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

Document Version	1.0.2
Document Status	Final
Part of Release	3.2
Revision	1

Document Change History

Date	Version	Changed by	Change Description
27.04.2011	1.0.2	AUTOSAR Administration	Legal disclaimer revised
23.06.2008	1.0.1	AUTOSAR Administration	Legal disclaimer revised
14.12.2007	1.0.0	AUTOSAR Administration	Initial Release

Disclaimer

Advice for users

AUTOSAR Specification Documents may contain exemplary items (exemplary reference models, "use cases", and/or references to exemplary technical solutions, devices, processes or software).

Any such exemplary items are contained in the Specification Documents for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such Specification Documents, nor any later documentation of AUTOSAR conformance of products actually implementing such exemplary items, imply that intellectual property rights covering such exemplary items are licensed under the same rules as applicable to the AUTOSAR Standard.

1 Table of Contents

2 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.

2.1 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”.

3 Conventions to be Used

- In requirements, the following specific semantics shall be used (based on the Internet Engineering Task Force IETF).

  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:

  - SHALL: This word 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.
  - MUST: This word means that the definition is an absolute requirement of the specification due to legal issues.
  - MUST NOT: This phrase means that the definition is an absolute prohibition of the specification due to legal constraints.
  - SHOULD: This word, or the adjective "RECOMMENDED", mean 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" mean 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, MUST 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, MUST be prepared to interoperate with another implementation, which does not include the option (except, of course, for the feature the option provides.)

4 Requirements Guidelines

Existing specifications shall be referenced (in form of a single requirement).
Differences to these specifications are specified as additional requirements.

4.1 Requirements quality

All Requirements shall have the following properties:

- Redundancy
  Requirements shall not be repeated within one requirement or in other requirements
- Clearness
  All requirements shall allow one possibility of interpretation only. Only technical terms of the glossary may be used.
- Atomicity
  Each Requirement shall only contain one requirement. A Requirement is atomic if it cannot be split up in further requirements.
- Testability
  Requirements shall be testable by analysis, review or test.
- Traceability
  The source and status of a requirement shall be visible at all times.

4.2 Requirements identification

Each requirement has its unique identifier starting with a to be defined prefix (default is “REQ”, but please change that to enable a unique requirements ID calculation).
For any review annotations, remarks and/or questions, please refer to this unique ID rather than chapter or page numbers!

5 Naming Convention Requirements

5.1 [MG001] Distinguish Standardized vs not standardized model elements of type ARElement]

Initiator:	10.1 -> reformulated by WP10.0
Date:	<date of creation>
Short Description:	Distinguish Standardized vs not standardized AUTOSAR model elements of type ARElement.
Type:	changed
Importance:	high | medium | low
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:	
Conflicts:	Usage of Prefix or suffix could have an impact on requirements asking for a limitation on the length of the names.
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.

5.2 [MG002] Name should reflect the purpose of the interface and/or data element

Initiator:	10.1
Date:	<date of creation>
Short Description:	Name should reflect the purpose of the interface and/or data element
Type:	changed
Importance:	high | medium | low
Description:	Identify whether an interface and/or data element is a command, a status, a request, a value, etc..
Rationale:	It is necessary to avoid the creation of identical names for signals having a different purpose.
Use Case:	
Dependencies:	[MG005] Easy creation of names
Conflicts:	
Supporting Material:	Source : 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)

5.3 [MG005] Easy creation of names

Initiator:	10.1 -> reformulated by WP10.0
Date:	<date of creation>
Short Description:	Easy creation of names
Type:	new
Importance:	high | medium | low
Description:	
Rationale:	
Use Case:	
Dependencies:	
Conflicts:	
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 and calibration.

5.4 [MG006] Model Elements names shall be self-explanatory

Initiator:	10.1 -> reformulated by WP10.0
Date:	<date of creation>
Short Description:	Model Elements names shall be self-explanatory.
Type:	new
Importance:	high | medium | low
Description:	
Rationale:	
Use Case:	E.g. data-element, ports, interfaces, composition, etc.
Dependencies:	
Conflicts:	[MG005] Easy creation of names
Supporting Material:	

5.5 [MG007] Distinguish model elements of different model element suppliers

Initiator:	Siemens VDO -> reformulated by WP10.0
Date:	07.12.2006
Short Description:	Distinguish model elements of different model element suppliers.
Type:	New
Importance:	Medium
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 ModeDeclarationGroupPrototype, DataElementPrototype, CalprmElementPrototype, OperationPrototype, ArgumentPrototype due to the required uniformity of names as precondition for connectability of ports
Conflicts:	

Supporting Material:	Could either be done by naming convention or by usage of other model elements like AdminData.

