Document Title	Requirements on SW-C and System Modeling
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	267

Document Status	published
Part of AUTOSAR Standard	Classic Platform
Part of Standard Release	R25-11

Document Change History

Date	Release	Changed by	Description
2025-11-27	R25-11	AUTOSAR Release Management	• Editorial changes
2024-11-27	R24-11	AUTOSAR Release Management	• No content changes
2023-11-23	R23-11	AUTOSAR Release Management	• No content changes
2022-11-24	R22-11	AUTOSAR Release Management	• No content changes
2021-11-25	R21-11	AUTOSAR Release Management	• No content changes
2020-11-30	R20-11	AUTOSAR Release Management	• No content changes
2019-11-28	R19-11	AUTOSAR Release Management	• No content changes • Changed Document Status from Final to published
2018-10-31	4.4.0	AUTOSAR Release Management	• Editorial changes
2017-12-08	4.3.1	AUTOSAR Release Management	• Editorial changes

△

2016-11-30	4.3.0	AUTOSAR Release Management	• Editorial changes
2014-10-31	4.2.1	AUTOSAR Release Management	• Editorial changes
2013-03-15	4.1.1	AUTOSAR Administration	• New format for Requirement Tables adopted to fill the goal of complete traceability among AUTOSAR official documents. • Official Requirements Identification procedure introduced according to the Standardization Template. • Naming Convention Requirements extended to cover Long Names domain.
2011-12-22	4.0.3	AUTOSAR Administration	• Renaming of tags, accordingly with one Methodology group
2010-02-02	3.1.4	AUTOSAR Administration	• The following requirements have been removed: MG015, MG050 • The following requirements have been added: MG059, MG060, MG061 • Length limitation of short names in MG014 set to 128 characters • Legal disclaimer revised
2008-08-13	3.1.1	AUTOSAR Administration	• Legal disclaimer revised
2007-12-21	3.0.1	AUTOSAR Administration	• Initial Release

Disclaimer

Table of Contents

1 Scope of Document

This document defines general rules and formats for requirements specification within AUTOSAR. It shall be used as a basis for each requirements document.

1.1 Terminology

Identifiable: any model element that can have a set of attributes. Please refer to the AUTOSAR Meta Model for further and detailed explanation of this term ("Instances of this class can be referred to by their identifier (while adhering to namespace borders))". Use this term instead of "element" , "data name", etc. unless a requirement is applicable to a specific Meta Model Identifiable such as Port, Data Type, etc..

ARElement: As defined into AUTOSAR Meta Model: "An element that can be defined stand-alone, i.e. without being part of another element (except for packages of course).

Opposed to packages, the elements are closed sets, i.e. that in a file based description, one ARElement needs to be described completely and cannot be extended or completed by another file".

ARPackage: As defined into AUTOSAR Meta Model: "AUTOSAR package, allowing to create top level packages to structure the contained ARElements.

ARPackages are open sets, which means that in a file based description system, multiple files can be used to partially describe the contents of a package.

This is an extended version of MSR's SW-SYSTEM".

2 Conventions to be used

2.1 Document Conventions

The representation of requirements in AUTOSAR documents follows the table specified in [TPS_STDT_00078], see [1, Standardization Template].

The verbal forms for the expression of obligation specified in [TPS_STDT_00053] shall be used to indicate requirements, see [1, Standardization Template].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as follows.

Note that the requirement level of the document in which they are used modifies the force of these words.

- MUST: This word, or the adjective "LEGALLY REQUIRED", means that the definition is an absolute requirement of the specification due to legal issues.
- MUST NOT: This phrase, or the phrase "MUST NOT", means that the definition is an absolute prohibition of the specification due to legal issues.
- SHALL: This phrase, or the adjective "REQUIRED", means that the definition is an absolute requirement of the specification.
- SHALL NOT: This phrase means that the definition is an absolute prohibition of the specification.
- SHOULD: This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- SHOULD NOT: This phrase, or the phrase "NOT RECOMMENDED", means that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- MAY: This word, or the adjective "OPTIONAL", means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item.

An implementation, which does not include a particular option, SHALL be prepared to interoperate with another implementation, which does include the option, though perhaps with reduced functionality. In the same vein an implementation, which does include a particular option, SHALL be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides).

