and is established through a separate activity of validating simulation results (see 6.6.3).

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 limits of the system's capabilities. Examples of structured tests applied during development may 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 behaviour, 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 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. 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 modelling: the generation of novel scenarios in simulation requires realistic road user behaviour 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. proving ground);
- quantitative assessment of the system in specific driving scenarios.

6.6.1 Types of simulation

Numerous different types of simulation exist and can contribute towards different testing goals (see Clause 6). Regardless of the type of simulation, any simulation result is reproducible at a later point for traceability and maintenance purposes (see also 6.8). 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 follow the same approach as for SAE L0–L2 systems, see Reference [29]. 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 automated driving systems), functional safety testing evaluates 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 describing time delay of the bus communication 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 (including ViL and Driving Simulator) 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. 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 (depending on the necessary accuracy), compute-intensive sensor models. Once the perception performance is assessed, follow-up tests (e.g. of the behaviour 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 behaviour may differ from modelled human behaviour. 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.

6.6.2 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 other 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;
- unknown scenarios explored by AI or other optimization algorithms/parameter variation.

Simulation commonly relies on particular scenarios (conditions to test the system in) described in some data format (e.g. in References [30]-[32]). 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 are captured by simulation. More details can be found in Reference [22].

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 6.5.1) could be considered.

6.6.3 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 technical report 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 simulated corner cases increases to an acceptable degree by further validating the simulation models at the element level in conjunction with system design and analysis including robustness against safety-related scenarios. The final confidence statement about the automated driving system's safety accounts the remaining uncertainty about the validity of the simulation.

6.6.4 Further applications of 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 by sending notifications of virtual hazards to the backend, see Reference [23].

Another specific role for simulation may be to estimate the system's behaviour 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.

6.7 Verification and validation of elements

To address the key challenge 4 described in 6.3.4, and due to the high number of combinations of factors and their concrete values (see 6.5), this document suggests decomposing the system into subsystems and components to individually validate and verify these elements and to test only these parameters on higher system level, which have a safety relevant influence on the output of the elements. This subclause explains in greater detail the specific verification and validation of each element listed in 5.3.

Some subclauses below are more detailed, because the verification and validation of these elements changes more for L3 and L4 automated driving systems. Regarding the elements and capabilities listed in 5.3, this document focuses on elements that are verified and validated differently to L0–L2 systems. The elements with presumably no or minor verification and validation changes are:

Processing unit

Typically, other SoCs (System on Chips) and MCUs (microcontroller units) are used for L3 and L4 automated driving systems. However, the verification and validation methods are the same.

Power supply

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

Communication networks and body control

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

Egomotion (including odometry)

This is more accurate for L3 and L4 automated driving systems. However, verification and validation 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 6.7.5, in 6.7.7 for body control and secondary actuators and in the solution for the key challenge 2 described in 6.3.2.

6.7.1 A-priori information and perception (map)

Because reality is continuously changing, map verification and validation is a continuous process over the service life of systems using the map. Verification and validation of the map within this framework occurs at three levels:

1. the map as a subsystem in automated driving operations;
2. the map as a holistic reflection of reality;
3. specific map phases that might introduce errors (source, process and publication as identified in 5.3.2.2.1).

Automated driving system verification and validation is detailed in 6.4 and 6.5, which outlines the testing that is conducted to validate and verify the safety of the automated driving system in case of map/ perception mismatch. This testing focuses on scenarios where map data is critical or less controllable for operation and/or a mismatch is predictable. Therefore, this subclause will focus on the map as a reflection of reality and the phases of map creation regarding verification and validation.

From a safety perspective, the end-to-end verification and validation of the safety relevant map content (RMAs) is verified by comparing the data to a reference dataset updated overtime. The reference data is 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, is 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 has again to be employed for more quantitative assertions regarding correctness. Furthermore, system tests are 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 are applied within each phase of map creation (sourcing, processing and publication).

- Source data errors are 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 are 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 (failure mode and effects analysis), FTA (fault tree analysis) and other such techniques).
- The confirmation of the effectiveness of measures concerning publication errors are 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.

6.7.2 Localization (including GNSS)

This subclause 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 and longitudinal 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 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 verification and validation for the localization system with respect to functional safety and safety of the intended functionality, dedicated testing is in place to ensure that the vehicle's behaviour is safe on the roadway. For instance, functional safety testing includes fault injection on the IMU or GNSS system.

6.7.3 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 are applicable.

— (Re-)processing is 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 fulfilment 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 are separate from the dataset used in development (see also Annex B).

6.7.4 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 mainly manageable with only synthetic inputs using readily available sensor and/ or fusion models. This enables the use of verification and validation techniques that are mainly based on SiL of scenarios (as per 6.7). 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. As detailed in Reference [9], for standard scenarios with high exposure (e.g. cut in), a SAE L3 or higher system must behave at least as safe as a competent and careful driver to achieve a positive risk balance and to avoid unreasonable risk. Reference [9] it states that the behaviour of the competent and careful driver can be derived by data analysis. To assess the behaviour of a SAE L3 or higher system, an appropriate reference dataset is necessary. For example, a planned trajectory has not to 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 important for the verification and validation of prediction and planning.

6.7.5 Motion control

In this subsystem, the robustness against different variants of actuators, chassis, tires, aerodynamic, friction level are 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 6.6.3).

6.7.6 Monitor, ADS mode manager (including the vehicle state)

All the monitors (performance of all modes, 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 is 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.

6.7.7 Human machine interaction and user state monitor

In general, this subsystem can be tested in HiL, DiL and real-world environments. The technical performance of the subsystem can be tested with HiL to verify that it meets its specifications. The HMI's usability and user experience can be investigated in expert and user studies with DiL as well as real-world environments. Among else, transitions between different driving automation levels are evaluated for this purpose (if applicable). These tests verify and validate the awareness of most drivers concerning the SAE automated driving level the system is currently in and based on that which task the driver has.

