



BSI Standards Publication

Road vehicles — Data communication between sensors and data fusion unit for automated driving functions — Logical interface

National foreword

This British Standard is the UK implementation of ISO 23150:2023. It supersedes BS ISO 23150:2021, which is withdrawn.

The UK participation in its preparation was entrusted to Technical Committee AUE/16, Data Communication (Road Vehicles).

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2023
Published by BSI Standards Limited 2023

ISBN 978 0 539 18483 9

ICS 43.040.15

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 30 June 2023.

Amendments/corrigenda issued since publication

Date	Text affected

Road vehicles — Data communication between sensors and data fusion unit for automated driving functions — Logical interface

*Véhicules routiers — Communication de données entre capteurs et
unité de fusion de données pour les fonctions de conduite automatisée
— Interface logique*





COPYRIGHT PROTECTED DOCUMENT

© ISO 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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.....	vi
Introduction.....	viii
1 Scope	1
2 Normative references	1
3 Terms and definitions.....	1
3.1 Architectural components.....	1
3.2 Terms for logical interface layers.....	2
3.3 Structure terms	3
3.4 Measurement terms.....	3
3.5 Requirement level terms	6
3.6 Road user relevant entity types	7
3.7 Axis and coordinate system terms	9
4 Abbreviated terms	15
5 Structure of the interface description.....	16
5.1 General	16
5.2 Signal	17
5.3 Interface.....	17
5.4 Specific signal grouping.....	18
5.5 Profile	19
6 Logical interface from a sensor as well as a sensor cluster to a fusion unit.....	19
6.1 General	19
6.2 Generic interface header	22
6.3 Generic interface entity.....	22
6.4 Profile: Uniqueness of interface versioning.....	23
7 Object level.....	23
7.1 General	23
7.2 Generic object level interface.....	24
7.2.1 Generic object level header	24
7.2.2 Generic object level entity	25
7.3 Potentially moving object interface.....	25
7.3.1 Potentially moving object header.....	34
7.3.2 Potentially moving object entity	35
7.3.3 Profile: Motion type	36
7.3.4 Profile: Motion state vector	37
7.4 Road object interface	38
7.4.1 Road object header	49
7.4.2 Road object entity.....	51
7.4.3 Profile: Colour model for RDOI.....	53
7.5 Static object interface.....	53
7.5.1 Static object header	81
7.5.2 Static object entity.....	83
7.5.3 Profile: Colour model for SOI.....	91
7.5.4 Profile: Detection references for 3D detections	91
7.6 Free space area object interface	94
7.6.1 Free space area object header	99
7.6.2 Free space area object entity.....	100

8	Feature level	101
8.1	General.....	101
8.2	Generic sensor cluster feature interface	102
8.2.1	Generic sensor cluster feature header	102
8.2.2	Generic sensor cluster feature entity	103
8.3	Camera feature interface.....	103
8.3.1	Camera feature header	107
8.3.2	Camera feature entity.....	108
8.3.3	Profile: Colour model for CFI	109
8.4	Ultrasonic feature interface	110
8.4.1	Ultrasonic feature header	113
8.4.2	Ultrasonic feature entity	114
9	Detection level.....	114
9.1	General.....	114
9.2	Generic sensor detection interface.....	115
9.2.1	Generic sensor detections header.....	115
9.2.2	Generic sensor detections entity	116
9.3	Radar detection interface	117
9.3.1	Radar detections header	119
9.3.2	Radar detections entity	121
9.3.3	Profile: Radar ambiguity	121
9.4	Lidar detection interface	121
9.4.1	Lidar detection header.....	124
9.4.2	Lidar detection entity	125
9.5	Camera detection interface	125
9.5.1	Camera detection header	130
9.5.2	Camera detection entity	131
9.5.3	Profile: Colour model for CDI.....	132
9.6	Ultrasonic detection interface.....	132
9.6.1	Ultrasonic detection header.....	135
9.6.2	Ultrasonic detection entity	136
9.6.3	Profile: Ultrasonic sensor cluster.....	137
10	Supportive sensor interfaces.....	139
10.1	General.....	139
10.2	Generic supportive sensor interface.....	140
10.2.1	Generic supportive sensor header.....	140
10.2.2	Generic supportive sensor entity	141
10.3	Sensor performance interface.....	141
10.3.1	Sensor performance header.....	145
10.3.2	Sensor performance entity	147
10.3.3	Profile: Uniqueness of interface versioning of SPIs.....	147
10.4	Sensor health information interface.....	147
10.4.1	Sensor health information header.....	150
10.4.2	Sensor health information entity	151
11	Sensor input interface	151
11.1	General.....	151
11.2	Generic sensor input interface.....	151
11.2.1	Generic sensor inputs header.....	151
11.2.2	Generic sensor inputs entity	152
11.2.3	Profile: Uniqueness of interface versioning of SII	152
11.3	Common sensor input interface	152
11.3.1	Common sensor input header	156
11.3.2	Common sensor input entity.....	156

Annex A (normative) Interface signals	158
Annex B (normative) Options and constraints	328
Bibliography	344

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*, Subcommittee SC 31, *Data communication*.

This second edition cancels and replaces the first edition (ISO 23150:2021), which has been technically revised.

The main changes are as follows:

- extension of the potentially moving object interface at object level (extension of the logical signal group person);
- update of the road object interface at object level (extension of the road marking sign);
- extension of the static object interface at object level (extension of the entity type traffic sign; addition of new entity type traffic sign board);
- addition of new free space area interface at object level;
- extension of the camera detection interface at detection level (addition of new entity type point; update of existing entity type shape);
- addition of new interfaces group/layer – sensor input interface (addition of new generic sensor input interface; addition of new common sensor input interface);

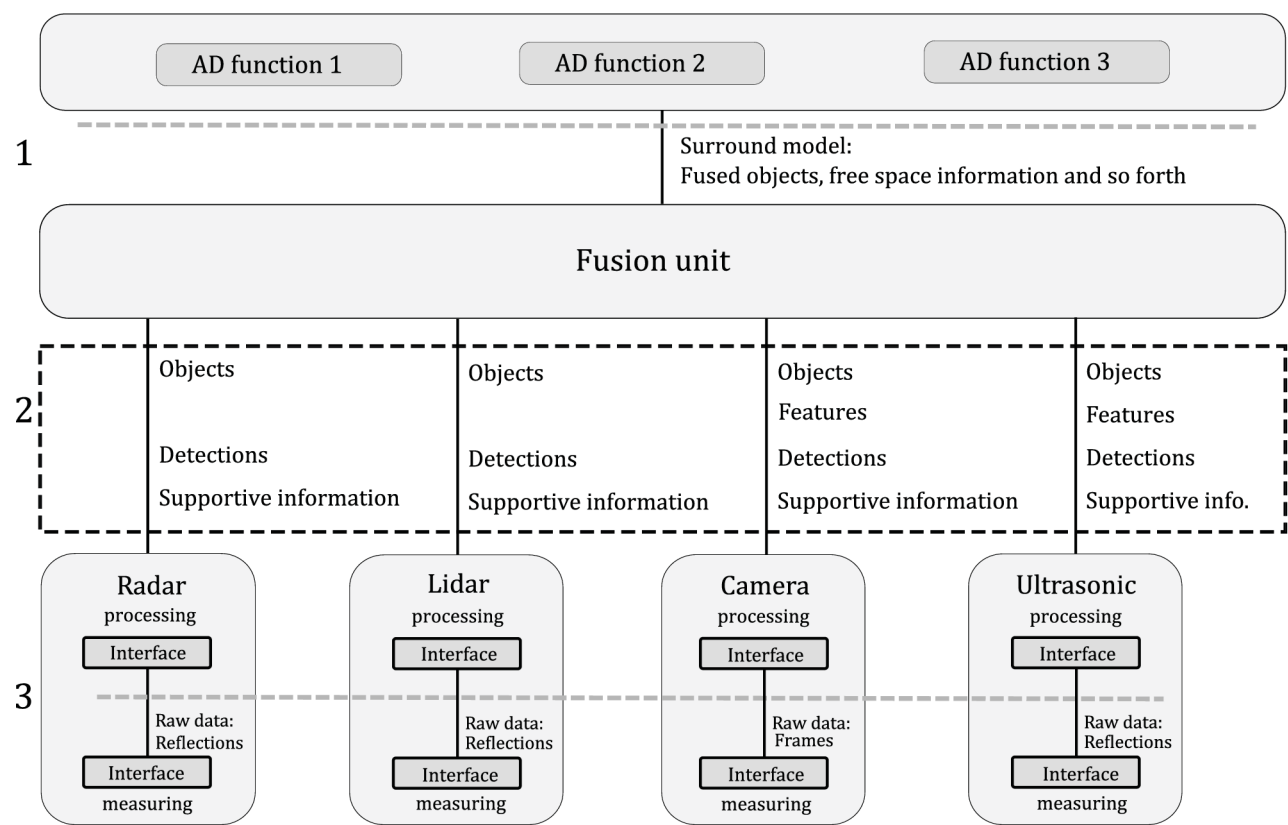
- extension of the error model (addition of covariances, cross-covariances and time series as error model implementation);
- refinement of the terms, for example, value as measured-, tracked- and predicted quantity value;
- new measures to link signals with their origin, that means linking signals at object level with, for example, detection entities;
- harmonisation of the document, for example, to achieve a better readability;
- update and add figures for clarification.

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

(Highly-)automated driving (AD) functions for road vehicles require a situation awareness of the surroundings of the vehicle and, preferably, a comprehensive scene understanding. For the fast and reliable recognition of real-world objects, a sensor suite is necessary to provide information for the fusion unit. Utilisation of different sensor technologies like radar, lidar, camera and ultrasonic with different detection capabilities is indispensable to ensure both complementary and redundant information. The fusion unit analyses and evaluates the different sensor signals and finally generates a dynamic surround model with sufficient scene understanding.

While current partly-automated functions utilise only particular objects (for example, vehicles, pedestrians, road markings) to generate a simple surround model, it is necessary for future highly-automated driving functions to merge not only the recognised objects but also to include other sensor-specific properties and characteristics of these objects for the generation of a coherent surround model of the surroundings. To minimise the development efforts for the sensors and the fusion unit and to maximise the reusability of development and validation efforts for the different functions on the sensor and fusion unit side, a standardised logical interface layer between the sensor suite and the fusion unit and a standardised logical interface layer to the sensor suite are worthwhile and beneficial for both the sensor supplier and the system supplier.

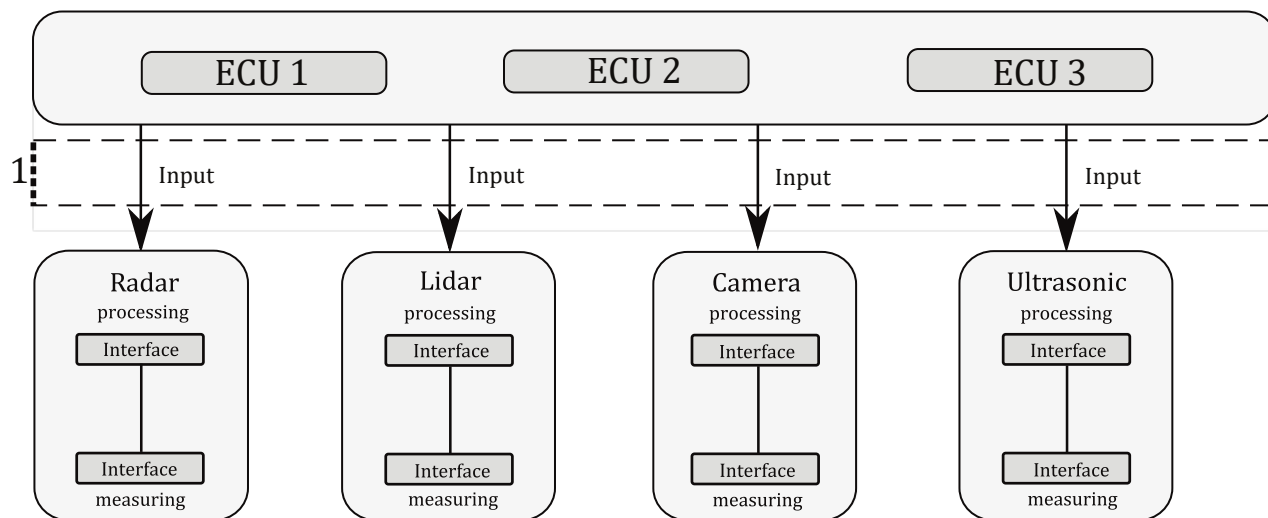


- Key**
- 1 logical interface layer between the fusion unit and AD functions
 - 2 logical interface layer between a single sensor as well as a single sensor cluster and the fusion unit
 - 3 interface layer on raw data level of a sensor’s sensing element(s) and its processing

Figure 1 — Architecture: sensors/sensor clusters – fusion unit – AD functions

The logical interface layer between a single sensor as well as a single sensor cluster and the fusion unit (see Figure 1, key 2) addresses the encapsulation of technical complexity as well as objects including

free space areas, features and detections to enable object-level, feature-level and detection-level fusion. Additional supportive information of the sensor as well as the sensor cluster will supplement the data for the fusion unit.