3 Acronyms and abbreviations

The glossary below includes acronyms and abbreviations relevant to SW-C and System Modeling that are not included in the AUTOSAR Glossary [2].

Abbreviation / Acronym:	Description:
AR	AUTOSAR
ECU	Electronic Control Unit
HMI	Human Machine Interface
MISRA	Motor Industry Software Reliability Association
RTE	Real Time Environment
SW-C	Software Component
WP	Work Package

Table 3.1: Acronyms and abbreviations used in the scope of this Document

4 Requirements Specification

This chapter describes all requirements driving the work to define the SWCModeling.

4.1 Naming Convention Requirements

4.1.1 [RS_SWMG_00001] Distinguish Standardized vs not standardized model elements of type ARElement

[RS_SWMG_00001] Distinguish Standardized vs not standardized model elements of type ARElement ⌈

Description:	The naming convention shall provide an attribute to distinguish between standardized and non standardized AUTOSAR model elements of type ARElement.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	Model elements are specified in the documents AUTOSAR SW-C Template, ECU-Resource Template, and System Template. Possible implementation of this requirement will be: • prefix of the model element name • suffix of the model element name • packages for standardized components (not applicable to Ports) this can be a solution to the requirement.

⌋

4.1.2 [RS_SWMG_00002] Name shall reflect the purpose of the model element

[RS_SWMG_00002] Name shall reflect the purpose of the model element ⌈

Description:	The naming convention should allow to define names that give at a glance an idea of the purpose of the element.
Rationale:	It is necessary to avoid the creation of identical names for elements having a different purpose. For example data flow property (such as Request and Status) is needed to differentiate names that otherwise would be equal.
Use Case:	Identify whether an interface and/or data element is a command, a status, a request, a value, etc.. Example: PGearEngaged and PGearRequest
Dependencies:	[RS_SWMG_00005] Easy creation of names

▽

△

Supporting Material:	Source : Internal document of Body Domain, AUTOSAR_CentralLocking_ ApplicationInterfaces.doc: Semantic of keywords (e.g. "operation") in the interface/ data element names: • Cmd(command) do/activate something (e. g. from Master to Actuator) • Req(request) demand to do/activate something (e. g. from Sensor to Master) • Sta(status) get functional status information • Hmi user request (e.g. from driver via switch, touch screen, ...) • Dis(display) feedback status for driver information display • Err(failure) operative/defective failure feedback (from actuator to master)

⌋

4.1.3 [RS_SWMG_00005] Easy creation of names

[RS_SWMG_00005] Easy creation of names ⌈

Description:	–
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	Possible solution: Model Element Names are composed by arranging predefined keywords in a predefined order. This will lead to definition of a set of predefined keywords but may conflict with the high number of required keywords/catchwords and the need to keep names short for use cases in function development, documentation calibration and to support compiler specification

⌋

4.1.4 [RS_SWMG_00006] Model Elements names shall be self-explanatory

[RS_SWMG_00006] Model Elements names shall be self-explanatory ⌈

Description:	–
Rationale:	–
Use Case:	E.g. data-element, ports, interfaces, composition, etc.
Dependencies:	–
Supporting Material:	–

⌋

4.1.5 [RS_SWMG_00007] Distinguish model elements of different model element suppliers

[RS_SWMG_00007] Distinguish model elements of different model element suppliers ⌈

Description:	The Modeling Guide should define an attribute to distinguish between model elements of different model element suppliers. This is only applicable for non standardized model elements
Rationale:	Avoid merge conflicts if software component descriptions of different suppliers are joined to a system model. Brand responsibly.
Use Case:	Usage of non standardized elements within an AUTOSAR package. If errors appear it is request to trace the SW-C supplier responsible for that.
Dependencies:	If solved by Naming Convention: Not applicable for ModeDeclarationGroup Prototype, DataElementPrototype, CalprmElementPrototype, Operation Prototype, ArgumentPrototype due to the required uniformity of names as precondition for connectability of ports
Supporting Material:	Could either be done by naming convention or by usage of other model elements like AdminData.

