Document Title	Requirements on Methodology
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	362
Document Classification	Auxiliary

Document Version	1.1.0
Document Status	Final
Part of Release	4.0
Revision	3

Document Change History

Date	Version	Changed by	Change Description
28.11.2011	1.1.0	AUTOSAR Administration	• Improved requirements tracing. • New requirements numbering according to AUTOSAR definition.
30.11.2009	1.0.0	AUTOSAR Administration	Initial Release

Disclaimer

This specification and the material contained in it, as released by AUTOSAR, is for the purpose of information only. AUTOSAR and the companies that have contributed to it shall not be liable for any use of the specification.

The material contained in this specification is protected by copyright and other types of Intellectual Property Rights. The commercial exploitation of the material contained in this specification requires a license to such Intellectual Property Rights.

This specification may be utilized or reproduced without any modification, in any form or by any means, for informational purposes only.
For any other purpose, no part of the specification may be utilized or reproduced, in any form or by any means, without permission in writing from the publisher.

The AUTOSAR specifications have been developed for automotive applications only. They have neither been developed, nor tested for non-automotive applications.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Advice for users

AUTOSAR specifications 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 specifications for illustration purposes only, and they themselves are not part of the AUTOSAR Standard. Neither their presence in such specifications, 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.

Table of Contents

1 Scope of this document

This document defines the requirements needed to specify the AUTOSAR methodology. Even if the AUTOSAR methodology specification already exists, this document will make the link between the needs and how it has been implemented. This document is a good tool to test the specification, see what is missing, what effort should be done in terms of planning and means.

2 Acronyms and Abbreviations

Acronyms and abbreviations that have a local scope are not contained in the AUTOSAR glossary. These must appear in a local glossary.

Acronym:	***Description:***
SIL	Safety Integrity Level (IEC61508 definition)

Abbreviation:	***Description:***
ASAM MCD	Association for Standardization of Automation- and Measuring Systems Measurement, Calibration and Diagnostics
AUTOSAR	Automotive Open System Architecture
BSW	Basic Software
CPU	Central Processing Unit
DWARF	Debug With Arbitrary Record Format
ECU	Electronic Control Unit
MCAL	MicroController Abstraction Layer
OEM	Original Equipment Manufacture
OS	Operating System
RTE	Runtime Environment
SW	Software
SWC	Software Component
VFB	Virtual Functional Bus
XML	Extensible Markup Language

3 Requirements Traceability

Requirement	***Satisfied by***
[Main10] Using AUTOSAR a system reliability shall be achievable with a failure rate down to 10^{-8} per hour	Not Applicable
[Main11] AUTOSAR shall provide means to reduce system down-time	[RS_METH_0033] <Methodology should support VFB concept >
[Main20] AUTOSAR shall provide mechanisms to support redundancy paths	Not Applicable
[Main30] A SIL-3-Level compatible development process must be possible with AUTOSAR	[RS_METH_0018] < Methodology shall be modular > [RS_METH_0069] <It shall be possible to add Documentation to a Work Product> [RS_METH_0063] <Work Products shall be capable to be version controlled > [RS_METH_0050] < Work products shall have a description> [RS_METH_0056] < AUTOSAR methodology shall not be bound to a particular lifecycle model > [RS_METH_0057] < AUTOSAR Methodology shall support traceability to external artifacts >
[Main50] AUTOSAR shall support inter- and intra-ECU-communication mechanisms with high reliability	Not Applicable
[Main60] AUTOSAR shall provide open and standardized software interfaces for intra-ECU and inter-ECU communication	Not Applicable
[Main70] AUTOSAR shall provide complete interfaces to application software and basic software modules	Not Applicable
[Main80] AUTOSAR shall ease the reusability of software and its concepts and implementations	[RS_METH_0062] < Methodology shall support configuration of parameters with different binding time >
[Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process	[RS_METH_0017] < Methodology shall clearly define what is standardized and what is not standardized > [RS_METH_0021] < Methodology shall define Activities > [RS_METH_0043] < Activities shall have a purpose> [RS_METH_0046] < Activities shall have input work products> [RS_METH_0047] < Activities shall have output work products> [RS_METH_0048] < Activities shall include roles> [RS_METH_0027] < Methodology shall define unambiguous guidance terminology > [RS_METH_0042] < Methodology shall incorporate the usage of industry standard tools > [RS_METH_0066] < Activities shall include tools >