Key

- 1 logical interface layer between other in-vehicle electronic control units (ECUs) (for example, odometry) and a single sensor or a single sensor cluster

Figure 2 — Architecture: ECUs' sensor input – sensors/sensor clusters

The logical interface layer between an electronic control unit and a single sensor as well as a single sensor cluster (see Figure 2, key 1) addresses the input of a single sensor as well as a single sensor cluster.

Road vehicles — Data communication between sensors and data fusion unit for automated driving functions — Logical interface

1 Scope

This document is applicable to road vehicles with automated driving functions. The document specifies the logical interface between in-vehicle environmental perception sensors (for example, radar, lidar, camera, ultrasonic) and the fusion unit which generates a surround model and interprets the scene around the vehicle based on the sensor data. The interface is described in a modular and semantic representation and provides information on object level (for example, potentially moving objects, road objects, static objects) as well as information on feature and detection levels based on sensor technology specific information. Further supportive information is available.

This document does not provide electrical and mechanical interface specifications. Raw data interfaces are also excluded.

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 terminology 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 Architectural components

3.1.1 fusion

act of uniting *signals* (3.3.1) from two or more *sensors* (3.1.5) as well as *sensor clusters* (3.1.6) to create a *surround model* (3.1.7)

3.1.2 fusion unit

computing unit where the *fusion* (3.1.1) of *sensor* (3.1.5) data as well as a *sensor cluster* (3.1.6) data is performed

3.1.3 interface

shared boundary between two functional units, defined by various characteristics pertaining to the functions, physical interconnections, *signal* (3.3.1) exchanges and other characteristics of the units, as appropriate

[SOURCE: ISO/IEC 2382:2015, 2124351, modified — Notes to entry have been removed.]

3.1.4

logical interface

interface (3.1.3) between a *sensor* (3.1.5) as well as a *sensor cluster* (3.1.6) and the *fusion unit* (3.1.2), defined by logical characteristics

Note 1 to entry: Logical means a semantic description of the interface.

Note 2 to entry: Mechanical and electrical interfaces are excluded.

Note 3 to entry: This document uses the term interface as a shortcut for the term logical interfaces.

3.1.5

sensor

in-vehicle unit which detects entities external of the vehicle with pre-processing capabilities serving at least one *logical interface* (3.1.4)

Note 1 to entry: A sensor may use one or more sensing elements.

3.1.6

sensor cluster

group of *sensors* (3.1.5) of the same technology serving common *logical interfaces* (3.1.4)

Note 1 to entry: A sensor cluster can exceptionally consist of only one sensor.

EXAMPLE A stereo camera, a surround-view camera, an ultrasonic sensor array, a corner radar system.

3.1.7

surround model

representation of the real world adjacent to the ego-vehicle

3.1.8

in-vehicle communication

communication network used in vehicles to connect devices to exchange information

Note 1 to entry: A in-vehicle communication connects, for example, electric control units and *sensors* (3.1.5) with each other.

3.2 Terms for logical interface layers

3.2.1

detection

sensor technology specific entity represented in the *sensor coordinate system* (3.7.18) based on a single *measurement* (3.4.5) of a *sensor* (3.1.5)

Note 1 to entry: A small amount of history can be used for some detection *signals* (3.3.1), for example, model-free filtering may be used in track-before-detect algorithms.

3.2.2

detection level

set of *logical interfaces* (3.1.4) that provides *detections* (3.2.1)

3.2.3

feature

sensor technology specific entity represented in the *vehicle coordinate system* (3.7.16) based on multiple *measurements* (3.4.5)

Note 1 to entry: Multiple measurements can originate from a *sensor cluster* (3.1.6).

Note 2 to entry: Multiple measurements can originate from multiple *measurement cycles* (3.4.1).

Note 3 to entry: The term feature is used in this document not as function or group of functions as specified in ISO/SAE PAS 22736:2021.

3.2.4

feature level

set of *logical interfaces* (3.1.4) that provides *features* (3.2.3)

3.2.5

object

representation of a real-world entity with defined boundaries and characteristics in the *vehicle coordinate system* (3.7.16)

Note 1 to entry: The geometric description of the object is in the vehicle coordinate system.

Note 2 to entry: Object *signals* (3.3.1) are basically sensor technology independent. Sensor technology specific signals may extend the object signals.

EXAMPLE A *potentially moving object* (3.6.1), a *road object* (3.6.2), a *static object* (3.6.3), a *free space area object* (3.6.4).

3.2.6

object level

set of *logical interfaces* (3.1.4) that provides *objects* (3.2.5)

3.2.7

sensor input

data received by a *sensor* (3.1.5) or a *sensor cluster* (3.1.6) via the *in-vehicle communication* (3.1.8)

3.3 Structure terms

3.3.1

signal

entity consisting of one or more values and which is part of a *logical interface* (3.1.4)

3.3.2

logical signal group

grouping of *signals* (3.3.1) that has a logical relationship and a name for the grouping

3.3.3

classification

attribute-based differentiation

Note 1 to entry: An attribute is defined by a list of enumerators.

3.4 Measurement terms

3.4.1

measurement cycle

time period from the start of a data acquisition event to the start of the next data acquisition event

Note 1 to entry: A measurement cycle of one *sensor* (3.1.5) is a consistent view of an observed scene and not overlapping in time.

3.4.2

measured quantity value

value of a quantity resulting from a *measurement* (3.4.5)

3.4.3

tracked quantity value

value of a quantity determined from observed sequential changes, using information related to the same characteristic

3.4.4

predicted quantity value

value of a quantity assessed before it is actually observable, using information related to the same characteristic

EXAMPLE Related information can be recent and previous *measured quantity values* (3.4.2), *tracked quantity values* (3.4.3) and state variables.

[SOURCE: IEV 192-13-02, modified — EXAMPLE has been added and the word "quantity" has been added to the term.]

3.4.5

measurement

processing result of a *measurement cycle* (3.4.1)

3.4.6

tracking

computation process used to calculate the *tracked quantity value* (3.4.3) of a quantity

3.4.7

prediction

computation process used to obtain the *predicted quantity value* (3.4.4) of a quantity

[SOURCE: IEV 192-11-01]

3.4.8

error

discrepancy between a *measured quantity value* (3.4.2), *tracked quantity value* (3.4.3) or *predicted quantity value* (3.4.4) or condition, and the true, specified or theoretically correct reference quantity value or condition

Note 1 to entry: An error within a system can be caused by failure of one or more of its components, or by the activation of a systematic fault.

[SOURCE: IEV 192-03-02, modified — "computed, observed or measured value" was modified to "measured quantity value, tracked quantity value or predicted quantity value", "value" was modified to "reference quantity value", Note 1 to entry has been adapted and Note 2 to entry was deleted.]

3.4.9

accuracy

closeness of agreement between a *measured quantity value* (3.4.2), *tracked quantity value* (3.4.3) or *predicted quantity value* (3.4.4) and a true quantity value

Note 1 to entry: The concept accuracy is not a quantity and is not given a numerical quantity value. A *measurement* (3.4.5), *tracking* (3.4.6) or *prediction* (3.4.7) is said to be more accurate when it offers a smaller *error* (3.4.8).

Note 2 to entry: The term accuracy should not be used for *trueness* (3.4.10) and the term *precision* (3.4.11) should not be used for accuracy, which, however, is related to both these concepts.

Note 3 to entry: Accuracy is sometimes understood as closeness of agreement between measured, tracked or predicted quantity values that are being attributed to the measurand.

[SOURCE: ISO/IEC Guide 99:2007, 2.13, modified — The terms "measurement accuracy" and "accuracy of measurement" were deleted, definition was extended for tracked or predicted quantity values and the Notes to entry have been adapted.]

3.4.10

trueness

closeness of agreement between the average of an infinite number of replicated *measured quantity values* (3.4.2), *tracked quantity values* (3.4.3) or *predicted quantity values* (3.4.4) and a reference quantity value

Note 1 to entry: Trueness is not a quantity and thus cannot be expressed numerically, but measures for closeness of agreement are given in the ISO 5725 series.

Note 2 to entry: Trueness is inversely related to systematic error but is not related to random error.

Note 3 to entry: The term *accuracy* (3.4.9) should not be used for trueness.

[SOURCE: ISO/IEC Guide 99:2007, 2.14, modified — The terms "measurement trueness" and "trueness of measurement" were deleted, definition was extended for tracked or predicted quantity values and the Notes to entry have been adapted.]

3.4.11

precision

closeness of agreement between indications or *measured quantity values* (3.4.2), *tracked quantity value* (3.4.3) or *predicted quantity values* (3.4.4) obtained by replicate *measurements* (3.4.5), *tracking* (3.4.6) or *prediction* (3.4.7) on the same or similar measurands under specified conditions

Note 1 to entry: Precision is usually expressed numerically by measures, trackings or predictions of imprecision, such as standard deviation, variance or coefficient of variation under the specified conditions of measurement, tracking or prediction.

Note 2 to entry: The specified conditions can be, for example, repeatability conditions of measurement, intermediate precision conditions of measurement or reproducibility conditions of measurement (see ISO 5725-1:1994).

Note 3 to entry: Precision is used to define measurement, tracking or prediction repeatability, intermediate measurement or prediction precision and measurement, tracking or prediction reproducibility.

Note 4 to entry: Sometimes precision is erroneously used to mean *accuracy* (3.4.9).

Note 5 to entry: Precision is inversely related to random error but is not related to systematic error.

[SOURCE: ISO/IEC Guide 99:2007, 2.15, modified — The term "measurement precision" was deleted, the word "objects" was replaced by "measurands", definition was extended for tracked or predicted quantity values, the Notes to entry have been adapted and Note 5 to entry has been added.]

3.4.12

measurement error

measured quantity value (3.4.2) minus a reference quantity value

Note 1 to entry: The concept of measurement error can be used both:

- a) when there is a single reference quantity value to refer to, which occurs if a calibration is made by means of a measurement standard with a measured quantity value having a negligible measurement uncertainty or if a conventional quantity value is given, in which case the *error* (3.4.8) is known, and
- b) if a measurand is supposed to be represented by a unique true quantity value or a set of true quantity values of negligible range, in which case the error is not known.

Note 2 to entry: Measurement error should not be confused with production error or mistake.

[SOURCE: ISO/IEC Guide 99:2007, 2.16, modified — The terms "error" and "error of measurement" were deleted and the Notes to entry have been adapted.]

3.4.13

tracking error

quantitative statement about the *tracked quantity value* (3.4.3) and the reference quantity value

3.4.14

prediction error

quantitative statement about the *predicted quantity value* (3.4.4) and the reference quantity value

3.4.15

error model

model used to estimate the *error* (3.4.8)

3.4.16

fixation

short temporal holds of movements that keep alignment of the eyes to a particular point within an area of interest which falls on the fovea (the middle of the retina responsible for our central, sharpest vision) for a given time period

[SOURCE: ISO 15007:2020, 3.1.4, modified — Notes to entry have been deleted.]