⌋

4.1.6 [RS_SWMG_00010] Model Element Names shall follow semantic rules

[RS_SWMG_00010] Model Element Names shall follow semantic rules ⌈

Description:	–
Rationale:	By doing so, the compliancy to the naming convention would be verifiable by name checkers or name creator tools.
Use Case:	–
Dependencies:	[RS_SWMG_00005] Easy creation of names [RS_SWMG_00048] Easy lookup of names in databases
Supporting Material:	Modeling Guide, AI Specification

⌋

4.1.7 [RS_SWMG_00011] Model Element Names are composed by arranging standardized keywords

[RS_SWMG_00011] Model Element Names are composed by arranging standardized keywords ⌈

Description:	–
Rationale:	By doing so, the compliancy to the naming convention would be verifiable by name checkers or name creator tools. Names length restriction can lead to not comprehensible names if keyword and acronyms are not standardized.
Use Case:	–
Dependencies:	[RS_SWMG_00005] Easy creation of names [RS_SWMG_00034] Usage of Unique Keywords
Supporting Material:	Modeling Guide, AI Specification

⌋

4.1.8 [RS_SWMG_00012] Semantic of Model Element Names shall allow variable number of keywords

[RS_SWMG_00012] Semantic of Model Element Names shall allow variable number of keywords ⌈

Description:	The number of composed keyword shall be dependent from the need of explanation.
Rationale:	Created names shall be simple as possible but complex as required.
Use Case:	–
Dependencies:	[RS_SWMG_00005] Easy creation of names [RS_SWMG_00010] Model Element Names shall follow semantic rules [RS_SWMG_00034] Usage of Unique Keywords
Supporting Material:	Modeling Guide An example of solution: Eng_tqCluReqDrvSlow -> Engine Torque at Clutch Slow Request Veh_v -> Vehicle Speed

⌋

4.1.9 [RS_SWMG_00014] Length restriction for short names of Identifiable

[RS_SWMG_00014] Length restriction for short names of Identifiable ⌈

Description:	Short Names of Identifiable shall be restricted to total length of 128 characters.
Rationale:	Short Names partly used for the creation of C Language Names. These created names shall have a predictable maximum length to avoid tool problems. (Even if this length will greater than MISRA guideline recommendation it shall not be infinite.)
Use Case:	–
Dependencies:	–
Supporting Material:	A rule restricting the number of characters to 128 is already present in the Meta Model. [a-zA-Z][a-zA-Z_0-9]{0-127}

⌋

4.1.10 [RS_SWMG_00016] Names shall allow to indicate if the value is a direct measurement or a conditioned value

[RS_SWMG_00016] Names shall allow to indicate if the value is a direct measurement or a conditioned value ⌈

Description:	Names should indicate if the value is measured from sensors (maybe of-set compensated and/or filtered) or calculated/estimated from a set of information or model based.
Rationale:	–
Use Case:	Sensor SW-C outputting a measured physical value and feeding it to another SW-C in charge of filtering it. In this case the names of data elements, ports, and interface would differ only for a keyword, and the data type could be the same.
Dependencies:	–
Supporting Material:	Possible solution: use a dedicated keyword in the name semantic to indicate such information.

⌋

4.1.11 [RS_SWMG_00017] Names shall follow the ISO 8855 for English naming.

[RS_SWMG_00017] Names shall follow the ISO 8855 for English naming. ⌈

Description:	This standard defines the principle terms of vehicle dynamics applicable (not only) to passenger cars. Definitions are provided in more than one language, only definition in English shall be followed.
Rationale:	–

▽

△

Use Case:	–
Dependencies:	[RS_SWMG_00030] Use English as Standard Language for Names.
Supporting Material:	–

⌋

4.1.12 [RS_SWMG_00030] Use English as Standard Language for Names.

[RS_SWMG_00030] Use English as Standard Language for Names. ⌈

Description:	English language shall be used for names and acronyms.
Rationale:	Internationality and common understanding of names and keywords.
Use Case:	Designers of different nationality will come up with the same solution while defining new names.
Dependencies:	[RS_SWMG_00017] Names shall follow the ISO 8855 for English naming.
Supporting Material:	Naming Convention 1.0 from Powertrain Domain §1.4