To demonstrate that HMI meets all the requirements on usability and safety, studies with subjects unfamiliar with automated driving are 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 is 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 and controllability", see Reference [33].

6.8 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 may be focused focuses 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 subclauses describe the steps recommended for reasonably safe field operation of the automated driving system.

NOTE Lagging measures are applied to demonstrate that safety goals have been achieved.

6.8.1 Testing traceability

For successive software releases, a test plan is 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 26 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.

1. Distinguish whether the changes influence the safety of the system.

2. Analyse which safety relevant parts are changed or influenced.

The number of influencing factors to be tested at the entire system level is reduced as follows.

— Testing the influencing factors at the component/subsystem level and demonstrate robustness. Only factors influencing the entire system is tested at the entire system level. Due to the size of the test space and combinatorial explosion due to the number of test parameters, statistical evidence cannot be presented with a 100% certainty. As per ISO/PAS 21448:2019, Figure 9 and 12.2, a valid argument needs to be made so that the residual risk of undetected faults after testing is acceptably low. To minimize this effect, continuous field monitoring is highly important.

— Checking which test cases and scenarios are repeated and add additional tests.

— If the impact is not known in advance, the procedure is carried out for at least a sufficient number of 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 approval 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 cybersecurity 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 are 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 (fail-safe or fail-degraded), a robust configuration and change management process, system analysis, regression prevention and enforcement of corrective actions.

6.8.2 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 document that the ISO 26262 series 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 a region, city, system configuration, or even individual routes based upon the ODD. To achieve this, the full system and all its software and hardware components are described by using a unique identifier, and they have each to be individually authenticated. Each of these variants has then to 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 6.8.2 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 are 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 also matches with the version of the hardware and software that has been released.

6.8.3 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 6.2) from ISO/PAS 21448 is to define capability via a list of known operations. 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 involuntary customer participation.

6.8.4 Cybersecurity monitoring and updates

The previous discussion about cybersecurity in 5.2.4 discusses the processes and controls used to defend the systems and to find and fix vulnerabilities pro-actively. However, cybersecurity 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 driving systems 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 cybersecurity community (e.g. Auto-ISAC, national computer emergency response teams, conferences) remain important but may not be enough. Automated vehicles require a level of cybersecurity 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 6.8 also hold for cybersecurity incidents. With these capabilities, automated driving systems will adapt with the threat landscape. To respond effectively, the means to quickly update released systems can be used also in case of cybersecurity incidents. Furthermore, the lessons learned are 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 result in cybersecurity 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 cybersecurity discussed in this document, appropriate 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.

6.8.5 Continuous monitoring and corrective enforcement

Upon conclusion of the field monitoring and hardware/software change process, the new hardware/software will 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 cybersecurity-related change from a supplier or customer or a change initiated by a safety or cybersecurity impact observed in the field. For each of these triggering actions (see Figure 27), 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.

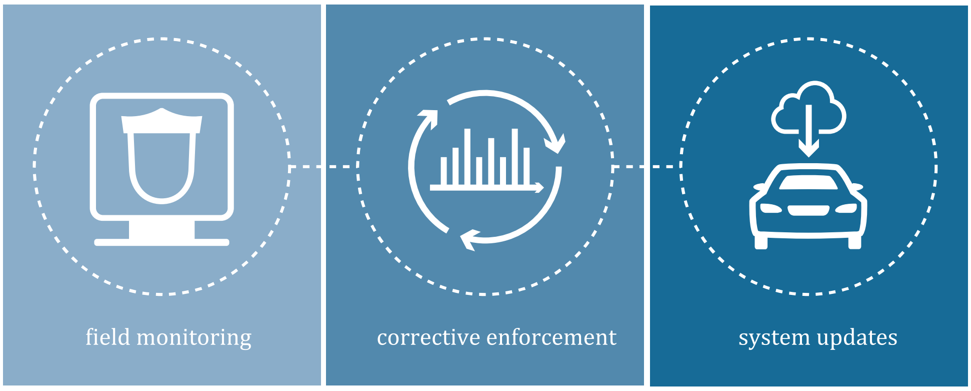

Figure 27 — Field operation process

Table 9 — Corrective enforcement

	Safety impact score			
	1	2	3	4
Description	No safety or cybersecurity impact with the planned function release	Potential safety or cybersecurity impact without recognizable change in the functionality or HMI	Potential safety or cybersecurity impact with recognizable change in the functionality or HMI	Immediate safety or cyber 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

It is technically possible to automatically update all elements for all risk levels in Table 9. In addition, a customer notification is considered for all levels in Table 9. In the event of changes to user 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 a 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, 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 28.

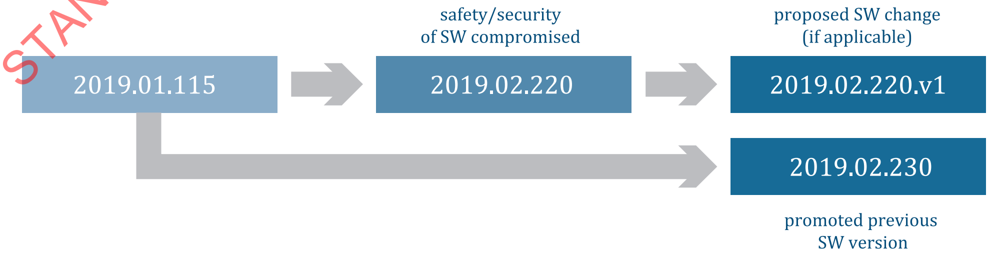

Figure 28 — Traceability and configuration management purposes

Annex A
(informative)

Development examples

A.1 General

This annex demonstrates how the examples from 4.3 may be implemented in accordance with the derived generic logical architecture from 5.3.3. First, the four examples and their MRCs and MRMs are defined using the IDs introduced in 5.2.5. 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.

The development examples are for explanatory purpose only and are on purpose not complete in terms of used elements and used capabilities.