3.5 Requirement level terms

3.5.1

conditional

required under certain specified conditions

Note 1 to entry: One of three obligation statuses applied to a *requirement level* (3.5.4) of a *logical interface* (3.1.4) specification, indicating the conditions under which the *signal* (3.3.1) or *logical signal group* (3.3.2) is required. In other cases, the signal or logical signal group is optional. See also *mandatory* (3.5.2) and *optional* (3.5.3).

[SOURCE: ISO/IEC 11179-3:2023, 3.2.77, modified — Note 1 to entry has been adapted and Note 2 to entry has been removed.]

3.5.2

mandatory

always required

Note 1 to entry: One of three obligation statuses applied to a *requirement level* (3.5.4) of a *logical interface* (3.1.4) specification, indicating the conditions under which the *signal* (3.3.1) or *logical signal group* (3.3.2) is required. See also *conditional* (3.5.1) and *optional* (3.5.3).

[SOURCE: ISO/IEC 11179-3:2023, 3.2.75, modified — Note 1 to entry has been adapted and Note 2 to entry has been removed.]

3.5.3

optional

permitted but not required

Note 1 to entry: One of three obligation statuses applied to a *requirement level* (3.5.4) of a *logical interface* (3.1.4) specification, indicating the conditions under which the *signal* (3.3.1) or *logical signal group* (3.3.2) is required. See also *conditional* (3.5.1) and *mandatory* (3.5.2).

[SOURCE: ISO/IEC 11179-3:2023, 3.2.76, modified — Note 1 to entry has been adapted and Note 2 to entry has been removed.]

3.5.4

requirement level

definition of the obligation status of a *logical interface's* (3.1.4) *logical signal group* (3.3.2), *signal* (3.3.1) as well as a signal's identifier or signal's enumerator

Note 1 to entry: Each requirement level entry has one of three possible obligation statuses applied: *conditional* (3.5.1), *mandatory* (3.5.2) or *optional* (3.5.3).

3.6 Road user relevant entity types

3.6.1

potentially moving object

real-world entity which can potentially move and is relevant for driving situations

Note 1 to entry: A representation of a potentially moving object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

EXAMPLE A vehicle, a bicycle, a pedestrian, an obstacle.

3.6.2

road object

marking or structure of a road which is relevant for driving situations

Note 1 to entry: A representation of a road object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

EXAMPLE A *road marking* (3.6.2.1), a *road boundary* (3.6.2.2), the *road surface* (3.6.2.3).

3.6.2.1

road marking

line, symbol or other mark on the surface of a road or a structure intended to limit, regulate, warn, guide or inform road users

Note 1 to entry: Other marks could be text, numbers, arrows or combinations.

EXAMPLE A lane marking, Botts' dots.

[SOURCE: ISO 6707-1:2020, 3.3.5.80, modified — "user" was modified to "road users", "limit" was added and the Note 1 to entry and EXAMPLE have been added.]

3.6.2.2

road boundary

structure that limits the road

EXAMPLE A curb stone, a guard rail, the end of the surface of the road.

3.6.2.3 road surface

surface supporting the tyre and providing friction necessary to generate shear forces in the *road plane* (3.7.6)

Note 1 to entry: The surface may be flat, curved, undulated or of other shape.

[SOURCE: ISO 8855:2011, 2.6]

3.6.3 static object

real-world stationary entity which can be used for information and/or localisation

Note 1 to entry: A representation of a static object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

EXAMPLE A *general landmark* (3.6.3.1), a *traffic sign* (3.6.3.2), a *traffic sign board* (3.6.3.3), a *traffic light* (3.6.3.4).

3.6.3.1 general landmark

real-world stationary entity which can be used for localisation

Note 1 to entry: A stationary *traffic sign* (3.6.3.2) or *traffic light* (3.6.3.4) is also regarded as a general landmark.

EXAMPLE A building, a tunnel, a bridge, a sign gantry structure, a tree.

3.6.3.2 traffic sign

traffic relevant, authorised sign that limits, regulates, warns, guides or informs road users

Note 1 to entry: One traffic sign usually consists of one *main sign* (3.6.3.2.1) and none, one or several *supplementary signs* (3.6.3.2.2).

EXAMPLE A speed limit which is restricted for trucks.

3.6.3.2.1 main sign

traffic sign (3.6.3.2) which gives a general message, obtained by a combination of colour and geometric shape and which, by the addition of a graphical symbol or text, gives a particular message for road users

[SOURCE: ISO 3864-1:2011, 3.12, modified — The original term was "safety sign", "sign" has been replaced by "traffic sign" and the phrases "or text" and "for road users" have been added to the definition.]

3.6.3.2.2 supplementary sign

traffic sign (3.6.3.2) that is supportive of a *main sign* (3.6.3.2.1) and the main purpose of which is to provide additional clarification

[SOURCE: ISO 3864-1:2011, 3.14, modified — "traffic sign" now replaces "sign" and "main sign" replaces "traffic sign".]

3.6.3.3 traffic sign board

traffic relevant, authorised sign board that limits, regulates, warns, guides or informs road users by abstract representations of lanes and streets ahead to control traffic on the road

3.6.3.4

traffic light

traffic relevant, official lights

Note 1 to entry: One traffic light consists of one or several light spots with different light colours and/or shapes.

EXAMPLE A pedestrian traffic light.

3.6.4

free space area object

area entity of the *road surface* (3.6.2.3) that can be driven on and is free of obstacles that can limit driving

Note 1 to entry: A representation of a free space area object is part of *logical interfaces* (3.1.4) on *object level* (3.2.6).

3.7 Axis and coordinate system terms

3.7.1

reference frame

geometric environment in which all points remain fixed with respect to each other at all times

[SOURCE: ISO 8855:2011, 2.1]

3.7.2

axis system

set of direction vectors

3.7.3

coordinate system

numbering convention used to assign a unique ordered trio of values to each point in a *reference frame* (3.7.1) and which consists of an *axis system* (3.7.2) plus an origin point

[SOURCE: ISO 8855:2011, 2.4, modified — "(x, y, z)" has been removed from the definition.]

3.7.4

cartesian coordinate system

set of numerical coordinates (x, y, z), which are the signed distances to the YZ-, ZX- and XY-planes using a set of three orthogonal directions associated with X, Y and Z axes

Note 1 to entry: A right-handed *axis system* (3.7.2) is assumed throughout this document, where: $\vec{Z} = \vec{X} \times \vec{Y}$.

3.7.5

spherical coordinate system

set of one distance vector and two angles associated with radial distance, azimuth and elevation

Note 1 to entry: The azimuth angle is the angle in XY-plane [of a *cartesian coordinate system* (3.7.4)] of the *axis system* (3.7.2) counted from the X-axis. The elevation angle is the angle from the azimuth direction in the XY-plane of the axis system towards the direction of the distance vector, that is XY-plane has an elevation angle = 0 rad.

Note 2 to entry: The angles of the spherical coordinate system have increasing values in counterclockwise direction.

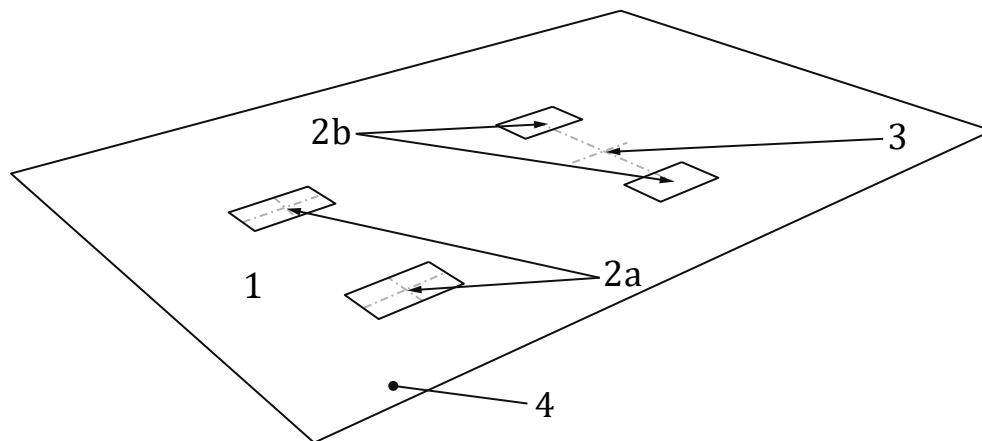
3.7.6

road plane

plane representing the *road surface* (3.6.2.3) within the front tyre contact patches and the *vehicle road-level reference point* (3.7.13)

Note 1 to entry: See Figure 3.

Note 2 to entry: For tyre contact patches, see ISO 8855:2011, 4.1.5.



Key

- 1 vehicle front
- 2a vehicle's front tyre contact patches
- 2b vehicle's rear tyre contact patches
- 3 vehicle road-level reference point (3.7.13)
- 4 vehicle road plane

Figure 3 — Road plane

[SOURCE: ISO 8855:2011, 2.7, modified — “tyre contact patch” was modified to “front tyre contact patches and the vehicle road-level reference point”, the figure has been added and the Notes to entry have been modified.]

3.7.7

road level

point related to a *road plane* (3.7.6)

3.7.8

vehicle unsprung mass

unsprung mass

mass that is not carried by the suspension, but is supported directly by the tyres

[SOURCE: ISO 8855:2011, 4.11, modified — The term “vehicle unsprung mass” has been added.]

3.7.9

vehicle sprung mass

sprung mass

mass that is supported by the suspension, that is the total vehicle mass less the *vehicle unsprung mass* (3.7.8)

[SOURCE: ISO 8855:2011, 4.12, modified — The term vehicle sprung mass has been added and Note 1 to entry has been removed.]

3.7.10**vehicle rear-axle reference point**

point fixed in the *vehicle sprung mass* (3.7.9) and located at the centre of the rear-axle

3.7.11**vehicle sprung mass axis system**

axis system (3.7.2) fixed in the *reference frame* (3.7.1) of the *vehicle sprung mass* (3.7.9), so that the *X*-axis is substantially horizontal and forwards (with the vehicle at rest) and is parallel to the vehicle's longitudinal plane of symmetry, additionally the *Y*-axis is perpendicular to the vehicle's longitudinal plane of symmetry and points to the left with the *Z*-axis pointing upward

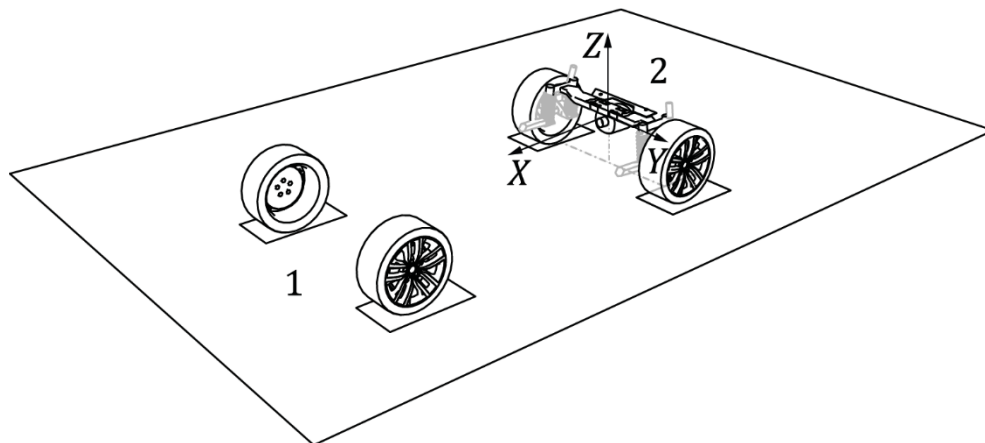
Note 1 to entry: The axis system is a set of three orthogonal directions associated with *X*, *Y* and *Z* axes.