⌋

4.1.13 [RS_SWMG_00031] No Architectural Information in Names

[RS_SWMG_00031] No Architectural Information in Names ⌈

Description:	No definition of architectural or implementation information shall be present into names.
Rationale:	To increase the re-usability and decrease the maintenance of standard elements.
Use Case:	Creating different composition of components without changing any element name.
Dependencies:	–
Supporting Material:	Naming Convention 1.0 from Powertrain Domain §1.4

⌋

4.1.14 [RS_SWMG_00034] Usage of Unique Keywords

[RS_SWMG_00034] Usage of Unique Keywords ⌈

Description:	Keywords used to compose Names shall be unique. Double or multiple meaning of keywords is possible unless violation of semantic rules is detected.
Rationale:	–
Use Case:	Automated checking of Names with respect to conformance will be possible.
Dependencies:	[RS_SWMG_00010] Model Element Names shall follow semantic rules [RS_SWMG_00011] Model Element Names are composed by arranging standardized keywords
Supporting Material:	Naming Convention 1.0 from Powertrain Domain §1.4

⌋

4.1.15 [RS_SWMG_00039] Avoid usage of Trailing underscores

[RS_SWMG_00039] Avoid usage of Trailing underscores ⌈

Description:	Names shall not end with an underscore [_] character.
Rationale:	AUTOSAR tool such as RTE generates the name with the "_" to indicate the information flow path across the AR layer. This will facilitate the better understanding for tool generated names and also with limitation to number of characters in the name
Use Case:	–
Dependencies:	–
Supporting Material:	Naming Convention 1.0 from Powertrain Domain §2

⌋

4.1.16 [RS_SWMG_00040] Avoid sequences of underscores characters.

[RS_SWMG_00040] Avoid sequences of underscores characters. ⌈

Description:	Avoid sequences of underscore characters directly after each other [__].
Rationale:	Waste of characters space.
Use Case:	–
Dependencies:	–
Supporting Material:	Naming Convention 1.0 from Powertrain Domain §2

⌋

4.1.17 [RS_SWMG_00041] Do not rely on uppercase/lowercase difference only.

[RS_SWMG_00041] Do not rely on uppercase/lowercase difference only. ⌈

Description:	Avoid distinguish Names only from uppercase/lowercase format
Rationale:	Human user can easily mix up names that differ only for capitalization
Use Case:	–
Dependencies:	–
Supporting Material:	Naming Convention 1.0 from Powertrain Domain §2

⌋

4.1.18 [RS_SWMG_00048] Easy lookup of names in databases

[RS_SWMG_00048] Easy lookup of names in databases ⌈

Description:	–
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	[RS_SWMG_00005] Easy creation of names

⌋

4.1.19 [RS_SWMG_00049] Support Identifiable already present in the MasterTable

[RS_SWMG_00049] Support Identifiable already present in the MasterTable ⌈

Description:	All model element types that are used in the Master Table such as Sender Receiver interfaces, DataElements, DataTypes, Unit, Component Types, etc. shall be supported by the Modelling Rules.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	AI Specification is a place-holder for Identifiable that are listed in that file

⌋

4.1.20 [RS_SWMG_00054] Provide guidelines how to resolve name conflicts

[RS_SWMG_00054] Provide guidelines how to resolve name conflicts ⌈

Description:	The modelling guide should provide guidelines how to resolve name conflicts between related elements.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	One possible implementation of this requirement is the use of prefixes. To define a PrimitiveTypeWithSemantics a CompuMethod definition is also necessary. Using the prefix solution, the names could look like: PrimitiveTypeWithSemantic : Veh_v used for vehicle speed CompuMethode: Compu_Veh_v used for vehicle speed data type Interface If_Veh_v Interface for vehicle speed The prefix solution has the disadvantage of increasing the length of the names and could lead to a violation of [RS_SWMG_00014]. Another possible solution is the use of sub packages.

⌋

4.1.21 [RS_SWMG_00059] There shall be a single set of keywords

[RS_SWMG_00059] There shall be a single set of keywords ⌈

Description:	The modelling guide shall provide a list of standardized keywords
Rationale:	To ensure uniqueness of the naming convention all keywords shall be collected in one keyword list
Use Case:	–
Dependencies:	–
Supporting Material:	A possible solution is to use a separate document as development work-product for keywords and include the finalized list of keywords only when a milestone of the Modelling Guide document is needed. This would save the Modelling Guide from frequent iterations resulting from discussions and evolutions of the keyword list.