A.2 SAE level 3 traffic jam chauffeur system (TJCS)

Table A.1 describes the traffic jam chauffeur systems.

Table A.1 — SAE level 3 traffic jam chauffeur system (TJCS)

Nominal function definition	SAE level 3 traffic jam chauffeur system as an option for vehicle customers; conventional driver receptive to ADS requested to intervene with driver's license, driving only on structurally separated roads, typically no pedestrians or cyclists, 60 km/h maximum, only with leading vehicles including motorcycles, no lane changing, no construction sites, only during daylight, only temperatures higher than freezing point.
Minimal risk condition	**TJCS_MRC_1.1** Driver has taken over control.
	TJCS_MRC_3.1 Vehicle is stopped in-lane. NOTE In this example, TJCS has no limited operation (MRC_2) minimal risk condition.
Minimal risk manoeuvre	**TJCS_DTO.1** The driving task is handed over to driver by issuing a takeover request and detecting takeover.
	TJCS_MRM_3.1 Speed is reduced until vehicle is stopped in-lane. Avoid collisions by braking.

A.3 SAE level 3 motorway chauffeur system (MCS)

Table A.2 describes the motorway chauffeur system (MCS).

Table A.2 — SAE level 3 motorway chauffeur system (MCS)

Nominal function definition	L3 motorway chauffeur system as an option for vehicle customers: conventional driver receptive to ADS requested to intervene 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.
Fail-degraded mode / minimal risk condition	**MCS_MRC_1.1** Driver has taken over control.
	MCS_MRC_2.1 Vehicle is driving in-lane with speed reduced to 80 km/h.
	MCS_MRC_3.1 Vehicle is stopped in-lane.
Minimal risk manoeuvre	**MCS_DTO.1** Issue takeover request to driver.
	MCS_MRM_2.1 Speed is reduced to 80 km/h. Continue longitudinal and lateral vehicle control (avoid collisions and keep lane).
	MCS_MRM_3.1 Speed is reduced until vehicle is stopped in-lane. Continue longitudinal and lateral vehicle control (avoid collisions and keep lane).
	MCS_RECOVERY_1 After MCS_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 A.1 depicts a possible functional architecture of the motorway chauffeur system. It is created from the generic architecture discussed in 5.3.3. Redundant instantiations of relevant elements are introduced to enable the availability of functional-degraded mode. The performance of the respective elements is adjusted to fulfil the capabilities in functional-degraded mode.

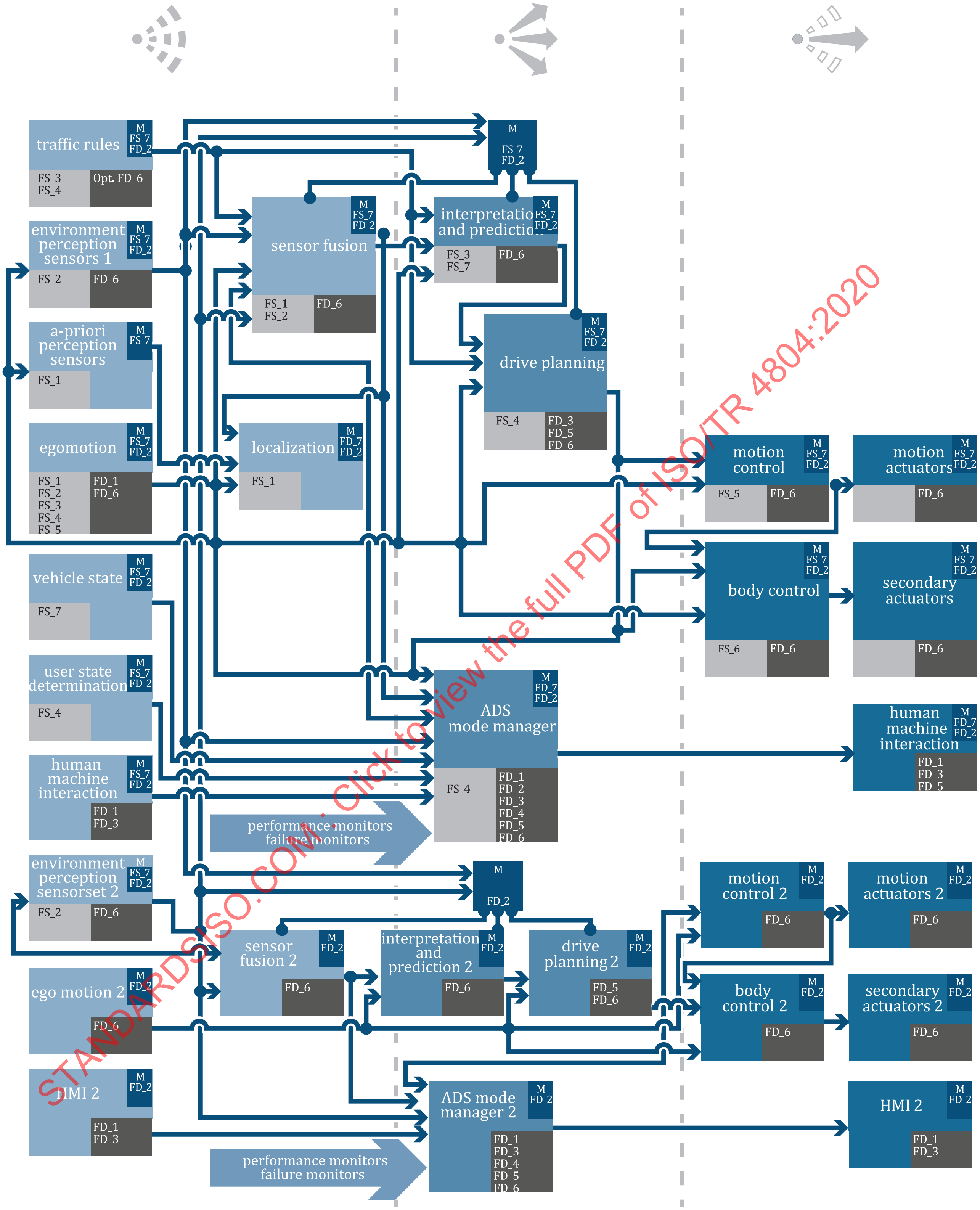

Figure A.1 — Generic Architecture

A.4 SAE level 4 urban chauffeur system (UCS)

Table A.3 describes SAE level 4 urban chauffeur system (UCS).