[Main100] All AUTOSAR standard software functions shall be standardized across OEM and Supplier	Not Applicable
[Main110] AUTOSAR shall provide a software architecture that is applicable across different functional domains	Not Applicable
[Main120] AUTOSAR shall provide means to clearly check the conformance of an AUTOSAR-implementation with its AUTOSAR-specification	Not Applicable
[Main130] AUTOSAR shall provide an abstraction of the application software from hardware	[RS_METH_0033] <Methodology should support VFB concept >
[Main140] AUTOSAR shall provide an independency of application software from in-vehicle communication technologies	Not Applicable
[Main141] AUTOSAR should provide an independency of application software from operating systems	[RS_METH_0033] <Methodology should support VFB concept >
[Main150] AUTOSAR shall provide mechanisms, methods, processes, and tools to encapsulate application software from the infrastructure	[RS_METH_0033] <Methodology should support VFB concept > [RS_METH_0002] <Methodology shall explain the usage of SW-C template > [RS_METH_0003] < Methodology shall explain the usage of BSW Module Template > [RS_METH_0004] < Methodology shall explain the usage of the ECU Configuration template > [RS_METH_0005] <Methodology shall explain the usage of the System Template >
[Main160] AUTOSAR shall provide a functional interface view of the entire system	Not Applicable
[Main170] AUTOSAR shall provide secure access to ECU	Not Applicable
[Main180] AUTOSAR shall provide protection/unlock mechanisms for software through appropriate services in the infrastructure	Not Applicable
[Main190] AUTOSAR shall provide interoperability with legacy software	[RS_METH_0018] < Methodology shall be modular >
[Main200] AUTOSAR shall imply only small memory and performance impacts when used in today’s micro controllers	Not Applicable
[Main210] AUTOSAR shall provide means to integrate AUTOSAR ECU’s in non-AUTOSAR networks	[RS_METH_0016] < Methodology shall support building a system of both Autosar and Non-Autosar ECUs > [RS_METH_0018] < Methodology shall be modular >

[Main220] AUTOSAR shall support following programming languages: C, C++, Java	[RS_METH_0015] < Methodology shall be independent of programming language > [RS_METH_0038] < Methodology shall support the C programming language >
[Main230] AUTOSAR shall support networks of networks including sub networks	Not Applicable
[Main240] AUTOSAR shall provide means to protect SW-Components from malicious SW-components	Not Applicable
[Main250] AUTOSAR shall provide means to achieve compositionality	Not Applicable
[Main260] AUTOSAR shall provide diagnostics means during runtime, for production and services purposes	Not Applicable
[Main270] Releases of AUTOSAR shall be forward and backward compatible	Not Applicable
[Main280] The AUTOSAR architecture must provide dynamic communication pattern	Not Applicable
[Main290] AUTOSAR shall ensure the verification of all processes and products developed within AUTOSAR	[RS_METH_0002] <Methodology shall explain the usage of SW-C template > [RS_METH_0003] < Methodology shall explain the usage of BSW Module Template > [RS_METH_0004] < Methodology shall explain the usage of the ECU Configuration template > [RS_METH_0005] <Methodology shall explain the usage of the System Template >

[Main300] AUTOSAR shall support work-share in large inter-company development groups	[RS_METH_0033] <Methodology should support VFB concept > [RS_METH_0018] < Methodology shall be modular > [RS_METH_0020] < Methodology shall support iterations > [RS_METH_0002] <Methodology shall explain the usage of SW-C template > [RS_METH_0003] < Methodology shall explain the usage of BSW Module Template > [RS_METH_0004] < Methodology shall explain the usage of the ECU Configuration template > [RS_METH_0005] <Methodology shall explain the usage of the System Template > [RS_METH_0006] < Methodology shall explain how Autosar system is built > [RS_METH_0025] < Methodology shall define Work products > [RS_METH_0050] < Work products shall have a description> [RS_METH_0051] < Work products shall have a reference(s) to metaclass(es) in the Autosar Metamodel > [RS_METH_0052] < It must be possible to avoid duplication of data in Work Products > [RS_METH_0054] < Work Products shall not have circular references with other work products > [RS_METH_0061] < Methodology shall describe the change of existing work products > [RS_METH_0063] <Work Products shall be capable to be version controlled > [RS_METH_0067] < Methodology document shall include hyperlinks between Activities, Roles, Work Products, and Guidance >
[Main310] AUTOSAR shall support hierarchical design methods	[RS_METH_0041] <Methodology shall support the Bottom/Up Approach >
[Main320] Definitions of relations between SW components are exhaustive and formal	Not Applicable
[Main330] SW components are protected from illegal access	Not Applicable
[Main340] Protection of timing requirements is supported by AUTOSAR	Not Applicable

[Main350] AUTOSAR methods shall be FMEA compatible	[RS_METH_0041] <Methodology shall support the Bottom/Up Approach > [RS_METH_0025] < Methodology shall define Work products > [RS_METH_0050] < Work products shall have a description> [RS_METH_0051] < Work products shall have a reference(s) to metaclass(es) in the Autosar Metamodel > [RS_METH_0052] < It must be possible to avoid duplication of data in Work Products > [RS_METH_0054] < Work Products shall not have circular references with other work products > [RS_METH_0063] <Work Products shall be capable to be version controlled >
[Main360] Management of vehicle diversity is supported by AUTOSAR	[RS_METH_0062] < Methodology shall support configuration of parameters with different binding time >
[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method	[RS_METH_0021] < Methodology shall define Activities > [RS_METH_0043] < Activities shall have a purpose> [RS_METH_0046] < Activities shall have input work products> [RS_METH_0047] < Activities shall have output work products> [RS_METH_0048] < Activities shall include roles> [RS_METH_0028] < Methodology shall define Roles > [RS_METH_0064] < Roles shall have a description > [RS_METH_0066] < Activities shall include tools >
[Main380] Basic requirements to change process in design are predefined and supported by AUTOSAR	Not Applicable

4 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.)

