Document Title	Requirements on Basic Software Module Description Template
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	086
Document Classification	Auxiliary

Document Version	1.1.1
Document Status	Final
Part of Release	4.0
Revision	3

Document Change History

Date	Version	Changed by	Change Description
23.09.2011	1.1.1	AUTOSAR Administration	• Added detailed change history (chapter 6)
30.11.2009	1.1.0	AUTOSAR Administration	• Added support for concepts: • AUTOSAR Scheduler harmonization • Triggered Event • Debugging • A2L Generation Support • Legal disclaimer revised
23.06.2008	1.0.1	AUTOSAR Administration	• Legal disclaimer revised
13.11.2007	1.0.0	AUTOSAR Administration	• Initial Release

Disclaimer

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 Document

This document collects the requirements on the Basic Software Module Description template (BSWMD-T).
The main goal of the BSWMD-T is to provide the scheme for the BSWMD. The BSWMD holds information about an implementation of a BSW module or cluster to support the integration on an ECU. Another use-case is to support of the conformance tests of BSW modules.

Three stages of a BSW module can be distinguished in the methodology:

- 'BSW module specification' is provided as the standard by AUTOSAR.
  The API may be a specified for all use-cases. The configuration parameter may have a wide range of configuration possibilities. Certain crucial configuration parameters may be missing because of hardware dependencies which can not be described in the specification.
- 'BSW module implementation' is one possible implementation of the BSW module specification.
  Only a subset of the specified API may be implemented. Several configuration decisions may have been taken however other configuration parameters are still open for the integrator to choose.
  Vendor-specific configuration parameters may be added in order to allow the configuration of the module's behavior (applicable for all BSW modules), and/or to support the configuration of specific hardware elements, like special register settings (applicable to hardware dependent modules only).
- 'Configured BSW module' takes the still open configuration parameters from a concrete BSW module implementation and assigns values to them. The fully configured BSW module can be actually integrated on an ECU.

Each BSW module implementation does come with an own BSW Module Description. It is important to always use the right pair of BSW module implementation and the according BSWMD.

In Figure 1-1 the inputs to the activity 'Configure ECU' are shown:

- 'Collection of Available Software Components' contains references to all the descriptions of Software Components mapped to this specific ECU
- 'ECU Extract of System Description' contains the subset of the System Configuration which is relevant for this specific ECU. This includes the communication matrix and the data to signal mapping.
- 'BSW Module Description' (requirements are collected in this document).

The output is the 'ECU Configuration Description'.

Due to the high configurability of some BSW Modules the BSWMD is not capable of capturing all dependencies of the BSW Module's configuration. Therefore it is also possible to update the BSWMD after the BSW Module has been configured and generated to provide more specific information about the configured BSW Module.

One example of a highly configurable BSW Module is the RTE which is almost completely generated and the initial BSWMD - delivered with the unconfigured RTE – can not describe all possible configurations of the RTE in a formal way.
But after the RTE is configured its BSWMD can be updated to contain descriptions of the actual to be generated RTE. This updated BSWMD can then be used to aid in the configuration of other BSW Modules like Os, Debugger, Dlt.