⌋

4.1.22 [RS_SWMG_00060] Applicability of Naming Convention

[RS_SWMG_00060] Applicability of Naming Convention ⌈

Description:	Naming convention must be applicable to all vehicle application domains of AUTOSAR
Rationale:	1. In open environments where arbitrary parties are willing to cooperate all parties shall use the same naming convention. 2. If dedicated naming conventions for specific domains or parties would be supported, the acceptance of the conventions would be very low. Many parties would argue, that they need a specific convention for their area.
Use Case:	–
Dependencies:	[RS_SWMG_00002] Name shall reflect the purpose of the model element, [RS_SWMG_00005] Easy creation of names, [RS_SWMG_00006] Model Elements names shall be self-explanatory, [RS_SWMG_00034] Usage of Unique Keywords
Supporting Material:	The global acceptance of the universal naming conventions will take time but should not limit the claim of the standard.

⌋

4.1.23 [RS_SWMG_00061] Naming convention shall be unique

[RS_SWMG_00061] Naming convention shall be unique ⌈

Description:	Naming conventions must state clear and deterministic rules for the creation of names such that names can be uniquely determined from signal characteristics.
Rationale:	1. Support distributed development 2. Avoid definition of redundant signals since different developers will create names by application of the same rules. 3. Avoid misuse of signals. 4. Enable consistency checks and tool-based processing of names. 5. Enhance readability since all developers / users of the names develop the same mindset.
Use Case:	–

▽

△

Dependencies:	[RS_SWMG_00002] Name shall reflect the purpose of the model element, [RS_SWMG_00006] Model Elements names shall be self-explanatory, [RS_SWMG_00010] Model Element Names shall follow semantic rules, [RS_SWMG_00011] Model Element Names are composed by arranging standardized keywords [RS_SWMG_00016] Names shall allow to indicate if the value is a direct measurement or a conditioned value [RS_SWMG_00031] No Architectural Information in Names [RS_SWMG_00034] Usage of Unique Keywords [RS_SWMG_00054] Provide guidelines how to resolve name conflicts [RS_SWMG_00059] There shall be a single set of keywords
Supporting Material:	Idea behind the requirement is that the name for a signal can be uniquely determined from the characteristics of a signal like provider, physical unit, ...

⌋

4.1.24 [RS_SWMG_00062] Naming Convention shall rule Short Names and Long Names construction.

[RS_SWMG_00062] Naming Convention shall rule Short Names and Long Names construction. ⌈

Description:	Naming conventions shall rule short names and long names construction through a clear set of rules and recommendations.
Rationale:	To support clear and understandable short names and long names constructions and encourage elements reusage in AI domain.
Use Case:	–
Dependencies:	[RS_SWMG_00002] Name shall reflect the purpose of the model element, [RS_SWMG_00006] Model Elements names shall be self-explanatory creation of names, [RS_SWMG_00006] Model Elements names shall be self-explanatory, [RS_SWMG_00010] Model Element Names shall follow semantic rules, [RS_SWMG_00011] Model Element Names are composed by arranging standardized keywords [RS_SWMG_00012] Semantic of Model Element Names shall allow variable number of keywords [RS_SWMG_00016] Names shall allow to indicate if the value is a direct measurement or a conditioned value [RS_SWMG_00034] Usage of Unique Keywords [RS_SWMG_00049] Support Identifiable already present in the Master Table [RS_SWMG_00054] Provide guidelines how to resolve name conflicts [RS_SWMG_00059] There shall be a single set of keywords [RS_SWMG_00060] Applicability of Naming Convention [RS_SWMG_00061] Naming Conventionshall be unique
Supporting Material:	Modeling Guide, Meta Model, AI Specification

⌋

4.2 Modeling Requirements

4.2.1 [RS_SWMG_00052] Definition of Package Structure

[RS_SWMG_00052] Definition of Package Structure ⌈

Description:	The Modelling Guide shall specify the Package structure used for standardized AUTOSAR Elements
Rationale:	Model Exchange without path conflicts if standardized M1 AUTOSAR model elements are used.
Use Case:	The modelling guide should specify the packages for DataTypes, Sender ReceiverInterfaces, etc. that are used in the specification of the Functional Interfaces
Dependencies:	–
Supporting Material:	–