3.7.12**vehicle rear-axle coordinate system**

cartesian coordinate system (3.7.4) based on the *vehicle sprung mass axis system* (3.7.11) with the origin located at the *vehicle rear-axle reference point* (3.7.10)

Note 1 to entry: The vehicle rear-axle coordinate system is a *vehicle coordinate system* (3.7.16).

Note 2 to entry: See Figure 4.

**Key**

- 1 vehicle front
- 2 vehicle rear-axle reference point (3.7.10)

Figure 4 — Vehicle rear-axle coordinate system

3.7.13**vehicle road-level reference point**

point at *road level* (3.7.7) located in the middle of the rear tyre contact patches

Note 1 to entry: For tyre contact patches, see ISO 8855:2011, 4.1.5.

3.7.14**vehicle road-level axis system**

axis system (3.7.2) fixed in the *reference frame* (3.7.1) of the *vehicle unsprung mass* (3.7.8), so that the *X*-axis is parallel to the vehicle's longitudinal plane of symmetry and points into forward moving direction and the *Y*-axis is perpendicular to the vehicle's longitudinal plane of symmetry and points to the left with the *Z*-axis pointing upward

Note 1 to entry: Vehicle road-level axis system's XY -plane is parallel to the ego-vehicle's *road plane* (3.7.6).

Note 2 to entry: The axis system is a set of three orthogonal directions associated with X , Y and Z axes.

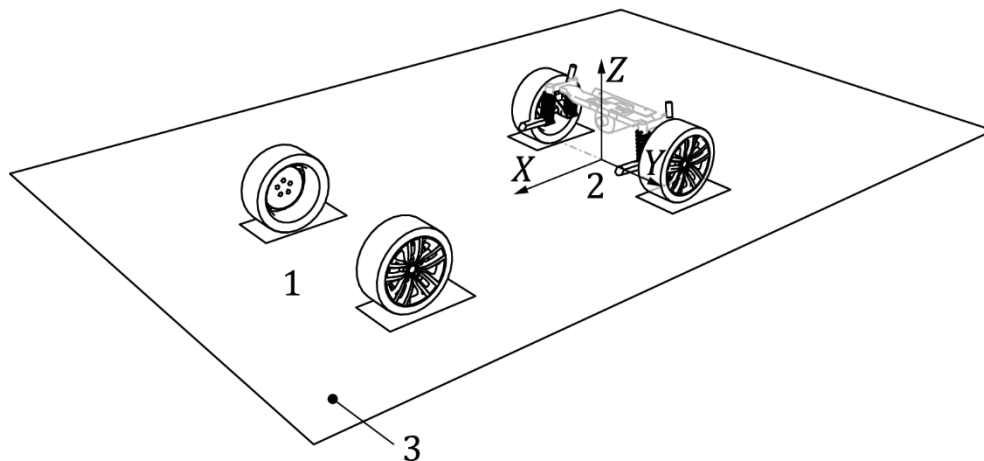
3.7.15

vehicle road-level coordinate system

cartesian coordinate system (3.7.4) based on the *vehicle road-level axis system* (3.7.14) with the origin located at the *vehicle road-level reference point* (3.7.13) at the *vehicle road level* (3.7.7)

Note 1 to entry: The vehicle road-level coordinate system is a *vehicle coordinate system* (3.7.16).

Note 2 to entry: See Figure 5.



Key

- 1 vehicle front
- 2 vehicle road-level reference point (3.7.13)
- 3 vehicle road plane (3.7.6)

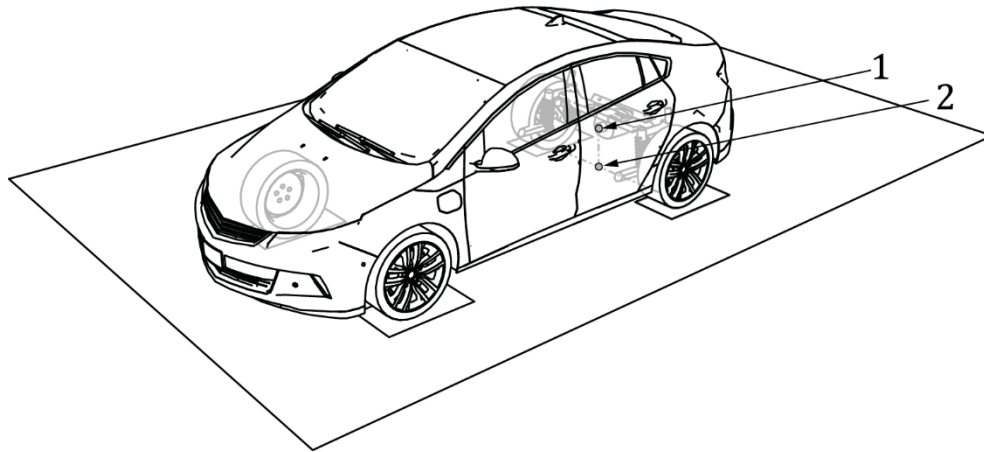
Figure 5 — Vehicle road-level coordinate system

3.7.16

vehicle coordinate system

cartesian coordinate system (3.7.4) which is either the *vehicle rear-axle coordinate system* (3.7.12) or the *vehicle road-level coordinate system* (3.7.15)

Note 1 to entry: See Figure 6.

**Key**

- 1 vehicle rear-axle reference point (3.7.10) of the vehicle rear-axle coordinate system (3.7.12)
- 2 vehicle road-level reference point (3.7.13) of the vehicle road-level coordinate system (3.7.15)

Figure 6 — Vehicle coordinate systems**3.7.17****sensor axis system**

axis system (3.7.2) fixed in the reference frame (3.7.1) of the sensor (3.1.5)

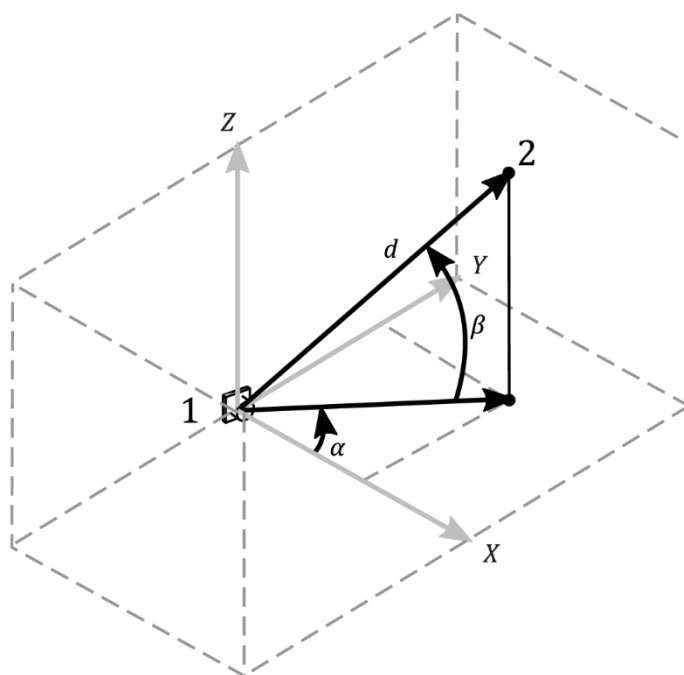
Note 1 to entry: The *X*-axis [of a *cartesian coordinate system* (3.7.4)] is in viewing direction of the sensor and the *Z*-axis pointing upward.

3.7.18**sensor coordinate system**

spherical coordinate system (3.7.5) based on the sensor axis system (3.7.17) at a defined origin point of the sensor (3.1.5)

Note 1 to entry: The origin point of the sensor coordinate system has to be selected in a way that *detections* (3.2.1) could easily be specified in a spherical coordinate system. For example, the origin point of a camera sensor is the virtual projection centre of the camera's optics.

Note 2 to entry: See Figure 7.



Key

- 1 sensor (3.1.5) with origin point of the sensor coordinate system
- 2 detection (3.2.1)
- d radial distance
- α azimuth angle
- β elevation angle

Figure 7 — Sensor coordinate system

3.7.19

affine coordinate system

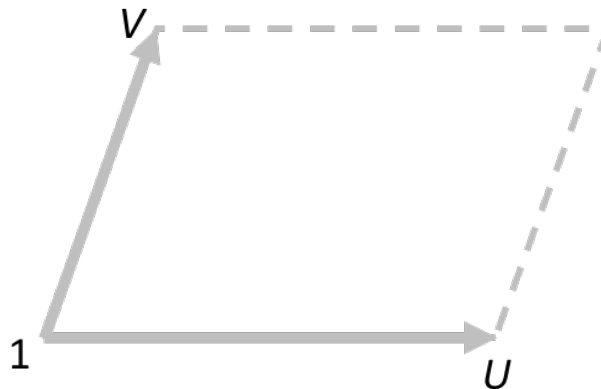
coordinate system (3.7.3) of an *object* (3.2.5) with an ordered set of non-collinear direction vectors (an affine basis) and a defined origin point

Note 1 to entry: The length of the direction vectors may not be normalised.

Note 2 to entry: For a planar affine coordinate system of an object, a pair of direction vectors (abscissa and ordinate) and an origin point are necessary to define the affine coordinate system.

Note 3 to entry: A right-handed *axis system* (3.7.2) is assumed for a three-dimensional affine coordinate system (with the direction vectors abscissa, ordinate and applicate).

Note 4 to entry: See Figure 8.

**Key**

- 1 origin point of the affine coordinate system
- U* *U*-axis – abscissa of the affine coordinate system
- V* *V*-axis – ordinate of the affine coordinate system

Figure 8 — Affine coordinate system**4 Abbreviated terms**

A2I	assign to interface
AD	automated driving
AER	alternative entity representation
C	conditional
CDI	camera detection interface
CFI	camera feature interface
CSII	common sensor input interface
D	dimensional
DLI	detection level interface
ECU	electronic control unit
EM	error model
FFT	fast Fourier transform
FLI	feature level interface
FLP	front-left position of the ego-vehicle
FOV	field-of-view
FRP	front-right position of the ego-vehicle
FSAOI	free space area object interface
FWHM	full width at half maximum
GPS	global positioning system
HW	hardware
ID	identifier
IQR	interquartile range

IR	infrared
IRI	international roughness index
IV	implicit value
LDI	lidar detection interface
LL	list length
LSG	logical signal group
M	mandatory
NIR	near infrared
O	optional
OLI	object level interface
PMOI	potentially moving object interface
RCS	radar cross section
RDI	radar detection interface
RDOI	road object interface
RL	requirement level
RLP	rear-left position of the ego-vehicle
RRP	rear-right position of the ego-vehicle
SHII	sensor health information interface
SII	sensor input interface
SNR	signal to noise ratio
SOI	static object interface
SPI	sensor performance interface
SSI	supportive sensor interface
UDI	ultrasonic detection interface
UFI	ultrasonic feature interface
UTL	unrolling tuple list
VRO	value representation optimisation

5 Structure of the interface description

5.1 General

The scope of this document are logical interfaces of sensors and sensor clusters. Therefore, the following parts of this document use the term interface as a shortcut for the defined term “logical interface” (3.1.4). The interface descriptions have the following structure:

- a description of each interface by logical signal group (LSGs), signal names and requirement levels (RLs) (including conditional requirements) and additional options;
- a detailed description of each signal in Annex A;
- generic options and generic conditional requirements in Annex B, which shall be used.

The requirement level attribute “conditional” shall be an element used in this document. The logical interlink as well as complex dependencies between signals of an interface shall be defined in an interface’s subclause as a profile subclause of the interface.

5.2 Signal

Table 1 provides an overview of a signal used by an interface which is defined in subclause “Interface” (5.3). All signals are specified in Annex A. Signal names shall be unique. Similar signals on different interface levels have the interface level added to the signal name to guarantee a unique signal naming.

Table 1 — Signal: <Signal name>

Name	Name of the signal and optional list of signal identifiers Each signal respectively signal identifier is implemented by exactly one value.		
Description	Definition of the signal		
Value type	Definition of the signal’s value type and optionally the number of values, specifically the number of dimensions defined by the number of identifiers (default dimension: 1D vector)	Unit	Unit definition of the signal value In this document, the unit defines no application or network representation of the signal value (for example, a scaling factor, a fix point representation, linear or non-linear increasing values).