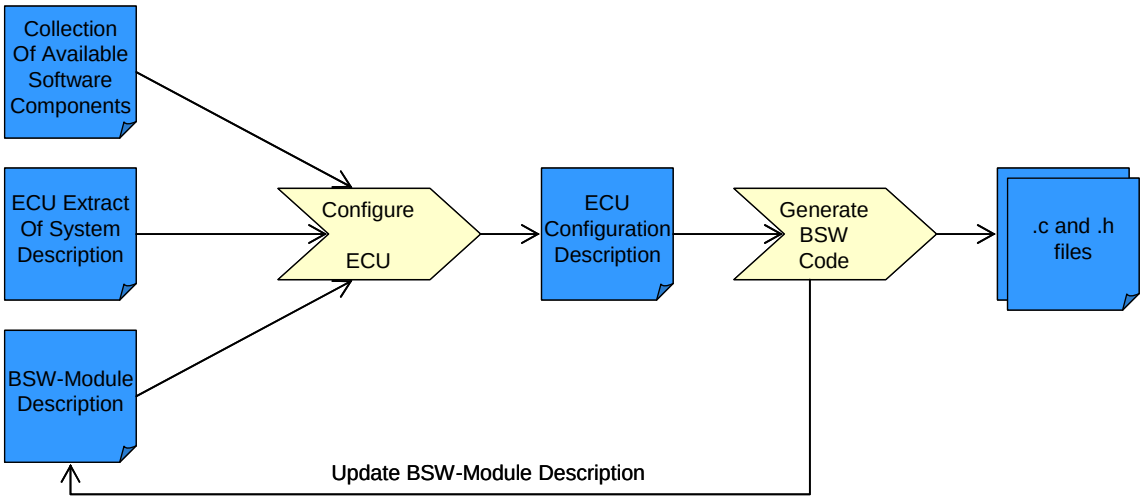

Figure 1-1: Overview ECU Configuration

The BSWMD-template specifies which content the actual Basic Software Module Description (BSWMD) is able to provide. From a technical point of view the template is provided as a document [13] and a XML schema [14] (see also [4] and [5]).
The actual Basic Software Module Descriptions are XML files which conform to the XML schema.

2 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.)

3 Related Documentation

3.1 Input Documents

The following input documents have been used in the development of these requirements:

- General Requirements on Basic Software Modules [2]
- Requirements on ECU Configuration [9]
- AUTOSAR RTE Software Requirement Specification [3]
- AUTOSAR Methodology [7]
- AUTOSAR Glossary [1]
- Technical Overview [8]
- AUTOSAR Generic Structure Template [4]
- AUTOSAR Model Persistence Rules for XML [5]

3.2 Specification Documents

The requirements collected in this document will be satisfied by two specification documents:

- Specification of the BSW Module Description Template
  This document implements most of the requirements stated here.
- ECU Configuration Specification [10]
  This document provides specification and guidelines for the creation of Standardized and Vendor-Specific Module Definitions.

3.3 Abbreviations and terms

Abbreviation	***Meaning***
BSW	Basic Software
BSWMD	Basic Software Module Description
BSWMD-T	Basic Software Module Description Template
ECUC	ECU Configuration Values [10]
ECUC Parameter Definition	ECU Configuration Parameter Definition [10]
ECUC-T	ECU Configuration Template [10]
ICS	Implementation Conformance Statement
StMD	Standardized Module Definition [10]
SWC	Software Component Description
SWC-T	Software Component Template
Vendor-Specific ECUC Parameter Definition	Vendor-Specific ECU Configuration Parameter Definition [10]
VSMD	Vendor-Specific Module Definition [10]

4 Requirements on BSW Module Description Template

4.1 Published Information

4.1.1 [BSWMD0043] Support description of common published information

Initiator:	WP Methodology and Templates
Date:	23.01.2007
Short Description:	Support description of common published information
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe the common published information provided by BSW module implementations according to the respective BSW SWS.
Rationale:	Configuration tools SHALL be able to read the common published information of a BSW implementation, as ECU Configuration Values may depend on common published information.
Use Case:	Providing common published information like: Module VERSION, REVISION number or AUTOSAR specification number.
Dependencies:	BSWMD0024
Conflicts:	--
Supporting Material:	--

4.1.2 [BSWMD0024] Support description of module specific published information

Initiator:	WP Methodology and Templates
Date:	23.03.2005
Short Description:	Support description of module specific published information
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe the module specific published information provided by BSW module implementations according to the respective BSW SWS.
Rationale:	Configuration tools SHALL be able to read the published information of a BSW implementation as ECU Configuration Values may depend on published information.
Use Case:	Make the value of MEMIF_BROADCAST_ID available to other modules (e.g. to NvM). Make the values of hardware dependent information like: EEPROM-ERASE-TIME or the width of API parameters like EEP-IF-ADDRESSTYPE (uint8, 16, 32) available to other modules (e.g. to MemIf).
Dependencies:	BSWMD0007, BSWMD0043
Conflicts:	--
Supporting Material:	--