5.6 [MG010] Model Element Names shall follow semantic rules

Initiator:	Siemens VDO -> reformulated by WP10.0
Date:	07.12.2006
Short Description:	Model Element Names shall follow semantic rules
Type:	new
Importance:	high
Description:	
Rationale:	By doing so, the compliancy to the naming convention would be verifiable by name checkers or name creator tools.
Use Case:	
Dependencies:	[MG005] Easy creation of names [MG048] Easy lookup of names in databases
Conflicts:	
Supporting Material:	AUTOSAR_NC-Guideline_V1-1.doc, AUTOSAR_NC-Keyword_V1-2.xls

5.7 [MG011] Model Element Names are composed by arranging standardized keywords

Initiator:	Siemens VDO
Date:	07.12.2006
Short Description:	Model Element Names are composed by arranging standardized keywords
Type:	new
Importance:	high
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:	[MG005] Easy creation of names [MG034] Usage of Unique Keywords
Conflicts:	[MG006] Model Elements names shall be self-explanatory [MG014] Length restriction for short names of Identifiable Model element names shall have a clear and understandable meaning even with length restriction for short names. - High number of required keywords/catchwords and the need to keep names short for use cases in function development, documentation and calibration require tool support and managed set of predefined keywords
Supporting Material:	AUTOSAR_NC-Guideline_V1-1.doc, AUTOSAR_NC-Keyword_V1-2.xls

5.8 [MG012] Semantic of Model Element Names shall allow variable number of keywords

Initiator:	Siemens VDO
Date:	08.12.2006
Short Description:	Semantic of Model Element Names shall allow variable number of keywords

Type:	new
Importance:	high
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:	[MG005] Easy creation of names [MG010] Model Element Names shall follow semantic rules [[MG034] Usage of Unique Keywords
Conflicts:	[MG014] Length restriction for short names of Identifiable
Supporting Material:	AUTOSAR_NC-Guideline_V1-1.doc An example of solution: Eng_tqCluReqDrvSlow -> Engine Torque at Clutch Slow Request Veh_v -> Vehicle Speed

5.9 [MG014] Length restriction for short names of Identifiable

Initiator:	Siemens VDO
Date:	07.12.2006
Short Description:	Length restriction for short names of Identifiable
Type:	New
Importance:	High
Description:	Short Names of Identifiable shall be restricted to total length of 32 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 it shall not be infinite.)
Use Case:	
Dependencies:	
Conflicts:	[MG005] Easy creation of names [MG006] Model Elements names shall be self-explanatory
Supporting Material:	A rule restricting the number of characters to 32 is already present in the MetaModel. [a-zA-Z][a-zA-Z_0-9]{0-31}

5.10 [MG015] In case of an array interface and data type names should indicate the array length

Initiator:	WP 10.3, modified by WP10.0
Date:	12.12.2006
Short Description:	In case of an array interface and data type names should indicate the array length.
Type:	New
Importance:	high | medium | low
Description:	
Rationale:	
Use Case:	Interface and Data Type Names indicating an array of 4 in case of wheel or "corner" individual data. Interface and Data Type Names indicating an array of 4 in case of doors.
Dependencies:	
Conflicts:	Which names are mentioned
Supporting Material:	This requirement only applies to arrays. The difference between an array of four elements and an interface of four elements are as follows: – With the array data type all the data are sent/received at the same time.

	– With the interface with multiple data elements, data elements are updated independently.

5.11 [MG016] Names shall allow to indicate if the value is a direct measurement or a conditioned value

Initiator:	WP 10.3 -> reformulated by WP10.0
Date:	12.12.2006
Short Description:	Names shall allow to indicate if the value is a direct measurement or a conditioned value.
Type:	New
Importance:	high | medium | low
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 SWC outputting a measured physical value and feeding it to another SWC 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:	
Conflicts:	
Supporting Material:	Possible solution: use a dedicated keyword in the name semantic to indicate such information.