In general, for each signal with value type “enumeration”, a list of enumerators or exemplary enumerators is defined in an additional table (see Table 2) in Annex A. Each list of exemplary enumerators is informative. During the system design phase, specific enumerators shall be defined, based upon the list of exemplary enumerators (see Annex A). This can be done by adding new, using a subset of or adapting the existing exemplary enumerators. The implementation of a list of exemplary enumerators shall contain at least one enumerator and should have default ones (B.1.7). An enumeration with exemplary enumerators may allow output of multiple enumerators belonging to different classes. Enumerators of different classes may be otherwise specified during the system design phase by separated enumeration signal identifiers.

Each enumeration with exemplary enumerators shall implement default enumerators (B.1.7). At least one enumerator shall be defined for each implemented exemplary enumeration list. An enumerator with exemplary enumerations may be clustered by multiple enumerations to deal with different classes of enumerators, rather than addressing multiple classes in the enumerators.

Table 2 — Enumeration: <Signal name>

Name	Description	RL enumerator
Name of the enumerator (the signal name is abbreviated in the name of the enumerator)	Definition of the enumerator	Requirement level of the enumerator (mandatory, conditional or optional) or “exemplary”

5.3 Interface

Table 3 provides an overview of a logical interface. An interface is comprised of signals. Signals can be part of a logical signal group (LSG) for ease of logical grouping.

Table 3 — Interface: <Interface name>

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
Name of the logical signal group (LSG) – for a logical group of multiple signals	Requirement level for LSG: mandatory , conditional or optional Conditional RL with a reference to a clause in Annex B	Name of the signal, optional: {list of signal identifiers} with a reference to a clause in Annex A	Requirement level for signal: mandatory , conditional or optional is specified for the signal, including all signal identifiers, or for each signal identifier individually Conditional RL with a reference to a clause in Annex B	Additional options for the signal: — a reference to a clause in Annex B — a reference to a profile subclause

Areas in the interface table marked in grey indicate that the enclosed signals are part of a list of entities. The valid number of entities is defined at the beginning of the grey area (specifically, grey area header).

If an interface provides multiple interface entity types (for example, the road object interface [RDOI] provides road surface, road markings and road boundaries), the interface may require not all entity types at once (see specified in B.2.3). Different entry types of an interface are separated with a double horizontal line.

A conditional LSG requirement level shall define the conditions. Under all other conditions, the LSG is optional [see the definition conditional (3.5.1)].

An LSG shall have at least one signal that is mandatory. If a signal of an optional LSG is used, all signals of the LSG shall be considered as defined on their signal requirement levels.

In case of a list, the requirement level of the signal “Number of valid <...>” and its LSG are the base requirement level of all signals in the list.

Symbol “” indicates that LSG belongs to another LSG, that means the LSG is a nested LSG.

If only one signal requirement level is defined for a signal with signal identifiers, the requirement level of each signal identifier individually corresponds to the requirement level for the signal.

Signals with RL mandatory and static or fixed values, which are defined during the system design phase and/or are implicitly known by the sensor and the fusion unit may be optimised. In this case the applicability of optimisation methods of B.1 are allowed which essentially means that, for example, the mandatory signals will not be explicitly transferred by the interface message. During the system design phase, additional measures from Annex B are applicable and shall be defined for the sensor as well as the sensor cluster.

5.4 Specific signal grouping

Table 4 provides an overview of a specific signal grouping, for example, an LSG, which refers a specific signal grouping of another subclause and extends the specific signal grouping to achieve similar interfaces.

Table 4 — Specific signal grouping: <Name for LSG(s)>

Signal	RL signal	Option
--------	-----------	--------

Name of the signal, optional: {list of signal identifiers} with a reference to a clause in Annex A	Requirement level for signal: mandatory, conditional or optional is specified for the signal, including all signal identifiers, or for each signal identifier individually Conditional RL with a reference to a clause in Annex B Optional: information of the RL for the signal's LSG, if it is not mandatory	Additional options for the signal: — a reference to a clause in Annex B — a reference to a profile subclause
---	---	---

Symbol “” indicates that the signal belongs to a list of signals.

Unchanged, inherited signals, signal requirement levels and signal options of the referenced subclause's specific signal grouping are marked in grey.

5.5 Profile

If multiple signals of an interface have an interface-specific semantic interdependency, this interdependency is described in a profile subclause and additional complex conditions are defined. Each signal which is contained in the profile has an annotation of the profile in the signal's option-column definition of the interface (see column “Option” in Table 3 and Table 4).

6 Logical interface from a sensor as well as a sensor cluster to a fusion unit

6.1 General

The aim of this document is to support different levels of fusion. Therefore, the document provides three logical interface levels (detection-, feature- and object level). Supportive sensor interfaces (SSIs) complement the interfaces of the three logical interface levels. Detections, features, objects and supportive sensor information are the basis for a fusion unit. A tracking algorithm within the sensor solely or the sensor cluster using their own information, results in objects with limited quality. The fusion unit can improve these and is able to merge the information on different levels, using object-level fusion as well as feature- and/or detection-level fusion. In order to perform object-level fusion from a sensor suite¹, sufficient evidence in the form of correlated detections belonging to a particular real-world entity shall be provided by each sensor and sensor cluster. An advantage of feature-level fusion is the possible creation of an object in the common fusion unit out of detections, features or objects coming from a sensor suite, which would not have resulted in an object recognition within each sensor or each sensor cluster itself. Detection-level fusion may use machine learning algorithms to handle complexity. Future vehicles will integrate a sensor suite with many sensors and sensor clusters using different sensor technologies. Different technologies will complement each other to achieve a robust and reliable fusion.

Supportive sensor interfaces are provided so that the fusion is able to process the reliability of the used measurement method or data. These interfaces are sensor technology independent, specifically there are uniform interfaces for various sensor technologies. These supportive sensor interfaces are also used for the implementation of safety concepts.

Sensor input interfaces are used to provide data and parametrise the sensors or the sensor cluster from other devices via the in-vehicle communication.

Figure 9 shows the interfaces that are in the scope of this document.

¹ It consists of two or more different and independent sensors or sensor clusters.

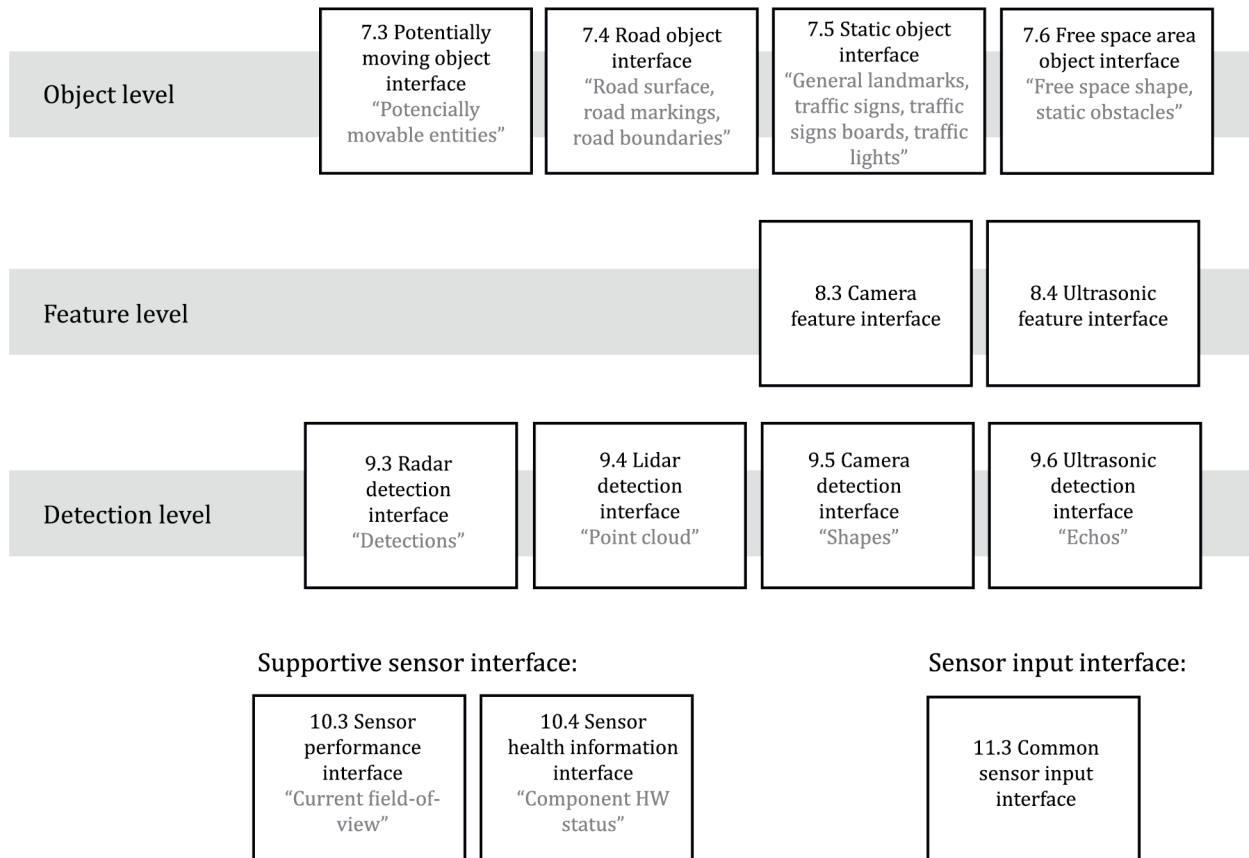


Figure 9 — Interfaces in the scope of this document

The interfaces covered in the scope of this document are the following:

- objects:
 - Potentially moving object interface (7.3): with specific attributes for, for example, lights, pedestrian;
 - Road object interface (7.4): road surface, road markings, road boundaries;
 - Static object interface (7.5): general landmarks, traffic signs, traffic sign boards, traffic lights;
 - Free space area object interface (7.6): free space areas;
- sensor or sensor cluster specific features:
 - Camera feature interface (8.3);
 - Ultrasonic feature interface (8.4);
- sensor or sensor cluster specific detections:
 - Radar detection interface (9.3);
 - Lidar detection interface (9.4);
 - Camera detection interface (9.5);
 - Ultrasonic detection interface (9.6);
- supportive information:
 - Sensor performance interface (10.3): with specific sensor-attributes for, for example, current field-of-view (FOV) segments, object recognition rates, reference target recognition rates;

- Sensor health information interface (10.4): with specific attributes for, for example, generic sensor statuses, calibration information, sensor cluster definition;
- sensor or sensor cluster specific input:
 - Common sensor input interface (11.3).

A sensor as well as a sensor cluster can provide several logical interfaces. They can provide none, one or several interfaces on object level. Additionally, a sensor as well as a sensor cluster can provide a sensor technology specific interface on feature- as well as detection level. Each sensor as well as each sensor cluster shall provide at least one interface either on object-, feature- or detection level to fulfil this document. Supportive sensor interfaces can be provided optionally. Sensors and sensor clusters may serve interfaces on all levels. Sensors mainly serve detection level interfaces (DLIs) whereas sensor clusters mainly serve feature level interfaces (FLIs) and/or object level interfaces (OLIs). Sensor clusters are based on sensors and therefore, sensor clusters without a group of defined sensors should be avoided. If a sensor cluster offers no DLIs, the sensor cluster can be considered as a single sensor with multiple sensing elements. A sensor cluster provides consistent data over all its interfaces. The fusion unit does not have to use each provided interface of the sensor or sensor cluster. The provided interfaces can be used by other applications (for example, AD functions), not only by the fusion unit alone.