5 Methodology Requirements

This chapter provides a definition of the requirements.

5.1 Main Requirements

5.1.1 [RS_METH_0006] < Methodology shall explain how Autosar system is built >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall explain how Autosar system is built
Type:	**New**
Importance:	**high**
Description:	Methodology shall explain how Autosar system is built using the templates and activities supported by guidance. It should be like a user manual to help an organization efficiently apply Autosar.
Rationale:	A strong methodology is necessary to effectively manage building a large system.
Use Case:	Engineer would like to complete an activity and would like to know what inputs are needed, Guidance should be used, etc. Typical use cases involved to build an Autosar system include: SWC implementation ECU integration System integration
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups

5.1.2 [RS_METH_0033] <Methodology should support VFB concept >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	16.10.2007
Short Description:	Methodology should support VFB concept
Type:	**new**
Importance:	**medium**
Description:	Virtual Functional Bus concept allows early checks between SW-C with a complete abstraction of the Hardware. The methodology should include this concept.
Rationale:	To improve the integration phases and the concurrent development.
Use Case:	In AUTOSAR, an application is modeled as a composition of interconnected components. The “virtual functional bus” is the communication mechanism that allows these components to interact. Even if all the resources used by these components are not available (HW / Network) some basic checks can be done and early problems can be solve that will ease the integration phase later.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main11] AUTOSAR shall provide means to reduce system down-time

	[Main130] AUTOSAR shall provide an abstraction of the application software from hardware [Main141] AUTOSAR should provide an independency of application software from operating systems [Main150] AUTOSAR shall provide mechanisms, methods, processes, and tools to encapsulate application software from the infrastructure [Main300] AUTOSAR shall support work-share in large inter-company development groups

5.1.3 [RS_METH_0041] <Methodology shall support the Bottom/Up Approach >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	16.10.2007
Short Description:	Methodology shall support Bottom/Up Approach
Type:	**new**
Importance:	**medium**
Description:	Methodology shall support the Bottom/Up (B/U) Approach. In this approach, all constraints coming from the Hardware in the B/U (ECUs/Sensors/Actuators) should be taken in account
Rationale:	To improve the integration phases, and to master the complexity in embedded RT distributed systems
Use Case:	If in a given vehicle architecture, a new ECU is added or an existing ECU is replaced with a new one, all the new or modified resources from the ECU need to be included into the system configuration during integration.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main310] AUTOSAR shall support hierarchical design methods [Main350] AUTOSAR methods shall be FMEA compatible

5.1.4 [RS_METH_0016] < Methodology shall support building a system of both Autosar and Non-Autosar ECUs >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall support building a system of both Autosar and Non-Autosar ECUs
Type:	**New**
Importance:	**high**
Description:	Methodology needs to show how to build a system of AUTOSAR compliant ECUs that are on the same architecture with non AUTOSAR compliant ECUs.
Rationale:	Provide a migration path to AUTOSAR for non-AUTOSAR applications.
Use Case:	Legacy ECUs and LIN slaves need to interoperate with Autosar ECUs.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main210] AUTOSAR shall provide means to integrate AUTOSAR ECU's in non-AUTOSAR networks

5.1.5 [RS_METH_0017] < Methodology shall clearly define what is standardized and what is not standardized >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall clearly define what is standardized and what is not standardized
Type:	**New**
Importance:	**medium**
Description:	Show which portions of Methodology portions are normative and informative. The inputs/outputs, and content must be compatible among Autosar tools for certain core activities.
Rationale:	Necessary to ensure Interoperability of Tools to support all Autosar Activities
Use Case:	Systems analysis tools are undergoing significant innovations that may not appropriate for standardization. Since the benefit of including these activities within the methodology still exists to help provide completeness, but should be clearly shown to be non-standardized.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.1.6 [RS_METH_0018] < Methodology shall be modular >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall be modular
Type:	**New**
Importance:	**High**
Description:	Utilize process components. Sub processes shall be complete and testable on their own to allow usage of certain portions of the Methodology while still integrating with legacy tools and processes.
Rationale:	Easier to understand and verify all portions of the Methodology Easier to manage modifications, encapsulates ripple effect due to changes to allow migration of current processes Easier to utilize both legacy and Autosar activities It should be possible to start from an intermediate activity and not necessarily from the beginning of the methodology. A modular Methodology facilitates organizations to migrate from or merge with their current processes. A modular Methodology allows organizations to insert intermediate activities such as quality gates, or other inspections, as well as collect metrics necessary to comply with CMMI processes and/or SIL-3.
Use Case:	An organization is planning to introduce an Autosar ECU into their existing architecture, but is not planning to use the System Template and their respective activities and work products. Rather they plan to begin directly at the ECU level.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main190] AUTOSAR shall provide interoperability with legacy software [Main210] AUTOSAR shall provide means to integrate AUTOSAR ECU’s in non-AUTOSAR networks [Main300] AUTOSAR shall support work-share in large inter-company development groups [Main30] A SIL-3-Level compatible development process must be possible

	with AUTOSAR

5.1.7 [RS_METH_0032] < Methodology shall respect the different levels of Abstractions >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	16.10.2007
Short Description:	The methodology shall respect the different levels of Abstractions
Type:	**New** | change(ID)
Importance:	High | **medium** | low
Description:	The methodology shall respect the Software Component, the System, and the ECU levels of Abstractions.
Rationale:	To improve the integration phases and to master the complexity in embedded RT distributed systems.
Use Case:	AUTOSAR is using several abstractions levels to describe the information exchanged between the different players. In an early phase the “Virtual Functional Bus” is used in other phases we are working with the implementation of the SWC in several ECUs. The exchange between the “real” world and the Virtual world should be described and supported by the methodology.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main130] AUTOSAR shall provide an abstraction of the application software from hardware