Table A.3 — SAE level 4 urban chauffeur system (UCS)

Nominal function definition	L4 urban chauffeur system in fleet operation in urban areas: non-vigilant conventional 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 indirect teleoperation if necessary.
Fail-degraded mode / minimal risk condition	**UCS_MRC_2.1** Vehicle is driving in-lane with speed reduced to 15 km/h.
	UCS_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).
	UCS_MRC_3.2 Vehicle is stopped in-lane. NOTE Because there is no driver present and no teleoperation intended in this example, UCS has no (MRC_1) minimal risk condition for driver takeover.
Minimal risk manoeuvre	**UCS_MRM_2.1** Speed is reduced to 15 km/h. Keep lane and avoid in-lane collisions by braking.
	UCS_MRM_3.1 Speed is reduced to 15 km/h. The vehicle is stopped at the next safe area (not on/in front of train tracks or in intersections). The vehicle informs the operator about the current state and position.
	UCS_MRM_3.2 Immediate stop in current location. No collision avoidance.

A.5 SAE level 4 automated valet parking systems (AVPS)

Table A.4 describes SAE level 4 automated valet parking systems (AVPS).

Table A.4 — SAE level 4 automated valet parking systems (AVPS)

Nominal function definition	L4 automated valet parking systems 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).
Fail-degraded mode / minimal risk condition	**AVPS_MRC_2.1** Vehicle is driving at crawling speed and avoids collisions.
	AVPS_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).
Minimal risk manoeuvre	**AVPS_MRM_2.1** Speed is reduced to crawling speed. The vehicle does not enter intersections or ramps.
	AVPS_MRM_3.1 Speed is reduced until vehicle is stopped in a safe location. The vehicle informs the remote operator (if available) or vehicle user.

A.6 Selection of the discussed elements

The following subclauses 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 and FS_7 determine if specified nominal performance is not achieved.
- Sensing elements requirements resulting from FS_2 perceive relevant 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 ODD functional adaptation.
- 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 ODD functional adaptation.
- Monitor element requirements resulting from FS_7 determine if specified nominal performance is not achieved and FD_2 detect when degradation is not available.

A.6.1 Sensing elements for FS_1 localization

A.6.1.1 Traffic jam chauffeur system 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.

A.6.1.2 Motorway chauffeur system L3

Localization determines the location of the vehicle on the map. Higher lateral than longitudinal localization precision is required. Localization aligns the perception capabilities with map matching needs. For example, landmarks that are included in the map attribute are 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.

A.6.1.3 Urban chauffeur system L4

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

A.6.1.4 Automated valet parking systems L4

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

A.6.2 Sensing elements for FS_2 perceive relevant objects

A.6.2.1 Traffic jam chauffeur system L3

Surrounding vehicles including motorcycles and obstacles principally in front of the ego-vehicle are detected with high degree of dependability. Lane markings are also relevant static objects. Sensors handle events that can happen inside ODD, and to be capable of detecting ODD violations.

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

A.6.2.2 Motorway chauffeur system L3

In addition to the traffic jam chauffeur system, the following relevant objects are detected with the highest possible dependability:

— vehicles including motorcycles at large distances in front of and behind the ego-vehicle, and vehicles including motorcycles 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 TJCS and MCS, the detection range of the sensor set to the front is increased and sensor sets added to the side and back.

A.6.2.3 Urban chauffeur system L4

Compared to the motorway chauffeur system, this scenario becomes much more complex and unstructured due to the:

— variation of objects and their degrees of freedom to move (particularly other road users);

— high probability of occlusion;

— traffic control elements;

— additional infrastructure elements and layout.

The sensor set capability is 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.

A.6.2.4 Automated valet parking systems L4

See the urban chauffeur system. In addition, the following challenges may apply:

1. objects on or close to ramps;
2. 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.

A.6.3 Interpretation and prediction in FS_3 predict future movements

A.6.3.1 Traffic jam chauffeur system L3

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

A.6.3.2 Motorway chauffeur system L3

The current situation is 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 behaviour of other road users when driving in a traffic jam differs vastly to their behaviour 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 behaviour 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 is 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 is interpreted, as this forms the basis for predicting future motion.

A.6.3.3 Urban chauffeur system L4

In this case, the interpretation and predict element takes other road users into account. For this development example, other road users may have a much more complex motion behaviour than for the traffic jam chauffeur system or motorway chauffeur system, 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 chauffeur system example. The interpretation and prediction model take this into account.

A.6.3.4 Automated valet parking systems L4

The challenges for this development example are comparable with those of the urban chauffeur system.

A.6.4 Acting elements in FS_5 execute driving plan and FD_6 perform ODD functional adaptation

A.6.4.1 Traffic jam chauffeur system L3

Nominal function

The nominal function consists of transforming the trajectory to a longitudinal and lateral vehicle movement up to 60 km/h and realizing a trajectory within given limits derived from lane, other objects and ego-vehicle width with the given and nominal performing actuators.

Minimal risk manoeuvre

TJCS_MRM_3.1: immediately stopping the vehicle with fixed deceleration, lateral vehicle movement based on last valid trajectory.

A.6.4.2 Motorway chauffeur system L3

Nominal function

The nominal function consists of transforming the trajectory to a longitudinal and lateral vehicle movement up to 130 km/h and realizing a trajectory within given limits derived from lane, other objects and ego-vehicle width with the given and normal performing actuators.

Minimal risk manoeuvre

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

HP_MRM_3.1: realizing 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.