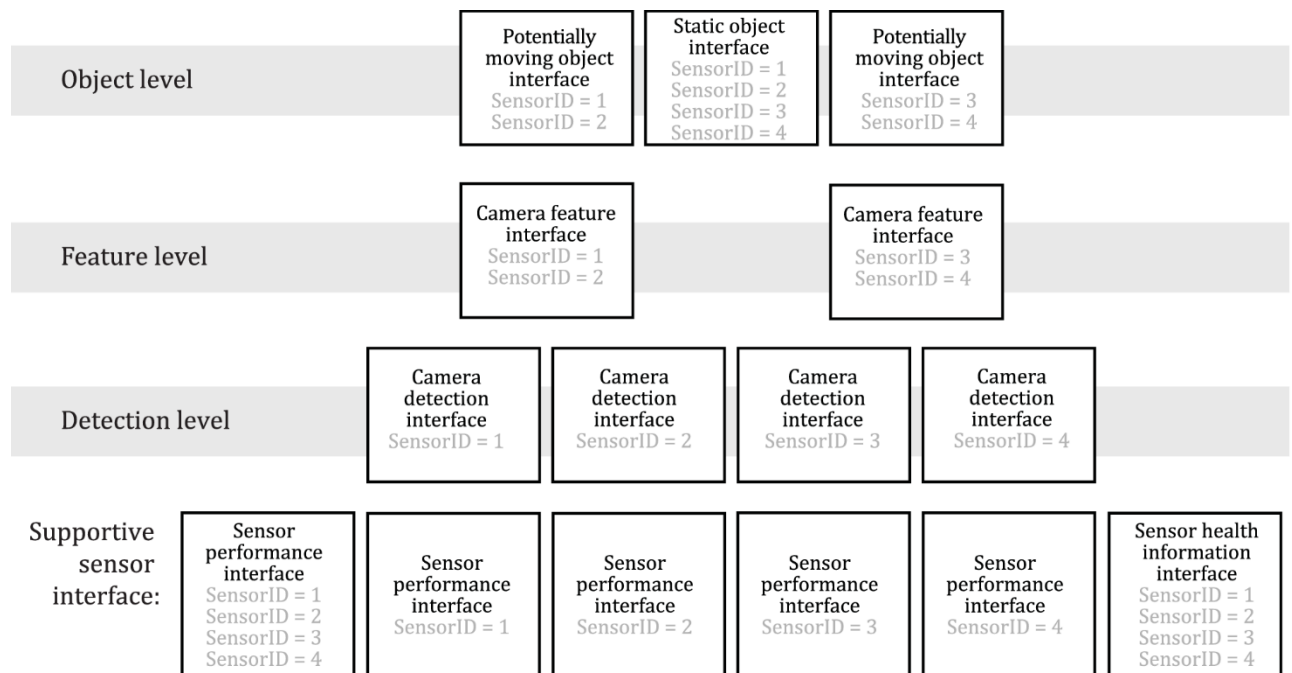


Figure 10 —Example of a camera sensor cluster and its sensor and sensor cluster interfaces

In Figure 10, sensor 1, sensor 2, sensor 3, sensor 4 and sensor sub-cluster {1, 2} and {3, 4} belong to the sensor cluster {1, 2, 3, 4}. The information of all sensors and sensor clusters belongs together and may be referenced between the interfaces. In this example, different combinations of sensors provide interfaces:

- the sensor cluster with the list of the signal “Sensor ID” (A.1.4) values {1, 2, 3, 4} provides the static object interface (SOI), sensor performance interface (SPI) and sensor health information interface (SHII);
- the sensor sub-cluster with the list of the signal “Sensor ID” (A.1.4) values {1, 2} provides potentially the moving object interface (PMOI), camera feature interface (CFI);

- the sensor sub-cluster with the list of the signal “Sensor ID” (A.1.4) values {3, 4} provides PMOI, CFI;
- the sensors with signal “Sensor ID” (A.1.4) value 1, 2, 3 or 4 provide each a camera detection interface (CDI) and SPI;
- the sensor or sensor cluster provides no free space area object interface;
- the sensor or sensor cluster may not receive a sensor input interface.

6.2 Generic interface header

Table 5 defines the structure of (6.2) as a specific signal grouping. This header is used by each interface of this document and is adapted or extended accordingly. However, all the interface headers have at least these or similar signals.

Table 5 — Specific signal grouping: Generic interface header

Signal	RL signal	Option
Interface version ID {major, minor, patch} (A.1.1)	Mandatory	Profile: Uniqueness of interface versioning (6.4)
Interface ID (A.1.2)	Optional	Profile: Uniqueness of interface versioning (6.4) Default enumerators (B.1.7)
Number of valid serving sensors (A.1.3)	Mandatory	Profile: Uniqueness of interface versioning (6.4) Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Sensor ID (A.1.4)
 Sensor ID (A.1.4)	Mandatory	Profile: Uniqueness of interface versioning (6.4) Alternative VRO (B.1.3)
Time stamp – <...> (A.1.5)	Mandatory	
<...> counter (A.1.6)	Optional	Redundancy (B.1.2) Signal: Time stamp – <...> (A.1.5)
Interface cycle time (A.1.7)	Optional	
Interface cycle time – variation (A.1.8)	Optional	
Data qualifier (A.1.9)	Mandatory	Default enumerators (B.1.7)
Interface specific header extensions		
Recognised <...> – capability (A.1.10)	Optional	
Recognised <...> – status (A.1.11)	Optional	Redundancy (B.1.2) Signal: Recognised <...> – capability (A.1.10) Default enumerators (B.1.7)
Number of valid <...> (A.1.12)	Mandatory	Optimise LL (B.1.1)

6.3 Generic interface entity

The interface entities define the set of signals without the header signals of the interface. Each logical interface provides at least one list of interface specific entities (except SHII and CSII). The number of valid entities of each interface entity type list is defined by the signal “Number of valid <...>” (A.1.12) (see Table 5). Interfaces with multiple entity types may not implement all interface entity type lists [see (B.2.3) for conditional entity types]. The signals “Recognised <...> – capability” (A.1.10) and “Recognised

<...> – status” (A.1.11) determine the minimal performance of the sensor for the interface entity type list, for example, at full load.

6.4 Profile: Uniqueness of interface versioning

The concatenation of the signal’s “Interface version ID {major, minor, patch}” (A.1.1) value, the signal’s “Interface ID” (A.1.2) value and the signal’s “Sensor ID” (A.1.4) values of a provided interface is a unique identifier for the interpretation of the interface message implementation. The signal “Number of valid serving sensors” (A.1.3) is also included. The order of the signal’s “Sensor ID” (A.1.4) values is also relevant. If a sensor could provide an interface more than once, this profile is adjusted in the corresponding interface to distinguish each interface implementation of one interface.

Sensor input interfaces define a profile for uniqueness of interface versioning (11.2.3).

7 Object level

7.1 General

The object level interfaces (OLI) originate from several sensors. Therefore, at object level, the term sensor cluster is always used, even if a single sensor is serving the interface. The OLIs provide recognised real-world entities and recognition is based on historical data. Recognised objects are tracked and filtered over time. Model-based algorithms classify these objects and rate them with an existence probability value. The sensor cluster refers its recognised objects to one of two predefined vehicle coordinate systems [see signal “Vehicle coordinate system type – header” (A.1.20)]. The properties, and therefore the signals of the recognised objects, are generally independent of the sensor cluster’s sensor technology. Additional properties of the recognised objects can depend on the sensor technology and complement the object description of the interface. These sensor technology dependent signals are grouped in LSGs for each sensor technology.

To link recognised objects with non-OLI interface entities of the sensor cluster or its sensors, each recognised object has a unique signal’s “Object ID” (A.2.2) value that does not change over time and which is referenced by these entities. The signal’s “Object ID” (A.2.2) value can only be reused after the object has not been recognised for a defined amount of time and the sensor cluster and the fusion unit are therefore certain that the object is no longer visible. Specifically, a reidentification of a previous recognised object with the same signal’s “Object ID” (A.2.2) value, after it has not been visible to the sensor cluster for some time, is not guaranteed.

To link recognised objects of the sensor cluster’s OLIs, the set of related objects use the same consistent unique signal’s “Object grouping ID” (A.2.3) value to link all related objects. One dedicated value may indicate that the object is not linked to another object.

The object recognition algorithm is based on multiple measurements of the sensor cluster and is updated over time. For this reason, the entire object properties are not assigned to the time of one single measurement, but to an estimated point in time at which the properties of the interface’s objects fit together consistently. The object existence probability relates to this consistent set of estimated signals (tracked or predicted quantity values) of the object at this estimated point in time.

On object level the following interfaces are available:

- Potentially moving object interface (7.3);
- Road object interface (7.4);
- Static object interface (7.5);
- Free space area object interface (7.6).

7.2 Generic object level interface

Table 6 defines the generic structure of an OLI.

Table 6 — Generic object level interface

Structure	Multiplicity	Option
Generic object level header (7.2.1)	1	
Generic object level entity (7.2.2), specifically an individual object level entity list	Multiple	Size type: dynamic/fixed

Each OLI's individual object level entity list can also be a set of different, specific object type lists [for example, for subclause "Road object interface" (7.4) there are two entity lists for the object types: road marking and road boundary and a single entity for the object type: road surface].

7.2.1 Generic object level header

Table 7 defines the interface header in subclause "Generic object level header" (7.2.1) and the changes due to the adaptation in comparison to the generic interface header in subclause "Generic interface header" (6.2).

Table 7 — Specific signal grouping: Generic object level header

Signal	RL signal	Option
Interface version ID {major, minor, patch} (A.1.1)	Mandatory	Profile: Uniqueness of interface versioning (6.4)
Interface ID (A.1.2)	Optional	Profile: Uniqueness of interface versioning (6.4) Default enumerators (B.1.7)
Number of valid serving sensors (A.1.3)	Mandatory	Profile: Uniqueness of interface versioning (6.4) Optimise LL (B.1.1) Alternative UTP (B.1.6) Key: Sensor ID (A.1.4)
 Sensor ID (A.1.4)	Mandatory	Profile: Uniqueness of interface versioning (6.4) Alternative VRO (B.1.3)
Time stamp – prediction (A.1.5.1)	Mandatory	
Cycle counter (A.1.6.1)	Optional	Redundancy (B.1.2) Signal: Time stamp – prediction (A.1.5.1)
Interface cycle time (A.1.7)	Optional	
Interface cycle time – variation (A.1.8)	Optional	
Data qualifier (A.1.9)	Mandatory	Default enumerators (B.1.7)
OLI specific header extensions		
Recognised <...> – capability (A.1.10)	Optional	
Recognised <...> – status (A.1.11)	Optional	Redundancy (B.1.2) Signal: Recognised <...> – capability (A.1.10) Default enumerators (B.1.7)
Number of valid <...> (A.1.12)	Mandatory	Optimise LL (B.1.1)

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- Interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- Interface: SHII; LSG “Calibration” (see Table 53);
- Interface: SHII; LSG “Sensor cluster” (see Table 53).

7.2.2 Generic object level entity

Table 8 defines the generic signal grouping “Generic object level entity status” for each OLI object type, except of road surface.

Table 8 — Specific signal grouping: Generic object level entity status

Signal	RL signal	Option
Existence probability – object level (A.2.1)	Mandatory	
Object ID (A.2.2)	Mandatory	Alternative A2I (B.3.2)
Object grouping ID (A.2.3)	Optional	
Age (A.2.4)	Mandatory	
Number of valid observations – object level (A.2.5)	Optional	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Time stamp reference – object level (A.2.6) * n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)]
 Time stamp reference – object level (A.2.6)	Mandatory	Alternative VRO (B.1.3)
 Observation status – object level (A.2.7)	Mandatory	Redundancy (B.1.2) Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1) Default enumerators (B.1.7)
Track quality (A.2.8)	Optional	
Measurement status – object level (A.2.9)	Mandatory	Default enumerators (B.1.7)

Each interface type adds individual properties to the interface entity types. Geometric information of the object references the vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)]. Detections and features of the sensor cluster and its sensors shall reference objects of the OLIs with the signal “Object ID” (A.2.2). Potentially connected objects of a sensor cluster shall be linked together, and the sensor cluster shall use the same signal’s “Object grouping ID” (A.2.3) value for all objects of this connected object group.