5.1.8 [RS_METH_0020] < Methodology shall support iterations >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall support iterations
Type:	**New**
Importance:	**high**
Description:	Need to support both small and large iteration loops.
Rationale:	Meet Autosar Quality requirements. Redoing work is error-prone. Support round trip engineering.
Use Case:	Small iteration loops inside ECU Configuration Activity Large iteration loops from System Design impacting ECU Configuration. Automotive systems are typically developed in several sample phases (A, B, C, etc) A Single Software Component is updated in a AUTOSAR System. The updated ECU Extract still matches the existing ECU Configuration (as long as no contradicting changes are made in the iteration).
Dependencies:	[RS_METH_0062] < Methodology shall support configuration of parameters with different binding time >
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups

5.1.9 [RS_METH_0062] < Methodology shall support configuration of parameters with different binding time >

Initiator:	Jonas Hansson (FMC)
Date:	11.12.2007
Short Description:	Methodology shall support configuration of parameters with different binding time.
Type:	**New**
Importance:	**high**
Description:	Configuration of parameters can be performed in different process steps: pre-compile, link time, and post-build. The Autosar methodology must support system development with different combinations of these configuration classes.
Rationale:	--
Use Case:	OEM configuration of post-build data after a release from a Tier1 supplier. Handling information related to different configuration classes as separate configuration items (units for version control).
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main80] AUTOSAR shall ease the reusability of software and its concepts and implementations [Main360] Management of vehicle diversity is supported by AUTOSAR

5.2 Template Requirements

5.2.1 [RS_METH_0002] <Methodology shall explain the usage of SW-C template >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	28.09.2007
Short Description:	Methodology shall explain the typical usage of SW-C template
Type:	**new**
Importance:	**high**
Description:	The software component template includes numerous features including atomic software components, composition components, interfaces, ports, etc. The Methodology shall clearly show the activities to create/modify these and which activities require these to be completed before commencing.
Rationale:	Methodology Consistency using the SW-C template
Use Case:	An existing software component is implemented and is to be integrated into a suppliers ECU. A description of the component is needed in order to correctly integrate that component into the ECU. A new system is being defined and requires a software component to contain specified interfaces. That specification is given to a supplier to implement the required software.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main150] AUTOSAR shall provide mechanisms, methods, processes, and tools to encapsulate application software from the infrastructure

5.2.2 [RS_METH_0003] < Methodology shall explain the usage of BSW Module Template >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	28.09.2007
Short Description:	Methodology shall explain the typical usage of BSW Module Template
Type:	**new**
Importance:	**high**
Description:	The basic software module template includes numerous features describing Interfaces and Data properties of Basic Software.
Rationale:	Methodology Consistency using the BSW Module description
Use Case:	An existing BSW module is implemented and is to be integrated into an ECU. A description of the BSW Module is needed in order to correctly integrate that module into the ECU.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main150] AUTOSAR shall provide mechanisms, methods, processes, and tools to encapsulate application software from the infrastructure

5.2.3 [RS_METH_0004] < Methodology shall explain the usage of the ECU Configuration template >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	28.09.2007
Short Description:	Methodology shall explain the typical usage of the ECU Configuration template
Type:	**new**
Importance:	**High**
Description:	Depending on actors basic roles (supplier/ customer), the tasks and the way to configure the ECU can be completely different. Basic Use Cases should clarify the way of using this template.
Rationale:	Methodology consistency using the ECU configuration
Use Case:	Vendor of BSW is setting /giving recommended or predefined values Conformance test use the Template to document which configuration is used in the Conformance Test. Use to describe the configuration when integrating a BSW module into an ECU.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main150] AUTOSAR shall provide mechanisms, methods, processes, and tools to encapsulate application software from the infrastructure

5.2.4 [RS_METH_0005] <Methodology shall explain the usage of the System Template >

Initiator:	Bernard Sanchez (SIEMENS VDO Automotive)
Date:	28.09.2007
Short Description:	Methodology shall explain the typical usage of the System Template

Type:	**New**
Importance:	**High**
Description:	Depending on actors basic roles (supplier/ customer), and the different phases, the tasks and the way to describe the System can be quite different. Basic use cases should clarify the way of using this template.
Rationale:	Methodology consistency using the System Templates
Use Case:	OEM as integrator is fixing the ECUs, the topology and the list of the SW-C for one vehicle for the suppliers. Supplier is delivering to the OEMs a sub-system for integration.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main150] AUTOSAR shall provide mechanisms, methods, processes, and tools to encapsulate application software from the infrastructure