⌋

4.2.2 [RS_SWMG_00053] Model shall be compliant to the Meta Model

[RS_SWMG_00053] Model shall be compliant to the Meta Model ⌈

Description:	The AUTOSAR Meta Model defines the structure of AUTOSAR models. Since the MasterTable contains the data to describe the specification of every Domain in the Application Interfaces, it has to be kept consistent with the Meta Model. All model elements attributes shall be used like the Meta Model defines it.
Rationale:	–
Use Case:	–
Dependencies:	–
Supporting Material:	Meta Model

⌋

4.2.3 [RS_SWMG_00055] Continuous Data Type resolution should be a power of two

[RS_SWMG_00055] Continuous Data Type resolution should be a power of two ⌈

Description:	Continuous Data Type resolution should be a power of two, either as a magnitude or inverse.
Rationale:	For cost reasons, in a majority of the commercially available processors on the market today there is no hardware support for floating-point arithmetic. To avoid or limit software emulation of such feature, that would lead to software execution overhead, fixed-point (integer) mathematics is commonly used. A large portion of processors do not even have hardware support for integer multiplication. By assigning to fixed point (integer) numbers a resolution expressed by a power of two, software emulation of multiplication and divide is reduced only to those operation functionally needed by the algorithm.
Use Case:	In a SWC algorithm, apply a gain having a resolution of 0.001/lsb to a variable of type UInt16 having a resolution of 0,004/lsb, to obtain a result having the same resolution. In this case, besides the multiplication and range saturation needed to apply the gain, a divide by 1000 is needed to rescale the result to requested resolution. By converting the operands to power of two resolutions i.e. 2^{-8}/lsb for the variable and 2^{-10}/lsb for the gain, rescaling will be performed by a logical shift right of 10 bits (one instruction cycle in some microprocessors), with no accuracy loss with respect to the first solution.
Dependencies:	–
Supporting Material:	–

⌋

4.2.4 [RS_SWMG_00056] Standardized model elements shall not contain non standardized elements

[RS_SWMG_00056] Standardized model elements shall not contain non standardized elements ⌈

Description:	Standardized model elements shall not contain non standardized elements.
Rationale:	To avoid confusion it is necessary that an element is completely standardized, even not partially.
Use Case:	–
Dependencies:	–
Supporting Material:	A proposed solution to the conflict is the following: • Define a new non standardized composition type that contains the standardized component type and additional non standardized components. • Interfaces of such composition can be all ports of the standardized component type plus the additional non standardized ports.

⌋

4.2.5 [RS_SWMG_00057] Modeling Guide shall support the AUTOSAR methodology

[RS_SWMG_00057] Modeling Guide shall support the AUTOSAR methodology ⌈

Description:	Modelling Guide shall give guidelines that re-usability of model elements shall be exploited as much as possible.
Rationale:	By exploiting the full possibilities of the AUTOSAR methodology, conflicts due to inconsistencies will be less probable, unnecessary redundancies will be removed, maintenance of the data will be improved.
Use Case:	Defining Data Elements of different Interface with the same Data Type, if the same range and resolution is used.
Dependencies:	–
Supporting Material:	AUTOSAR Meta Model.

⌋

5 References

[1] Standardization Template
AUTOSAR_FO_TPS_StandardizationTemplate

[2] Glossary
AUTOSAR_FO_TR_Glossary

A Change history of AUTOSAR traceable items

Please note that the lists in this chapter also include traceable items that have been removed from the specification in a later version. These items do not appear as hyperlinks in the document.

A.1 Traceable item history of this document according to AUTOSAR Release R25-11

A.1.1 Added Requirements in R25-11

none

A.1.2 Changed Requirements in R25-11

none

A.1.3 Deleted Requirements in R25-11

none

A.2 Traceable item history of this document according to AUTOSAR Release R24-11

A.2.1 Added Requirements in R24-11

none

A.2.2 Changed Requirements in R24-11

none

A.2.3 Deleted Requirements in R24-11

none

A.3 Traceable item history of this document according to AUTOSAR Release R23-11

A.3.1 Added Requirements in R23-11

none

A.3.2 Changed Requirements in R23-11

none

A.3.3 Deleted Requirements in R23-11

none