A.6.4.3 Urban chauffeur system L4

Nominal function

The nominal function consists of transforming the trajectory to a longitudinal and lateral vehicle movement up to 70 km/h and realizing a trajectory within given limits derived from lane, safety distances to other objects, other road users and ego-vehicle width with the given and nominal performing actuators.

Minimal risk manoeuvre

UCS_MRM_2.1: transforming the trajectory to a longitudinal and lateral vehicle movement up to 15 km/h.

UCS_MRM_2.3: realizing 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.

A.6.4.4 Automated valet parking systems L4

Nominal function

The nominal function consists of transforming the trajectory to a longitudinal and lateral vehicle movement up to 60 km/h and realizing a trajectory within given limits derived from lane, other objects and ego-vehicle width with the given and nominal performing actuators.

Minimal risk manoeuvre

The minimal risk manoeuvre consists of realizing the last known valid trajectory with the available actuators and transitioning into degraded mode. Based on its definition, this means that the vehicle will stop in its lane.

AVPS_MRM_3.1: stopping in a safe location and inform the remote operator (if available) or vehicle user.

A.6.5 ADS mode manager in FS_7 detect nominal performance and FD_4 react to insufficient performance

A.6.5.1 Traffic jam chauffeur system 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 fail-degraded mode based on the outputs of the monitor.

Minimal risk manoeuvre

TJCS_DTO.1 and TJCS_MRM_3.1: Deactivating as soon driver has control or the vehicle is stopped.

A.6.5.2 Motorway chauffeur system L3

Changing from motorway chauffer system L3 to the traffic jam chauffeur system L3 is tied to the ODD specifics. In this case, it can be expected that the travelling speed of the vehicle is less than 130 km/h.

Minimal risk manoeuvre

The minimal risk manoeuvre consists of selecting the appropriate MRM. For example, reduced sensor performance due to reduced visibility leads to MCS_MRM_2.1. Reaching the end of the ODD leads to MCS_DTO.1 or MCS_MRM_3.1 to ensure either a takeover by the user or a safe stop before the ODD limitations are violated.

A.6.5.3 Urban chauffeur system L4

This could mean that the vehicle is inside a geofence, for example. It also checks the deactivation conditions to ensure that the vehicle has reached a fail-safe state. Additional states and transitions are introduced for the option of operating the vehicle by a remote operator. The ADS mode manager switches to fail-degraded mode based on the outputs of the monitor.

Minimal risk manoeuvre

The minimal risk manoeuvre consists of selecting the appropriate MRM. For example, reduced localization sensor performance leads to UCS_MRM_2.2. Cases where driving cannot continue due to a blocked lane or a solid lane marking lead to UCS_MRM_2.2. Once a rear-end collision has been detected, it is important to switch to UCS_MRM_2.3 and additionally to secure the vehicle as soon as a full vehicle stop has been reached.

A.6.5.4 Automated valet parking systems 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 fail-degraded mode based on the outputs of the monitor.

Minimal risk manoeuvre

Ability of a product to deliver a function, feature or service mode based on the failure, i.e. switching to an appropriate fail-degraded mode based on the failure.

A.6.6 User state determination in FD_1 ensure controllability for driver

A.6.6.1 Traffic jam chauffeur system L3

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

A.6.6.2 Motorway chauffeur system L3

Potentially no increase to the traffic jam chauffeur system.

A.6.6.3 Urban chauffeur system 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;

— tele operator: monitoring the tele operator is not necessary, because they are considered to be a trained expert.

A.6.6.4 Automated valet parking systems 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;

— tele operator: monitoring the tele operator is not required, because they are considered to be a trained expert.

A.6.7 HMI in FD_1 ensure controllability for operator and FD_6 perform ODD functional adaptation

A.6.7.1 Traffic jam chauffeur system 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 chauffeur system or to accept a takeover request.

A.6.7.2 Motorway chauffeur system L3

No additional requirements.

A.6.7.3 Urban chauffeur system L4

The user can introduce a vehicle stop by requesting from the navigation system. This could also lead to an immediate stop.

A.6.7.4 Automated valet parking systems L4

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

A.6.8 Monitors in FS_7 and FD_2

The monitors assess, e.g. 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.

A.6.8.1 Traffic jam chauffeur system L3

The monitor assesses the performance of the sensors, actuators, and the power supply necessary for the safety of the ADS.

A.6.8.2 Motorway chauffeur system L3

The monitor also assesses 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 is monitored.

A.6.8.3 Urban chauffeur system L4

In this case, there is the additional monitoring of the energy resources to ensure a longer operating period. The user state determination may no longer be monitored.

A.6.8.4 Automated valet parking systems L4

In this case, there is the additional monitoring of the energy resources to ensure a longer operating period. The user state determination may no longer be monitored.

Annex B
(informative)

Using deep neural networks to implement safety-related elements for automated driving systems

B.1 General

The aim of this annex 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 annex 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.

B.2 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, see Reference [34]. Consequently, a strict assessment of the development process for and the performance of machine learning algorithms responsible for executing safety-related tasks of automated driving systems may be done as outlined in the following.

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: the ISO 26262 series and ISO/PAS 21448. However, the safety standards available within the automotive and any other industry have been defined without explicitly introducing dedicated measures for qualifying machine learning algorithms for safety-related applications, with the exception of ISO/PAS 21448:2019, Annex G and the upcoming ISO 21448[2]that have and will have machine learning related contents that are relevant to the topics discussed in this Annex. In particular, machine learning models learn parameters from data and therefore they cannot be evaluated by only looking at source code. This leads to a challenging issue today for automated driving system manufacturers and suppliers who are incorporating machine learning for automated driving.

Deep neural networks (DNN) are inspired by how the human brain works. A neural network is made up of many simple processing nodes where inputs are multiplied according to weights for each incoming connection and added with biases then fed into activation functions which compute the output values. In DNNs, nodes are parameterized by those weights and biases, to be learned from labelled training dataset. Since the number of the parameters is large, the training dataset is also large. The quality of the training dataset has direct impacts on the quality of the trained model. Another factor that impacts the performance of DNNs is the choice of the DNN architecture.