5.3 Programming Language

5.3.1 [RS_METH_0015] < Methodology shall be independent of programming language >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall be independent of programming language
Type:	**New**
Importance:	**High**
Description:	The methodology shall be independent of programming language by providing generic solutions. For portions that are necessarily dependent on the programming language, these sections shall be explicitly noted and modular such that the overall methodology can be tailored to accommodate other programming languages.
Rationale:	By appropriately structuring the methodology to support existing and emerging programming languages, the Methodology can be consistently and successfully applied across an entire vehicle.
Use Case:	An ECU is built on a microcontroller optimized for programming language ABC. The Methodology explains when and how to specify and select the implementation of the software components deployed to that node that are compatible with that programming language.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main220] AUTOSAR shall support following programming languages: C, C++, Java

5.3.2 [RS_METH_0038] < Methodology shall support the C programming language >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall support the C programming language

Type:	**New**
Importance:	**high**
Description:	Methodology shall show how to realize an Autosar System that uses the C programming language.
Rationale:	Presently C is the most common programming language used in the automotive embedded environment.
Use Case:	A software component implemented in the C language is to be integrated into an Autosar ECU.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main220] AUTOSAR shall support following programming languages: C, C++, Java

5.4 Activities

5.4.1 [RS_METH_0021] < Methodology shall define Activities >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall define Activities
Type:	**New**
Importance:	**High**
Description:	Methodology shall define the activities necessary to satisfy the Autosar Use Cases.
Rationale:	Allow tools from multiple vendors to be used to complete all Autosar Activities in order to build an Autosar compliant System. Provides a common language to communicate among Autosar Members.
Use Case:	Activities include very specific tasks such as “Generate the RTE”, and “Define Topology”.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method [Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.4.2 [RS_METH_0043] < Activities shall have a purpose>

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Activities shall have a purpose
Type:	**New**
Importance:	**High**
Description:	Each activity shall include a textual purpose explaining why the activity is needed, its objectives, and what can be expected upon completion.
Rationale:	A clearly defined purpose provides a common understanding of the activity and helps to ensure consistency between tool implementations that complete the activity.
Use Case:	Example activity description for “Generate the RTE”: This purpose of this

	activity is to generate the RTE layer of an Autosar ECU. Example activity description for “Define Topology”: The purpose of this activity is to select available Autosar ECUs and instantiate them onto a vehicle topology.
Dependencies:	RS_METH_0021
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method [Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.4.3 [RS_METH_0046] < Activities shall have input work products>

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Activities shall have input work products
Type:	**New**
Importance:	**High**
Description:	Each activity shall include which work products are used to complete an activity. If no work products are needed, then this shall be explicitly stated.
Rationale:	Explicitly stating what work products are needed for an activity is necessary to ensure tool’s supporting that activity are interoperable.
Use Case:	Example input work products for Generate RTE Activity include Software Components, ECU Communication Database, ECU Configuration.
Dependencies:	RS_METH_0021
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method [Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.4.4 [RS_METH_0047] < Activities shall have output work products>

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Activities shall have output work products
Type:	**New**
Importance:	**High**
Description:	Each activity shall include which work products are produced or modified during an activity. If no work products are produced or modified, then this shall be explicitly stated.
Rationale:	Explicitly stating what work products are produced or modified during an activity is necessary to ensure tool’s supporting that activity are interoperable.
Use Case:	Example output work products for Generate RTE include the RTE source files.
Dependencies:	RS_METH_0021
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method

	[Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.4.5 [RS_METH_0048] < Activities shall include roles>

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Activities shall include roles
Type:	**New**
Importance:	**Low**
Description:	Each activity shall include the roles that are responsible to perform and, if any, to support it.
Rationale:	By assigning roles to activities, it is possible to take a view of the methodology from the point of view of a certain role. This answers questions such as “What are all the activities that I perform as a software developer?”
Use Case:	The Generate RTE Activity is performed by the role “Software Integration Engineer”.
Dependencies:	[RS_METH_0021] < Methodology shall define Activities > [RS_METH_0028] < Methodology shall define Roles >
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method [Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.4.6 [RS_METH_0066] < Activities shall include tools >

Initiator:	Ramy Asselin (General Motors)
Date:	14.01.2008
Short Description:	Activities shall include tools
Type:	**New**
Importance:	**High**
Description:	Activities shall include the names of tools used to help complete the activity.
Rationale:	By defining which tools are needed, the performers of the activity can ensure all the tools have been sourced and installed prior to beginning the activity. As well, the implementers of tools that are Autosar specific, have a clear understanding of what activities their tool should support and know what the input and output work products are available. This will help ensure interoperability of Autosar Tools.
Use Case:	The activity “Generate RTE” requires an RTE generator Tool and a compiler.
Dependencies:	RS_METH_0021
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method [Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.5 Work Products

5.5.1 [RS_METH_0025] < Methodology shall define Work products >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall define Work products
Type:	**New**
Importance:	**High**
Description:	Methodology shall define the work products necessary to build an Autosar System. Work products define a certain content and may exists as one or multiple files, which is not specified by Autosar.
Rationale:	A clear definition of what work products exists is necessary to allocate responsible roles and identify which activities consume, produce, or modify that work product.
Use Case:	According to SPEM, "A work product or artifact is anything produced, consumed, or modified by a process. It may be a piece of information, a document, a model, source code, and so on." Example work products include a software component description, an ECU instance description, a topology description, or the RTE source files.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main350] AUTOSAR methods shall be FMEA compatible