7.3 Potentially moving object interface

At object level, the potentially moving object interface (PMOI) contains the list of objects that could potentially move and are relevant for driving situations. This includes all objects that currently move, have moved (see Figure 11) and that are not definitely static three-dimensional structures. Table 9 defines the logical structure of the potentially moving object interface. The object interface defines sets of signals for more detailed potentially moving object types (for example, lights, persons).

Table 9 — Potentially moving object interface

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
Information: interface	M	Interface version ID {major, minor, patch} (A.1.1)	M	Profile: Uniqueness of interface versioning (6.4)
		Interface ID (A.1.2)	O	Profile: Uniqueness of interface versioning (6.4) Default enumerators (B.1.7)
		Number of valid serving sensors (A.1.3)	M	Profile: Uniqueness of interface versioning (6.4) Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Sensor ID (A.1.4)
		Size type: dynamic/fixed Size #: Number of valid serving sensors (A.1.3)		
		Sensor ID (A.1.4)	M	Profile: Uniqueness of interface versioning (6.4) Alternative VRO (B.1.3)
		Time stamp – prediction (A.1.5.1)	M	
		Cycle counter (A.1.6.1)	O	Redundancy (B.1.2) Signal: Time stamp – prediction (A.1.5.1)
		Interface cycle time (A.1.7)	O	
		Interface cycle time – variation (A.1.8)	O	
		Data qualifier (A.1.9)	M	Default enumerators (B.1.7)
		Motion type (A.1.13)	M	Profile: Motion type (7.3.3) Default enumerators (B.1.7)

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
Potentially moving objects	M	Recognised potentially moving objects – capability (A.1.10.1)	O	
		Recognised potentially moving objects – status (A.1.11.1)	O	Redundancy (B.1.2) Signal: Recognised potentially moving objects – capability (A.1.10.1) Default enumerators (B.1.7)
		Number of valid potentially moving objects (A.1.12.1)	M	Optimise LL (B.1.1)
		Size type: dynamic/fixed Size #: Number of valid potentially moving objects (A.1.12.1)		
 Potentially moving object status	M	Existence probability – object level (A.2.1)	M	
		Object ID (A.2.2)	M	Alternative A2I (B.3.2)
		Object grouping ID (A.2.3)	O	
		Age (A.2.4)	M	
		Number of valid observations – object level (A.2.5)	O	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Time stamp reference – object level (A.2.6) * n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)]
		Size type: dynamic/fixed Size #: Number of valid observations – object level (A.2.5)		
		Time stamp reference – object level (A.2.6)	M	Alternative VRO (B.1.3)

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
		Observation status – object level (A.2.7)	M	Redundancy (B.1.2) Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1) Default enumerators (B.1.7)
		Track quality (A.2.8)	O	
		Measurement status – object level (A.2.9)	M	Default enumerators (B.1.7)
☐ Potentially moving object information	M	Tracking motion model (A.2.10)	O	Default enumerators (B.1.7)
		Number of valid potentially moving object classifications (A.2.11)	M	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Potentially moving object classification type (A.2.12)
		Size type: dynamic/fixed Size #: Number of valid potentially moving object classifications (A.2.11)		
		Potentially moving object classification type (A.2.12)	M	Alternative VRO (B.1.3) Default enumerators (B.1.7)
		Potentially moving object classification type – confidence (A.2.13)	M	
☐ Potentially moving object position	M	Reference point (A.2.14)	O	Default enumerators (B.1.7)
		Position – object level {x, y, z} (A.2.15)	{M, M, O}	Profile: Motion state vector (7.3.4)
		Position – object level {x, y, z} – error (A.2.16)	{M, M, O}	Profile: Motion state vector (7.3.4) Implementation EM (B.4.1)

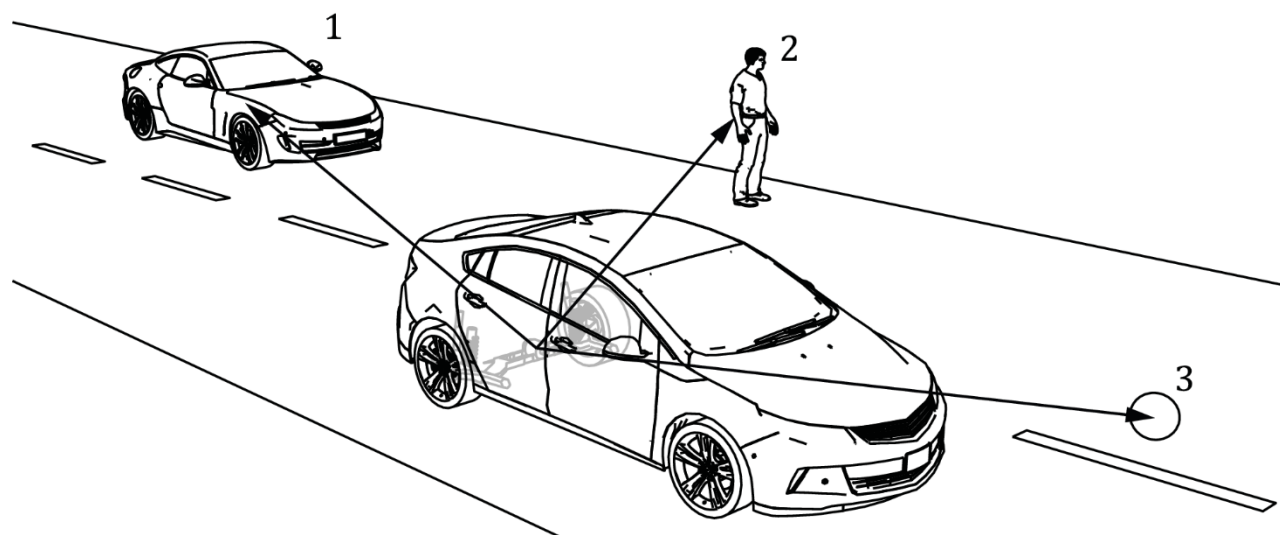
LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
		Orientation {yaw, pitch, roll} (A.2.17)	{C (B.2.1) Relevant: camera, O, O}	Profile: Motion state vector (7.3.4)
		Orientation {yaw, pitch, roll} – error (A.2.18)	O	Profile: Motion state vector (7.3.4) Implementation EM (B.4.1)
		Road level (A.2.19)	O	Redundancy (B.1.2) Signal: {z} of Position – object level {x, y, z} (A.2.15) Default enumerators (B.1.7)
 Potentially moving object bounding box	C (B.2.1) Relevant: camera	Bounding box extent {length, width, height} (A.2.20)	{M, M, O}	
		Bounding box extent {length, width, height} – error (A.2.21)	O	Implementation EM (B.4.1)
		Clearance of the object {height} (A.2.22)	O	
		Included geometric structures (A.2.23)	O	Default enumerators (B.1.7)
 Potentially moving object dynamics	M	Velocity {x, y, z} – object level (A.2.24)	{C, C, C (B.2.2) Signal: Position {z} (A.2.15)} (B.2.1) Relevant: except ultrasonic	Profile: Motion type (7.3.3) Profile: Motion state vector (7.3.4)
		Velocity {x, y, z} – object level – error (A.2.25)	O	Profile: Motion type (7.3.3) Profile: Motion state vector (7.3.4) Implementation EM (B.4.1)
		Acceleration {x, y, z} (A.2.26)	O	Profile: Motion type (7.3.3)
		Acceleration {x, y, z} – error (A.2.27)	O	Profile: Motion type (7.3.3) Implementation EM (B.4.1)

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
		Instantaneous centre of rotation {x, y} (A.2.28)	C (B.2.2) Signal: Rotation rate at instantaneous centre of rotation {yaw} (A.2.30)	
		Instantaneous centre of rotation {x, y} – error (A.2.29)	O	Implementation EM (B.4.1)
		Rotation rate at instantaneous centre of rotation {yaw} (A.2.30)	O	Profile: Motion type (7.3.3) Profile: Motion state vector (7.3.4)
		Rotation rate at instantaneous centre of rotation {yaw} – error (A.2.31)	O	Profile: Motion type (7.3.3) Profile: Motion state vector (7.3.4) Implementation EM (B.4.1)
		Movement status (A.2.32)	O	Default enumerators (B.1.7)
☐ Potentially moving object's lights	C (B.2.1) Relevant: camera	Number of valid lights (A.2.33)	M	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Light type (A.2.34)
		Size type: dynamic/fixed Size #: Number of valid lights (A.2.33)		
		Light type (A.2.34)	M	Alternative VRO (B.1.3) Default enumerators (B.1.7)
		Light status (A.2.35)	M	Default enumerators (B.1.7)
☐ Potentially moving object: persons	O	Number of valid persons (A.2.36)	M	Optimise LL (B.1.1)
		Size type: dynamic/fixed Size #: Number of valid persons (A.2.36)		
		Number of valid person type roles (A.2.37)	M	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Person type role (A.2.38)

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
		Size type: dynamic/fixed Size #: Number of valid person type roles (A.2.37)		
		Person type role (A.2.38)	M	Alternative VRO (B.1.3) Default enumerators (B.1.7)
		Person type role – confidence (A.2.39)	M	
		Number of valid person gestures (A.2.40)	O	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Person gesture indication type (A.2.41)
		Size type: dynamic/fixed Size #: Number of valid person gestures (A.2.40)		
		Person gesture indication type (A.2.41)	M	Alternative VRO (B.1.3) Default enumerators (B.1.7)
		Person gesture indication type – confidence (A.2.42)	O	
		Person gesture direction type (A.2.43)	O	Default enumerators (B.1.7)
		Person gesture direction type – confidence (A.2.44)	O	
		Number of valid person's body part poses (A.2.45)	M	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Person body part (A.2.46)
		Size type: dynamic/fixed Size #: Number of valid person's body part poses (A.2.45)		
		Person body part (A.2.46)	M	Alternative VRO (B.1.3) Default enumerators (B.1.7)

LSG	RL LSG M/C/O	Signal	RL signal M/C/O	Option
		Person body part orientation {yaw, pitch, roll} (A.2.47)	M	
		Person body part orientation {yaw, pitch, roll} – error (A.2.48)	0	Implementation EM (B.4.1)
		Person body part origin {x, y, z} (A.2.49)	0	
		Person body part origin {x, y, z} – error (A.2.50)	0	Implementation EM (B.4.1)
		Eye state (A.2.51)	0	Default enumerators (B.1.7)
		Eye state – confidence (A.2.52)	0	
		Viewing fixation time (A.2.53)	0	
		Viewing fixation time – error (A.2.54)	0	Implementation EM (B.4.1)
		Viewing fixation direction (A.2.55)	0	Default enumerators (B.1.7)
		Viewing fixation direction – confidence (A.2.56)	0	
		Action type (A.2.57)	0	Default enumerators (B.1.7)
		Action type – confidence (A.2.58)	0	
		Number of valid person accessories (A.2.59)	0	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Person accessory classification type (A.2.60)
		Size type: dynamic/fixed Size #: Number of valid person accessories (A.2.59)		
		Person accessory classification type (A.2.60)	M	Alternative VRO (B.1.3) Default enumerators (B.1.7)

LSG	RL LSG M/C/O	Signal			RL signal M/C/O	Option
				Person accessory classification type – confidence (A.2.61)	0	
				Accessory connection to person type (A.2.62)	0	Default enumerators (B.1.7)
				Included geometric structure of accessory (A.2.63)	0	Default enumerators (B.1.7)
☐ Lane related information	0			Object lane association (A.2.64)	M	Default enumerators (B.1.7)
				Angle between object edge and lane {left edge right lane, right edge left lane} (A.2.65)	0	
				Angle between object edge and lane {left edge right lane, right edge left lane} – error (A.2.66)	0	Implementation EM (B.4.1)
				Percentage side lane {left, right} (A.2.67)	0	
☐ Motion related information	0			Angular position {azimuth} (A.2.68)	M	
				Angular velocity {azimuth} (A.2.69)	M	
☐ Camera sensor technology specific	C (B.2.1) Relevant: camera			Scale change – object level (A.2.70)	M	
☐ Radar sensor technology specific	C (B.2.1) Relevant: radar			Entity radar cross section (A.2.71)	M	
☐ Lidar sensor technology specific	C (B.2.1) Relevant: lidar			Entity lidar reflectivity (A.2.72)	M	