Training is the process for machine learning algorithms to build its models from a dataset, which can either be supervised or unsupervised, see Reference [35]. In supervised learning, the machine learning model is presented an input and the desired output during training. This means that the data is already labelled with the correct answer. Unsupervised learning algorithms are trained using a dataset that does not have any labelling 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

[2] Under preparation. Stage at the time of publication: ISO/DIS 21448:2020.

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 unlabelled, see Reference [36]. This machine learning model learns via reward or penalty feedback received based on the interaction within the environment.

This annex focuses only on supervised offline learning for DNNs[37], i.e. learning at development phase. This approach is 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 Reference [38]). The annex is structured according to Figure B.1.

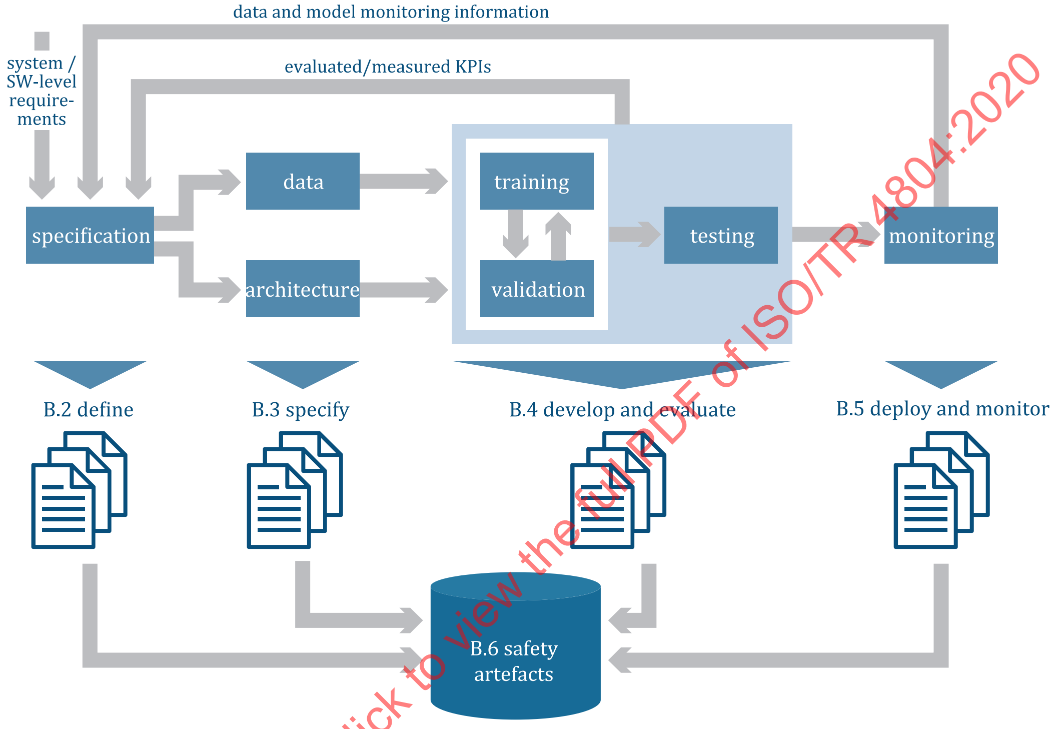

Figure B.1 — DNN lifecycle

For greater clarity for developers and assessors, this document defines a modular-based system architecture 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 5.3.2). 3D object detection[39] based on a DNN is used as an example in order to easily grasp the concepts described in this annex. 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 B.2).

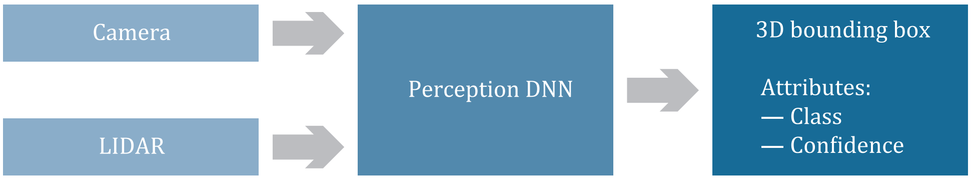

Figure B.2 — Camera and LIDAR-based DNN object detection

Define, specify, develop and evaluate, and deploy and monitor are the lifecycle steps of DNNs and provide the safety artefacts. Their verification provides evidence that support the safety case (see Figure B.1). These steps and safety artefacts are discussed in greater detail in the subclauses below.

B.3 Define (what and why)

Using a modular design approach, the requirements of the software components that form the automated driving system are defined (inputs, outputs, technical safety requirements, software safety requirements, functional requirements, etc.). When specifying a DNN, a non-exhaustive list of example considerations can include the following:

- software aspects:
  - the structure of the DNN,
  - how weights are stored,
  - how weights are accessed,
  - timing aspects,
- training aspects:
  - the ODD,
  - the outputs (in the form of probabilistic outputs),
  - the dataset attributes critical for the DNN objective and measurable metrics.

Updates to the requirements and artefacts concerning the data-driven development process are anticipated.

Table B.1 — Possible specification characteristics to consider when defining a DNN

ODD	Safety-related	Metrics	Hardware
Weather conditions	Characteristics of the target classes	Robustness measure	Memory footprint
Geographic domain	Labelling classes	(Class-wise) performance measures	Latency (timing)
Background scene	Labelling quality	Confidence quality	Optimization
Dynamic properties of the scene	Data coverage	Reproducibility	Sensor calibration
		Plausibility	
		Explainability	

Regarding the use case of 3D object detection, the characteristics outlined in Table B.1 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, colour)?
- 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 is taken when mapping the safety goals and requirements to measurable and reachable key performance indicators (KPIs) to train DNNs for automated driving systems, and when evaluating the safety of the resulting DNN models.