5.5.2 [RS_METH_0050] < Work products shall have a description>

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Work products shall have a description
Type:	**New**
Importance:	**High**
Description:	Each work product shall have a textual description describing its contents.
Rationale:	A description of the work product is necessary to form a common understanding of what the work product contains.
Use Case:	The AtomicSoftwareComponentType is the highest (most abstract) description level that corresponds to the Virtual Function Bus view. This Work Product includes the data types and interfaces. These components do not contain any other sub-components as they define the smallest granularity of components.
Dependencies:	[RS_METH_0025] < Methodology shall define Work products >
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main350] AUTOSAR methods shall be FMEA compatible [Main30] A SIL-3-Level compatible development process must be possible with AUTOSAR

5.5.3 [RS_METH_0051] < Work products shall have a reference(s) to metaclass(es) in the Autosar Metamodel >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Work products shall have a reference(s) to metaclass(es) in the Autosar Metamodel.
Type:	**New**
Importance:	**High**
Description:	The work product shall references to the Autosar Metaclasses that are used to exchange it. If no Metaclass exists, such as for source files, or requirements documents, then this requirement is exempt.
Rationale:	The Autosar Metaclasses provide a standard mechanism to exchange Autosar specific work products.
Use Case:	The AtomicSoftwareComponentType is exchanged using the AtomicSoftwareComponentType MetaClass.
Dependencies:	[RS_METH_0025] < Methodology shall define Work products >
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main350] AUTOSAR methods shall be FMEA compatible

5.5.4 [RS_METH_0052] < It must be possible to avoid duplication of data in Work Products >

Initiator:	Ramy Asselin (General Motors)
Date:	05.12.2007
Short Description:	It must be possible to avoid duplication of data in Work Products
Type:	**new**
Importance:	**High**
Description:	Work Products defined should avoid duplicating data already defined in other work products. Work Products defined in the methodology are “Logical” work products, not “physical” work products. So in your own process you can decide to include, for example, the interfaces and software component description in 1 physical file or duplicated in several files. However, in the Methodology it is defined only once, with duplications avoided.
Rationale:	Data duplication causes problems when iterating and changes are made inconsistently across all affected work products. Having one source of data eliminates this problem.
Use Case:	Two atomic software components exchange information via a Port Interface. Since the port interface referenced by both software components, it should treated as a separate work product in order to avoid defining it twice. Exceptions to this requirement arise when a work product undergoes a rule-based transformation such as compiling or code generation. In this case, the duplication in the resulting work products always remains consistent with the original source. For example, type definitions in object files are continuously updated based on the c type definition via the compiler. However it would be unwise to define the same type definition twice in two separate c header files.
Dependencies:	[RS_METH_0025] < Methodology shall define Work products >
Conflicts:	--

Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main350] AUTOSAR methods shall be FMEA compatible

5.5.5 [RS_METH_0054] < Work Products shall not have circular references with other work products >

Initiator:	Ramy Asselin (General Motors)
Date:	05.12.2007
Short Description:	Work Products shall not have circular references with other work products
Type:	**new**
Importance:	**High**
Description:	A circular reference exists when several work products sequentially reference each other and the last work product references the first. Work Products shall be defined such that circular references do not exist.
Rationale:	Work products with circular references are especially difficult to manage and exchange between parties and create problems when defining activities that create them.
Use Case:	A composition software component A aggregates other software component prototypes B and C, which may also be composition software components. These aggregated software components B and C and the software components they refer to can never reference the composition software component A.
Dependencies:	[RS_METH_0025] < Methodology shall define Work products >
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main350] AUTOSAR methods shall be FMEA compatible

5.5.6 [RS_METH_0061] < Methodology shall describe the change of existing work products >

Initiator:	Günther Reinhold (Bosch)
Date:	07.12.2007
Short Description:	Methodology shall describe the change of existing work products.
Type:	**new**
Importance:	**High**
Description:	Methodology shall include scenarios in which the output of an activity is not created newly, but updated from an older version of a work product.
Rationale:	Updating an existing work product may lead to a different process then creating a new one. For example, version information must be handled.
Use Case:	Software update: The methodology shall describe the workflow needed for SW updates and required conversions of already existing data (e.g. NVRAM data) of an AUTOSAR ECU.
Dependencies:	[RS_METH_0020] < Methodology shall support iterations >
Conflicts:	--
Supporting Material:	Feature 64 of methodology & configuration features

5.5.7 [RS_METH_0063] <Work Products shall be capable to be version controlled >

Initiator:	WP General Methodology and Configuration
Date:	14.01.2008
Short Description:	Work Products shall be capable to be version controlled
Type:	**new**
Importance:	**High**
Description:	It shall be possible to assign version numbers to work products and store them in a version control database.
Rationale:	This is necessary in order to support the creation of baselines which allow the recreation of an Autosar System.
Use Case:	A topology work product is defined. As new ECUs and networks are added and removed to the topology, the version number of the Topology work product is modified.
Dependencies:	[RS_METH_0025] < Methodology shall define Work products >
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups [Main350] AUTOSAR methods shall be FMEA compatible [Main30] A SIL-3-Level compatible development process must be possible with AUTOSAR