5.12 [MG017] Names should follow the ISO 8855 for English naming.

Initiator:	WP 10.3
Date:	12.12.2006
Short Description:	Names should follow the ISO 8855 for english naming.
Type:	New
Importance:	high | medium | low
Description:	
Rationale:	
Use Case:	
Dependencies:	[MG030] Use English as Standard Language for Names.
Conflicts:	
Supporting Material:	

5.13 [MG026] The naming convention shall provide an attribute to describe the data flow property

Initiator:	WP 10.3 -> reformulated by WP10.0
Date:	12.12.2006
Short Description:	The naming convention shall provide an attribute to describe the data flow property.
Type:	New
Importance:	high | medium | low
Description:	
Rationale:	Data flow property (such as Request and Status) is needed to differentiate names that otherwise would be equal.
Use Case:	Example: PGearEngaged and PGearRequest
Dependencies:	
Conflicts:	
Supporting Material:	Possible solution: See following lists: IRW, ISG, FLT, RSP, OSG, ORW, IFL, SYN, NFY, PRV, AVL . Complete list given here but a subset can be valid as well for 10.3 Flow Type : Definition irw : input raw – an electrical input signal isg : input signal – an internal node signal from the electrical signal read to the software flt : filtered – a filtered software value based on the input signal rsp : response – a response of the software based on an algorithm requiring multiple inputs osg : output signal – an internal node signal from the software drive to electrical signal orw : output raw – an electrical output signal ifl : information flow – a composite data flow syn : synchronized – a synchronized response nfy : notify – feedback from serial data that the data was sent prv : previous – a data flow that contains the previous value of a signal avl : available – a data flow that contains the availability status of a signal

5.14 [MG030] Use English as Standard Language for Names.

Initiator:	WP 10.2
Date:	12.13.2006
Short Description:	Use English as Standard Language for Names.
Type:	New
Importance:	high | medium | low
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:	[MG017] Names should follow the ISO 8855 for English naming.
Conflicts:	
Supporting Material:	Naming Convention 1.0 from WP10.2 §1.4

5.15 [MG031] No Architectural Information in Names

Initiator:	WP 10.2
Date:	12.13.2006
Short Description:	No Architectural Information in Names.
Type:	New
Importance:	high | medium | low
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:	
Conflicts:	
Supporting Material:	Naming Convention 1.0 from WP10.2 §1.4

5.16 [MG034] Usage of Unique Keywords

Initiator:	WP 10.2
Date:	12.13.2006
Short Description:	Usage of Unique Keywords.
Type:	New
Importance:	high | medium | low
Description:	Keywords used to compose Names shall be unique.
Rationale:	
Use Case:	Automated checking of Names with respect to conformance will be possible.
Dependencies:	[MG010] Model Element Names shall follow semantic rules [MG011] Model Element Names are composed by arranging standardized keywords
Conflicts:	
Supporting Material:	Naming Convention 1.0 from WP10.2 §1.4

5.17 [MG039] Avoid usage of Trailing underscores

Initiator:	WP 10.2
Date:	12.13.2006
Short Description:	Avoid usage of Trailing underscores.
Type:	New
Importance:	high | medium | low
Description:	Names shall not end with an underscore [_] character.
Rationale:	
Use Case:	
Dependencies:	
Conflicts:	
Supporting Material:	Naming Convention 1.0 from WP10.2 §2

5.18 [MG040] Avoid sequences of underscores characters.

Initiator:	WP 10.2
Date:	12.13.2006

Short Description:	Avoid sequences of underscore characters directly after each other.
Type:	New
Importance:	high | medium | low
Description:	Avoid sequences of underscore characters directly after each other [__].
Rationale:	Waste of characters space.
Use Case:	
Dependencies:	
Conflicts:	
Supporting Material:	Naming Convention 1.0 from WP10.2 §2

5.19 [MG041] Do not rely on uppercase/lowercase difference only.