For example, such KPIs cover:

- adversarial robustness of a DNN against adversarial attacks, i.e., mathematically optimized perturbations;
- corruption robustness of a DNN against naturally occurring effects, e.g. distributional shifts, edge cases;
- explainability of a DNN;
- plausibility of the resulting DNN-based software component;
- latency of the resulting DNN-based software component;
- ability of the DNN to generalize beyond the training data to data expected to be encountered within the ODD.

Typical artefacts are from this phase of development are:

- dataset specification (specification of the global dataset attributes);
- labelling specification (specification of the classes, boundaries, labelling guidelines);
- DNN requirements specification (specification of the ODD, functional objective requirements, technical safety requirements, etc.);
- KPI specification (robustness against perturbation, etc.).

B.4 Specify (how)

B.4.1 Defining and selecting the data

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

A DNN-based component is developed using three disjoint datasets: training, validation and testing (Figure B.1). 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 are designed to sufficiently cover the input domain. The datasets are designed to 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) includes enough heterogeneous examples of pedestrians in such challenging environments. Furthermore, the datasets can 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 (see Figure B.3).

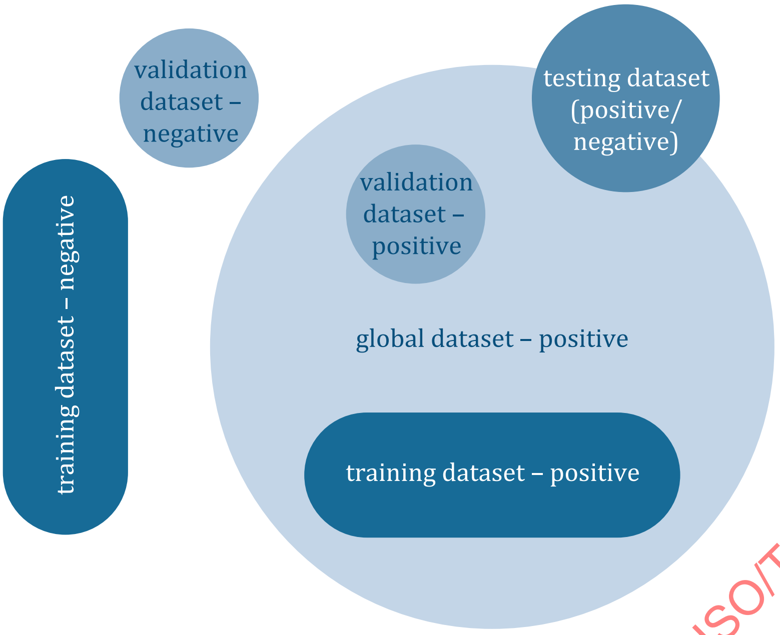
validation dataset – negative
testing dataset (positive/ negative)
validation dataset – positive
training dataset – negative
global dataset – positive
training dataset – positive

Figure B.3 — Dataset configurations representing the global dataset

It is recommended to have a dataset which, by projecting the dataset to certain criterion such as colour of the vehicle, can be partitioned into roughly equal subsets. For example, for 3D object detection algorithm, the objective may be to detect and classify pedestrians, vehicles and bicycles. The statistical distribution is considered for each class 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 B.2).

Table B.2 — Example of attributes for 3D object detection

	Class		
	Pedestrian	Vehicle	Bicycle
Class attributes	— Size — Position — Pose — Clothing — ...	— Size — Position — Colour — Type of vehicle — ...	— Number of wheels — Orientation — Human presence — Attachments (trailer) — ...
Environmental attributes	— Background colours (trees, buildings, ground cover) — Occlusions — Weather — Lighting — Adversarial perturbations — ...		

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

— coverage;

- relevance;
- balance of negative and positive examples.

The dataset is continuously improved as previously unseen scenes are discovered in the unknown unsafe space. Over time, the characteristics of the data might change in the operating environment. When this occurs, the dataset can be updated to reflect the new characteristic, and the DNN model can be retrained using the updated dataset. 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 includes 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, typically, real data dominates to ensure safety.

B.4.1.1 Dataset labelling

An expert carefully defines the labelling specification to ensure that the labelling characteristics are defined sufficiently and can efficiently relate to the target task.

There are many approaches for labelling. 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 labelling 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 are in place to ensure data is properly labelled, regardless of the labelling method used, to ensure error injection caused by the labelling process is minimized. Typical labelling errors in the case of 3D object detection are:

- incorrect classification of objects;
- objects that are overlooked by human annotators;
- wrongly positioned bounding boxes;
- bounding boxes with the wrong size or pose;
- split bounding boxes due to partial occlusion.

Following aspects can be considered when choosing the set of labelling 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 labelling classes (e.g. does a Segway[3] belong to the class “person” or “cyclist”?). Compliance with these limits is demonstrated by targeted quality assurance.

The following artefacts are expected from this phase of development:

- refined labelling specification;
- refined dataset specification;
- labelling quality report;
- labelled dataset (representative global dataset including the data splits for training, validation and testing datasets);

[3] Segway is an example of a suitable product available commercially. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO of this product.

- — dataset KPI report (measurables such as dataset coverage, algorithm robustness, dataset quality, etc.);
- — list of tools used for training and validation (dataset creation, labelling, KPI metrics, etc.).

B.4.2 Architecture design for DNNs

An architecture design is developed based on the requirements of the characteristics described in the Define phase in B.3. This can be achieved by considering different architectural design patterns at the software architecture and DNN architecture levels (see Figure B.4), which are described below.