5.5.8 [RS_METH_0069] <It shall be possible to add Documentation to a Work Product>

Initiator:	Günther Reinhold (Bosch)
Date:	28.01.2008
Short Description:	It shall be possible to add precise and human readable documentation to each work product.
Type:	**new**
Importance:	**High**
Description:	The methodology shall allow that precise and human readable documentation be added to each work product. This documentation shall be either part of the work product or uniquely referred.
Rationale:	This is necessary in order to document design decisions or restrictions, which cannot obviously be deduced from the formal content, e.g. from names. Such documentation will increase the traceability which is demanded by quality or safety standards.
Use Case:	Choosing a redundancy mechanism, e.g. in the configuration for a NVRAM data block, may be related to a safety requirement. This may need verbal explanation.
Dependencies:	[RS_METH_0025] < Methodology shall define Work products >
Conflicts:	--
Supporting Material:	[Main30] A SIL-3-Level compatible development process must be possible with AUTOSAR [Main290] AUTOSAR shall ensure the verification of all processes and products developed within AUTOSAR [Main300] AUTOSAR shall support work-share in large inter-company development groups

5.6 Guidance

[RS_METH_0027] < Methodology shall define unambiguous guidance terminology >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall define unambiguous guidance terminology
Type:	**New**
Importance:	**medium**
Description:	Activities can refer on guidance to assist execution of the activity. Guidance can take the form of checklists, tutorials, or tools. When tools are used, a name shall be given to it. An actual tool implementation may perform many other tasks, but within the context of an activity, the tool only performs the named guidance.
Rationale:	Facilitate sourcing and comparison of various tools, especially when they perform many, likely different, sets of Autosar Activities.
Use Case:	For example, there is likely a need to have a guidance that helps configure the RTE, the RTE Generator. Similarly, there is likely a need to have a guidance that actually generate the RTE, the RTE generator. It can be foreseen that both of these guidance can be completed by the same tool or separate tools.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.6.1 [RS_METH_0042] < Methodology shall incorporate the usage of industry standard tools >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall incorporate the usage of industry standard tools
Type:	**New**
Importance:	**High**
Description:	Where industry standard tools, such as compilers and linkers exist, the Methodology shall incorporate them.
Rationale:	AUTOSAR should not required the use of tools when industry standard tools already exist.
Use Case:	Compilers are industry standard tools.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main90] Tool-chains, which are developed for or adapted to AUTOSAR, must be compatible with the AUTOSAR process

5.7 Roles

5.7.1 [RS_METH_0028] < Methodology shall define Roles >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology shall define Roles
Type:	**New**
Importance:	**low**
Description:	Methodology shall capture the typical roles people act within Autosar
Rationale:	The definition of Roles will allow distinguishing the responsibilities of different parties working in the AUTOSAR activities and assign ownership to a Work Product. It will also help to define the level of granularity for the activities and work products by considering the “owner” and the “performer”.
Use Case:	Example roles include, “Software Developer”, “Module Integrator”, and “Electrical Architect”.
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method

5.7.2 [RS_METH_0064] < Roles shall have a description >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Roles shall have a description
Type:	**New**
Importance:	**low**
Description:	Methodology shall capture the typical roles people act within Autosar
Rationale:	--
Use Case:	A “Software Developer” is an engineer proficient in modelling or programming languages and is capable of translating requirements into software and resolving defects.
Dependencies:	[RS_METH_0028] < Methodology shall define Roles >
Conflicts:	--
Supporting Material:	[Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method

5.8 Process Requirements

5.8.1 [RS_METH_0056] < AUTOSAR methodology shall not be bound to a particular lifecycle model >

Initiator:	Holger Diekmann (Hella)
Date:	23.10.2007
Short Description:	AUTOSAR methodology shall not be bound to a particular lifecycle model
Type:	**New**

Importance:	high | medium | low
Description:	AUTOSAR methodology shall not be bound to a particular lifecycle model. Activities must be independent with respect to the time and phase of the development process they are executed.
Rationale:	Connection to company specific lifecycle model: The methodology shall enable the use of different lifecycle models such as e.g. V-Model, Rational Unified Process.
Use Case:	If e.g. extreme programming is used, the test cases are created prior to the implementation. For most other development processes, the implementation is generated prior to the creation of test cases.
Dependencies:	--
Conflicts:	--
Supporting Material:	[RS_METH_0041] <Methodology shall support the Bottom/Up Approach > [Main300] AUTOSAR shall support work-share in large inter-company development groups [Main370] AUTOSAR process shall provide a predefinition of typical roles and activities in work-share method [Main30] A SIL-3-Level compatible development process must be possible with AUTOSAR
How to test	Test the dependencies of the execution sequence of the defined activities. Have all members review the methodology and ensure the Methodology fits within their process.