4.1.3 [BSWMD0039] Identification of implemented API and functions

Initiator:	WP Conformance Test
Date:	04.12.2006
Short Description:	Identification of implemented API and functions
Type:	New
Importance:	High
Description:	Describes which API and functions are actually implemented by the BSW module/cluster.
Rationale:	The specifications of the BSW modules allow implementing only a subset of the specified APIs and functions. Which subset is actually implemented SHALL be described.
Use Case:	Conformance of a module (cluster) can only be attested for the functionality that a module/cluster actually provides.
Dependencies:	BSWMD0040, BSWMD0041
Conflicts:	--
Supporting Material:	--

4.1.4 [BSWMD0040] Identification of required API and functions

Initiator:	WP Conformance Test
Date:	04.12.2006
Short Description:	Identification of required API and functions
Type:	New
Importance:	High
Description:	Describes which API and functions of other modules this implementation requires.
Rationale:	Support the integration by listing the actually needed APIs which are used by this implementation.
Use Case:	Check whether the provided API, functions and operation signatures of other modules match the requirements of the BSW module implementation.
Dependencies:	BSWMD0039, BSWMD0041, BSWMD0047
Conflicts:	--
Supporting Material:	--

4.1.5 [BSWMD0041] Declaration of the provided API argument data types

Initiator:	WP Conformance Test
Date:	04.12.2006
Short Description:	Declaration of the provided API argument data types
Type:	New
Importance:	High
Description:	Description of the actual data types used by the implementation for API function arguments and ECU Configuration Parameter Definitions which have been left open in the specification documents.
Rationale:	The specifications of the BSW modules in some cases do not fix the data type to be used for the implementation. To allow the integration the actually implemented data types need to be described.
Use Case:	If the BSW SWS does specify an API argument to be either UInt8 or UInt16 the BSWMD template SHALL provide means to describe which type has been used in the actual implementation.
Dependencies:	--

Conflicts:	--
Supporting Material:	--

4.1.6 [BSWMD0042] Description of the required API argument data types

Initiator:	WP Conformance Test
Date:	04.12.2006
Short Description:	Description of the required API argument data types
Type:	New
Importance:	High
Description:	Description of the actual data types required by the implementation for API function arguments which have been left open in the specification documents.
Rationale:	The specifications of the BSW modules in some cases do not fix the data type to be used for the implementation. To allow the integration these actually implemented data types need to be described.
Use Case:	If the BSW SWS does specify an API argument to be either UInt8 or UInt16 the BSWMD template SHALL provide means to describe which type is expected in the actual implementation.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.7 [BSWMD0011] Guaranteed execution context of API calls

Initiator:	WP Methodology and Templates
Date:	18.03.2005
Short Description:	Guaranteed execution context of API calls
Type:	New
Importance:	High
Description:	For API calls to other modules it SHALL be possible to describe whether the call will be executed in interrupt context by the caller.
Rationale:	If both, the caller and the callee specify the context of the call it is possible to detect invalid call chains during ECU configuration activity. If a call is happening in interrupt context it has some restrictions on execution time and available instructions. The RTE Generator needs to know the context of calls from the BSW services to be able to decouple interrupt context from the application SW-Components.
Use Case:	The Com module expects the notifications from PduR happening in task context, but the PduR just handles the interrupt context which is coming from the CanIf. This is an invalid configuration and shall be detected.
Dependencies:	BSWMD0038, BSWMD0040, BSW00326 [2]
Conflicts:	--
Supporting Material:	--

4.1.8 [BSWMD0038] Required execution context of API calls

Initiator:	WP Methodology and Templates
Date:	11.10.2006
Short Description:	Required execution context of API calls

Type:	New
Importance:	High
Description:	A BSWMD template SHALL provide means to define the context for each provided API function in which it shall be invoked.
Rationale:	If both, the caller and the callee specify the context of the call it is possible to detect invalid call chains during ECU configuration activity.
Use Case:	The Com module expects the notifications from PduR happening in task context, but the PduR just handles the interrupt context which is coming from the CanIf. This is an invalid configuration and shall be detected.
Dependencies:	BSWMD0011, BSWMD0039, BSW00326 [2]
Conflicts:	--
Supporting Material:	--