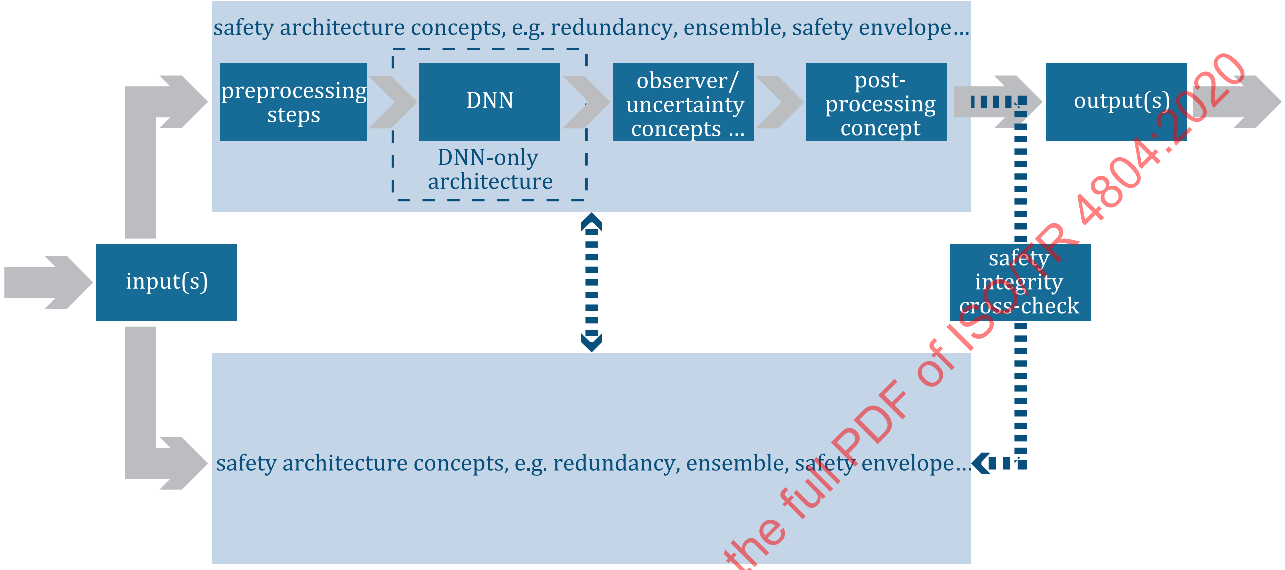
safety architecture concepts, e.g. redundancy, ensemble, safety envelope...
preprocessing steps
DNN
DNN-only architecture
observer/ uncertainty concepts ...
post-processing concept
output(s)
input(s)
safety integrity cross-check
safety architecture concepts, e.g. redundancy, ensemble, safety envelope...

Figure B.4 — Architecture design for DNNs

B.4.2.1 DNN-based software architecture level

The design intent of a software architecture incorporating a DNN is that the resulting software is able to detect and mitigate the risk associated with unexpected 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 improving the model, e.g. via more training data, changed training strategies, etc. Aleatoric uncertainty is inherent to nature itself and could be considered when defining a driving function to increase safety. In contrast, epistemic uncertainty could be considered in data selection and the choice of training strategies. Existing literature adopts different approaches for measuring uncertainty at inference, e.g. MC dropout, see References [40],[41].
- — Observers: observers are additional software components working in parallel to the DNN itself to supervise its behaviour with reference to safety. The concept of observers is strictly related to the test and scenarios for which the evaluator mechanism is designed, and to the concept of coverage to relate the test objectives to the requirements. Such observer components could include:
  - — plausibility check methods that check the model output for consistency (e.g. checking for implausible positions, sizes, dynamic properties of detected and tracked objects);

    NOTE It is possible that plausibility checks do not carefully consider certain corner cases.

- input observation methods that check the inputs or the features thereof are statistically close to the training dataset;
- saliency maps that check for plausible sensitivity of the DNN toward regions in the input;
- evaluation methods that evaluate the risk caused by false negative or false positive.

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. plausibility check) mechanisms for abnormal neural behaviour or abnormal input data can be used to identify situations of possible failure.
- Mechanisms such as heatmap can be employed to steer the effect of a particularly critical input on the model's output.
- Safety envelope concept, with a typical example being the doer/checker approach, see Reference [27]. 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. It is important that the boundaries between the doer and checker transition are specified precisely to avoid any major system failure.

The overall DNN-based software architecture may also include input pre-processing steps such as resizing, resampling, etc. Similarly, post-processing steps may be required to ensure the DNN output is compatible with the output interface of the subsequent subsystem.

B.4.2.2 DNN-only architecture level

This architecture level focuses on the DNN itself. The type and combination of DNN layers in the architecture are 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 is 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 artefacts 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.

B.5 Develop and evaluate

Having specified 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 is trained, optimized, evaluated and integrated into the overall automated driving system before a final safety argument can be carried out. This subclause covers possible steps for developing and evaluating the DNN before it goes into the integration.

The parameterization of a DNN model using labelled 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 considered when designing the loss function, and so the trained model is tested against the safety requirements. Possible solutions to ensure safe functioning could include adding additional terms in the loss function representing measurable safety requirements, i.e., train using safety-related fitting goals. Loss functions that maximally separate feature representations may increase robustness, but there may be trade-offs with other metrics such as traditional model performance metrics. Such trade-offs can be systematically evaluated during model development.

In addition to this, the training process includes the specification of hyperparameters:

- 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 above specified parameter choices are tracked, as they influence the resulting functional and non-functional properties of the DNN.

The training and validation process is iterative as depicted in Figure B.1. A trained DNN's knowledge is limited to the examples it has seen during training. The validation and testing datasets are designed to adequately cover the space of possible inputs to obtain a better understanding of the actual performance of the model. Moreover, they are designed to 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 data that causes the failure is 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. Failure cases are further analysed.

Once DNN failure cases have been identified, the model is retrained and potentially the datasets are 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.

Typically, the DNN is initially tested 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 is independent from validation and training data. It is used to provide the statistical evidence for the quality of the DNN. Therefore, it is not used in optimization cycles. Otherwise, the network might be overfitted to the specific test set and the role of the test set as an independent measure would be lost. Furthermore, 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, additional data is collected for validation and training in case failures are identified.