5.8.2 [RS_METH_0057] < AUTOSAR Methodology shall support traceability to external artifacts >

Initiator:	Holger Diekmann (Hella)
Date:	23.10.2007
Short Description:	AUTOSAR methodology shall support traceability to external artifacts
Type:	**New**
Importance:	high | medium | low
Description:	External artifacts are artifacts such as requirements or test cases that are not defined in the context of AUTOSAR. The mapping of external artifacts onto AUTOSAR work products must be possible.
Rationale:	Achieve traceability during the creation of the product.
Use Case:	E.g. the mapping of requirements to a) (the implementation of) a software component and b) the configuration of a software component
Dependencies:	--
Conflicts:	--
Supporting Material:	[RS_METH_0041] <Methodology shall support the Bottom/Up Approach > [Main300] AUTOSAR shall support work-share in large inter-company development groups [Main30] A SIL-3-Level compatible development process must be possible with AUTOSAR
How to test	Test the templates if it is possible to create relationships from artifacts to the work products or even parts of the work products.

	Test if work products exist (e.g. as a file?) and that a relationship can be created.

5.9 Development Requirements

5.9.1 [RS_METH_0009] < Methodology should be modeled >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology should be modeled
Type:	**new**
Importance:	**High**
Description:	Methodology should be modeled using consistent relationships.
Rationale:	Modelling the methodology will allow the relationships to be precise and will allow a tool to automatically generate the published document.
Use Case:	For example: The following pattern can be used, where each → defines a very specific relationship Work Product → Activity → Work Product ↑ Guidance
Dependencies:	--
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups

5.9.2 [RS_METH_0010] < Methodology should define rules to translate methodology model into a document >

Initiator:	Ramy Asselin (General Motors)
Date:	05.10.2007
Short Description:	Methodology should define rules to translate methodology model into a document
Type:	**new**
Importance:	**High**
Description:	Templates and rules can be applied to the modeled methodology to convert it into a document for publishing.
Rationale:	Using precise rules to automatically generate the methodology document will eliminate many sources of errors and inconsistencies.
Use Case:	Possible to generate Latex version of the document.
Dependencies:	[RS_METH_0009] < Methodology should be modeled >
Conflicts:	--
Supporting Material:	--

5.9.3 [RS_METH_0067] < Methodology document shall include hyperlinks between Activities, Roles, Work Products, and Guidance >

Initiator:	Ramy Asselin (General Motors)
Date:	14.01.2008
Short Description:	Methodology document shall include hyperlinks between Activities, Roles, Work Products, and Guidance.
Type:	**new**
Importance:	**High**
Description:	Cross references between the Activities, Roles, Work Products, and Guidance shall be hyperlinked.
Rationale:	By creating hyperlinks to all references, the Methodology can quickly answer many common queries such as “Show me all the work products Role X owns”, or “Show me all activities that require work product Y as an input”. This is very helpful to quickly learn the Methodology or to conduct a downstream impact analyses in order to identify the stakeholders.
Use Case:	Possible to generate Latex version of the Methodology.
Dependencies:	[RS_METH_0009] < Methodology should be modeled > [RS_METH_0010] < Methodology should define rules to translate methodology model into a document >
Conflicts:	--
Supporting Material:	[Main300] AUTOSAR shall support work-share in large inter-company development groups

5.10 Variant Handling Requirements

5.10.1 [RS_METH_0074] < Methodology shall specify Binding times>

Initiator:	WP II General Methodology and Configuration; Guenther Reinhold, Bernhard Weichel
Date:	2008-06-17
Short Description:	Methodology shall specify Binding times
Type:	--
Importance:	--
Description:	AUTOSAR Methodology shall specify particular points in the workflow on which variation can be resolved
Rationale:	Need for a stable reference on Binding times
Use Case:	During the development of an Autosar System and ECU, specific variants need to be created, and eventual chosen, e.g pre compile, or post build.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

5.10.2 [RS_METH_0075] < Methodology shall specify the tasks of resolving variant>

Initiator:	WP II General Methodology and Configuration; Guenther Reinhold, Bernhard Weichel
Date:	2008-06-17
Short Description:	Methodology shall specify the tasks of resolving variant

Type:	--
Importance:	--
Description:	AUTOSAR Methodology shall specify particular tasks/activities in which variation will be resolved
Rationale:	Need for clarification of methodology of variants
Use Case:	If two software components provide the same interface in different variants of the system, a task is needed to select the one provider to resolve that system variant.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

5.10.3 [RS_METH_0076] < Methodology shall specify a workproduct for values of variant selectors>

Initiator:	WP II General Methodology and Configuration; Guenther Reinhold, Bernhard Weichel
Date:	2008-06-17
Short Description:	Methodology shall specify specify a workproduct for values of variant selectors
Type:	--
Importance:	--
Description:	AUTOSAR Methodology shall specify particular workproducts to maintain the values of variant selectors.
Rationale:	This makes it clear where the values for variant selectors are stored and matained
Use Case:	The possible variants are known up front: they are created at a certain time and owned as a work product, and finally consumed when the variant is selected.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

6 References

In this section the references used in this specification are listed.

6.1 References to AUTOSAR documents

[1] Glossary
AUTOSAR_TR_Glossary.pdf

[2] Main requirements
AUTOSAR_RS_Main.pdf

[3] Meta Model
AUTOSAR_MMOD_MetaModel.eap

[4] Generic Structure Template
AUTOSAR_TPS_GenericStructureTemplate.pdf