**Key**

- 1 vehicle
- 2 pedestrian
- 3 potentially moving object – classification unknown, for example, “Potentially moving object classification type” (A.2.12) with default enumerator “PMOCT_Unknown”

Figure 11 — Example for potentially moving objects**7.3.1 Potentially moving object header**

Table 10 defines the interface header in subclause “Potentially moving object header” (7.3.1) and the changes due to the adaptation in comparison to the generic interface header in 7.2.1. The header of the PMOI can contain a list of valid potentially moving object entities (see 7.3.2)

Table 10 — Specific signal grouping: Potentially moving object header

Signal	RL signal	Option
Interface version ID {major, minor, patch} (A.1.1)	Mandatory	Profile: Uniqueness of interface versioning (6.4)
Interface ID (A.1.2)	Optional	Profile: Uniqueness of interface versioning (6.4) Default enumerators (B.1.7)
Number of valid serving sensors (A.1.3)	Mandatory	Profile: Uniqueness of interface versioning (6.4) Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Sensor ID (A.1.4)
 Sensor ID (A.1.4)	Mandatory	Profile: Uniqueness of interface versioning (6.4) Alternative VRO (B.1.3)
Time stamp – prediction (A.1.5.1)	Mandatory	
Cycle counter (A.1.6.1)	Optional	Redundancy (B.1.2) Signal: Time stamp – prediction (A.1.5.1)
Interface cycle time (A.1.7)	Optional	
Interface cycle time – variation (A.1.8)	Optional	
Data qualifier (A.1.9)	Mandatory	Default enumerators (B.1.7)

Signal	RL signal	Option
Motion type (A.1.13)	Mandatory	Profile: Motion type (7.3.3) Default enumerators (B.1.7)
Recognised potentially moving objects – capability (A.1.10.1)	Optional	
Recognised potentially moving objects – status (A.1.11.1)	Optional	Redundancy (B.1.2) Signal: Recognised potentially moving objects – capability (A.1.10.1) Default enumerators (B.1.7)
Number of valid potentially moving objects (A.1.12.1)	Mandatory	Optimise LL (B.1.1)

The following LSGs, which normally shall be provided by the sensor cluster in another interface, may be added to this header, if, for example, the other interface is not implemented by the sensor cluster (see B.3.3):

- interface: SPI; LSG “Information: vehicle coordinate system” (see Table 51);
- interface: SHII; LSG “Calibration” (see Table 53);
- interface: SHII; LSG “Sensor cluster” (see Table 53).

7.3.2 Potentially moving object entity

Each potentially moving object describes a recognised real-world object and consists of several LSGs. Each object, which may move, even if it might have not moved yet since first recognition of the real-world object, is an entity of potentially moving objects.

A potentially moving object is described by the following LSGs for potentially moving objects with attributes for different parts (for example, vehicle, pedestrian).

- **Status:** the status describes general information of the tracked potentially moving object. This information is based on history of the object tracking. Table 11 defines the signal grouping “Potentially moving object entity status”. It defines this status LSG of the potentially moving object and redefines the signal grouping “Generic object level entity status” (see Table 8).
- **Information:** the information describes properties such as the confidence of an object type.
- **Position:** the position describes the geometric position of the object relative to the vehicle coordinate system [see signal “Vehicle coordinate system type – header” (A.1.20)].
- **Bounding box:** the object bounding box describes the 3D rectangular hull that encloses the recognised object.
- **Dynamics:** the dynamics describe the motion information of the tracked potentially moving object with respect to the motion type defined by the signal “Motion type” (A.1.13) [see Profile: Motion type (7.3.3)].
- **Lights:** the lights describe the status of the tracked potentially moving object’s lights.
- **Persons:** the persons describe recognised person type, its gestures and poses of the tracked persons if the tracked potentially moving object may be a pedestrian or include persons (for example, the driver of a vehicle).

- **Lane related information:** the lane related information describes the tracked potentially moving object in relation with real-world lanes.
- **Motion related information:** the motion related information describes additional motion information of the tracked potentially moving object.
- **Camera sensor technology specific:** the camera sensor technology specific data describe dynamic information in the camera projection plane.
- **Radar sensor technology specific:** the radar sensor technology specific data describe different surface and material properties of the potentially moving object with respect to the radar.
- **Lidar sensor technology specific:** the lidar sensor technology specific data describes different surface and material properties of the potentially moving object with respect to the lidar (for example, reflection strength of a surface).

Table 11 — Specific signal grouping: Potentially moving object entity status

Signal	RL signal	Option
Existence probability – object level (A.2.1)	Mandatory	
Object ID (A.2.2)	Mandatory	Alternative A2I (B.3.2)
Object grouping ID (A.2.3)	Optional	
Age (A.2.4)	Mandatory	
Number of valid observations – object level (A.2.5)	Optional	Optimise LL (B.1.1) Alternative UTL (B.1.6) Key: Time stamp reference – object level (A.2.6) * n recent messages of the interface [see “Time stamp reference – object level” (A.2.6)]
 Time stamp reference – object level (A.2.6)	Mandatory	Alternative VRO (B.1.3)
 Observation status – object level (A.2.7)	Mandatory	Redundancy (B.1.2) Signal: Object ID (A.2.2), Time stamp – prediction (A.1.5.1) Default enumerators (B.1.7)
Track quality (A.2.8)	Optional	
Measurement status – object level (A.2.9)	Mandatory	Default enumerators (B.1.7)

Potentially moving objects or other object entities on object level may be linked together (for example, a truck with a trailer or a construction site vehicle with a traffic sign at the rear of the vehicle) by using the same signal's “Object grouping ID” (A.2.3) value in all connected objects.

7.3.3 Profile: Motion type

The signal “Motion type” (A.1.13) defines the interpretation of the signal “Velocity {x, y, z} – object level” (A.2.24), “Velocity {x, y, z} – object level – error” (A.2.25), “Acceleration {x, y, z}” (A.2.26), “Acceleration {x, y, z} – error” (A.2.27), “Rotation rate at instantaneous centre of rotation {yaw}” (A.2.30) and “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31) values.

7.3.4 Profile: Motion state vector

A potentially moving object entity provides a motion state vector. The sensor may provide a full or enhanced error model according to B.4.1.3.5. For example, the motion state vector of the sensor is provided together with an enhanced error model {“Position – object level {x, y, z}” (A.2.15), “Velocity {x, y, z} – object level” (A.2.24), “Orientation {yaw, pitch, roll}” (A.2.17) and “Rotation rate at instantaneous centre of rotation {yaw}” (A.2.30)}.

The resulting error model based on variances, covariances and cross-covariances (see B.4.1.3.4 and B.4.1.3.5) may provide a symmetric nD matrix:

$$\begin{pmatrix} \text{Position – error (A. 2.16)} & \text{Position x Velocity – error} & \text{Position x Orientation – error} & \text{Position x Rotation rate – error} \\ \text{Velocity x Position – error} & \text{Velocity – error (A. 2.25)} & \text{Velocity x Orientation – error} & \text{Velocity x Rotation rate – error} \\ \text{Orientation x Position – error} & \text{Orientation x Velocity – error} & \text{Orientation – error(A. 2.17)} & \text{Orientation x Rotation rate – error} \\ \text{Rotation rate x Position – error} & \text{Rotation rate x Velocity – error} & \text{Rotation rate x Orientation – error} & \text{Rotation rate – error (A. 2.30)} \end{pmatrix}$$

In addition to existing error signals “Position – object level {x, y, z} – error” (A.2.16), “Velocity {x, y, z} – object level – error” (A.2.25), “Orientation {yaw, pitch, roll} – error” (A.2.18) and “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31); the additional signals may be provided as sets of cross-covariances. All covariances of the motion state vector error model are symmetric and therefore additional optimisation methods are applicable; this means only half of non-diagonal elements need to be provided.

For example, the LSG “Potentially moving object motion state vector cross-covariances” may provide, for example, “Position {x, y, z} x Velocity {x, y, z} – object level – error”, “Position {x, y, z} x Orientation {yaw, pitch, roll} – object level – error”. The redundant symmetrical signals for the cross-covariances: “Velocity {x, y, z} x Position {x, y, z} – object level – error”, “Orientation {yaw, pitch, roll} x Position {x, y, z} – object level – error” may be skipped. Similar, the symmetric covariances of the signals “Position – object level {x, y, z} – error” (A.2.16), “Velocity {x, y, z} – object level – error” (A.2.25), “Orientation {yaw, pitch, roll} – error” (A.2.18) and “Rotation rate at instantaneous centre of rotation {yaw} – error” (A.2.31) may not transmit redundant values (see optimisation method B.1.5).

Table 12 provides a subset of the PMOI with the relevant signals for the motion state vector of this profile and the additional signals in the LSG “Potentially moving object motion state vector cross-covariances”.

Table 12 — Signal grouping: Exemplary motion state vector cross-covariances LSG

 Potentially moving object position	M	Position – object level {x, y, z} (A.2.15)	{M, M, O}	Profile: Motion state vector (7.3.4)
		Position – object level {x, y, z} – error (A.2.16)	{M, M, O}	Profile: Motion state vector (7.3.4) ...
		Orientation {yaw, pitch, roll} (A.2.17)	{C (B.2.1) Relevant: camera, O, O}	Profile: Motion state vector (7.3.4)
		Orientation {yaw, pitch, roll} – error (A.2.18)	O	Profile: Motion state vector (7.3.4) ...
		...	...	...

Potentially moving object dynamics	M		Velocity {x, y, z} – object level (A.2.24)	{C, C, C (B.2.2) Signal: Position {z} (A.2.15)} (B.2.1) Relevant: except ultrasonic	Profile: Motion state vector (7.3.4) ...
			Velocity {x, y, z} – object level – error (A.2.25)	0	Profile: Motion state vector (7.3.4) ...
			Rotation rate at instantaneous centre of rotation {yaw} (A.2.30)	0	Profile: Motion state vector (7.3.4) ...
			Rotation rate at instantaneous centre of rotation {yaw} – error (A.2.31)	0	Profile: Motion state vector (7.3.4) ...
			...	...	...
Potentially moving object motion state vector cross-covariances	0		Position {x, y, z} x Velocity {x, y, z} – object level – error	0 ^a	Implementation EM (B.4.1)
			Position {x, y, z} x Orientation {yaw, pitch, roll} – object level – error	0	Implementation EM (B.4.1)
			Position {x, y, z} x Rotation rate at instantaneous centre of rotation {yaw} – object level – error	0	Implementation EM (B.4.1)
			Velocity {x, y, z} x Orientation {yaw, pitch, roll} – object level – error	0	Implementation EM (B.4.1)
			Velocity {x, y, z} x Rotation rate at instantaneous centre of rotation {yaw} – object level – error	0	Implementation EM (B.4.1)
			Orientation {yaw, pitch, roll} x Rotation rate at instantaneous centre of rotation {yaw} – object level – error	0	Implementation EM (B.4.1)
^a The signal’s RL is optional by definition of B.4.1.3.5 but for this profile (7.3.4) it should be defined as mandatory during the system design phase so that at least one signal is mandatory in the LSG.					

Additionally, in most cases the description of the signals in planar space (XY-plane) is sufficient. Therefore, below covariances may either be only 2D (position, velocity) or 1D (orientation, rotation rate).

Optimisation method may be also applied by selection of most relevant covariance matrix elements (or cross-covariance matrix elements).

Missing covariance values shall be substituted by suitable values in the fusion tracking system to guarantee a stabile tracking.

7.4 Road object interface

The road object interface (RDOI) consists of a general road surface information and two types of objects: road markings and road boundaries (see Figure 12). Table 13 defines the logical structure of the road object interface.