Initiator:	WP 10.2
Date:	12.13.2006
Short Description:	Do not rely on uppercase/lowercase difference only.
Type:	New
Importance:	high | medium | low
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:	
Conflicts:	
Supporting Material:	Naming Convention 1.0 from WP10.2 §2

5.20 [MG048] Easy lookup of names in databases

Initiator:	WP 10.1 -> reformulated by WP10.0
Date:	31.01.2007
Short Description:	Easy lookup of names in databases
Type:	New
Importance:	high | medium | low
Description:	
Rationale:	
Use Case:	
Dependencies:	
Conflicts:	
Supporting Material:	[MG005] Easy creation of names

5.21 [MG049] Support Identifiable already present in the MasterTable

Initiator:	WP10.0
Date:	31.01.2007
Short Description:	Support Identifiable already present in the MasterTable
Type:	new
Importance:	medium
Description:	All model element types that are used in the Master Table such as SenderReceiver interfaces, DataElements, DataTypes, Unit, Component Types, etc. shall be supported by the Modelling Rules.
Rationale:	
Use Case:	
Dependencies:	

Conflicts:	
Supporting Material:	“Master Table” is a place-holder for Identifiable that are listed in that file

5.22 [MG050] comprehensive long names

Initiator:	Siemens VDO
Date:	25.01.2007
Short Description:	comprehensive long names
Type:	New
Importance:	High
Description:	The long name of identifiable shall be filled with a comprehensive description based on the Descriptions of the keywords.
Rationale:	Provide a comprehensive and meaningful name
Use Case:	Veh_v: vehicle velocity short name: Eng_nMax_C: calibrated maximum engine speed
Dependencies:	[MG011] Model Element Names are composed by arranging standardized keywords [MG012] Semantic of Model Element Names shall allow variable number of keywords . [MG010] Model Element Names shall follow semantic rules
Conflicts:	
Supporting Material:	AUTOSAR_NC-Guideline_V1-1.doc, AUTOSAR_NC-Keywords_V1-2.xls

5.23 [MG054] Provide guidelines how to resolve name conflicts

Initiator:	Siemens VDO
Date:	01.02.2007
Short Description:	Provide guidelines to resolve name conflicts
Type:	changed
Importance:	high | medium | low
Description:	The modelling guide should provide guidelines how to resolve name conflicts between related elements.
Rationale:	
Use Case:	
Dependencies:	
Conflicts:	
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 MG014. Another possible solution is the use of sub packages.

6 Modeling Requirements

6.1 [MG052] Definition of Package Structure

Initiator:	Siemens VDO, changed by WP10.0
Date:	26.01.2007
Short Description:	Definition of Package Structure
Type:	New
Importance:	High
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, SenderReceiverInterfaces, etc. that are used in the specification of the Functional Interfaces
Dependencies:	
Conflicts:	
Supporting Material:	

6.2 [MG053] Model shall be compliant to the Meta Model

Initiator:	WP10.0
Date:	31.01.2007
Short Description:	Model shall be compliant to the Meta Model
Type:	New
Importance:	High
Description:	The AUTOSAR Meta Model defines the structure of AUTOSAR models. Since the MasterTable contains the data to describe the specification of WP10.x, 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:	
Conflicts:	
Supporting Material:	Meta Model

6.3 [MG055] Continuous Data Type resolution should be a power of two

Initiator:	Magneti Marelli
Date:	07.02.2007
Short Description:	Continuous Data Type resolution should be a power of two.
Type:	New
Importance:	High
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:	
Conflicts:	
Supporting Material:	

6.4 [MG056] Standardized model elements shall not contain non standardized elements

Initiator:	WP10.0
Date:	28.02.2007
Short Description:	Standardized model elements shall not contain non standardized elements.
Type:	New
Importance:	High
Description:	Standardized model elements shall not contain non standardized elements.
Rationale:	To avoid confusion it is necessary that an element is completely standardized, not only partially.
Use Case:	
Dependencies:	
Conflicts:	It seems that sometime is necessary to extend a standardized component type with additional non standardized ports. To avoid confusion this should be not allowed.
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.

6.5 [MG057] Modeling Guide shall support the AUTOSAR methodology

Initiator:	WP10.0
Date:	13.06.2007
Short Description:	Modeling Guide shall support the AUTOSAR methodology
Type:	New
Importance:	High
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:	
Conflicts:	
Supporting Material:	AUTOSAR Meta Model.