4.1.9 [BSWMD0010] Compiler version and settings

Initiator:	WP Methodology and Templates
Date:	18.04.2005
Short Description:	Compiler version and settings
Type:	New
Importance:	High
Description:	It SHALL be possible to describe the actual compiler (vendor, version), and its settings, which has been used for object code delivery or which needs to be used for source code delivery.
Rationale:	When BSW is delivered as object code the integrator needs to know how the object code has been compiled. If it is delivered as source code, the code is often provided for specific compilers and versions.
Use Case:	Object code compiled with different compilers may have issues in the stack structure. Therefore the used compiler and its settings have to be described in order to detect such inconsistencies.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.10 [BSWMD0037] Needed libraries

Initiator:	WP Methodology and Templates
Date:	11.10.2006
Short Description:	Needed libraries
Type:	New
Importance:	High
Description:	It SHALL be possible to describe which libraries (vendor and version) have been used for object code deliveries or which need to be included for source code deliveries.
Rationale:	When a BSW module is delivered as object code the integrator needs to know how the object code has to be integrated. If it is delivered as source code, the code may need a specific version of expected libraries only.
Use Case:	If several BSW modules use the same library it only needs to be present on the ECU once. Describe the used libraries and version in order to be able to detect whether the libraries used by several BSW module implementations are not compatible.

Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.11 [BSWMD0025] Support for shipment information

Initiator:	WP Methodology and Templates
Date:	26.07.2005
Short Description:	Support for shipment information
Type:	new
Importance:	medium
Description:	The BSWMD template SHALL support the description which files (source, object, documentation) are included in the delivery of the BSW module.
Rationale:	Describe which artifacts are shipped in the delivery of the BSW module.
Use Case:	Check for completeness of the delivered artifacts before integration.
Dependencies:	BSWMD0044
Conflicts:	--
Supporting Material:	--

4.1.12 [BSWMD0014] Support of BSW Module clusters

Initiator:	WP Methodology and Templates
Date:	30.05.2007
Short Description:	Support of BSW Module clusters
Type:	New
Importance:	High
Description:	Support the description of BSW module clusters which implement several BSW modules.
Rationale:	AUTOSAR allows integrating several BSW modules (or even the whole BSW including the AUTOSAR Services) in a single cluster, treating this BSW cluster as one entity. It must be known how the cluster interacts with other modules / clusters in order to integrate it. Tests for clusters must know what parts (operation signatures and configurable functionality) are actually supported by the object under test.
Use Case:	Delivery of complete COM stack in a single implementation. Delivery of the whole AUTOSAR BSW in a single implementation.
Dependencies:	--
Conflicts:	There is no definition what the AUTOSAR clustering is - therefore this requirement is rephrased to be more generic.
Supporting Material:	--

4.1.13 [BSWMD0034] ECU Configuration Editor and Generation supported tool version information

Initiator:	WP Methodology and Templates
Date:	19.09.2006
Short Description:	ECU Configuration Editor and Generation supported tool version information
Type:	New
Importance:	Medium

Description:	It SHALL be possible to describe the supported ECU Configuration editor and generator tool (vendor, version) and its settings.
Rationale:	When a BSW module is delivered the integrator needs to know which editing and generation tools can be used to configure the BSW.
Use Case:	Since the BSW module implementation may need some vendor-specific ECU Configuration Parameter handling it SHALL be possible to state which generator can cope with these extensions.
Dependencies:	--
Conflicts:	--
Supporting Material:	This requirement does not exclude tools not explicitly listed from working with the specific XML file.

4.1.14 [BSWMD0013] Describe configuration class of ECU Configuration Parameters

Initiator:	WP Methodology and Templates
Date:	21.04.2005
Short Description:	Describe configuration class of ECU Configuration Parameters
Type:	New
Importance:	High
Description:	When the actual implementation of a BSW module has the freedom to choose the configuration class (pre-compile, link-time, post-build) it SHALL be possible to describe which alternative has been chosen.
Rationale:	An ECU Configuration Parameter needs to be handled differently depending on its configuration class.
Use Case:	The ECU Configuration editor SHALL be able to only allow changes on the post-build time ECU Configuration Parameters.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.15 [BSWMD0033] Pre-configured ECU Configuration Values

Initiator:	WP Methodology and Templates
Date:	07.02.2006
Short Description:	Pre-configured ECU Configuration Values
Type:	New
Importance:	High
Description:	The BSWMD template SHALL allow specifying a set of ECU Configuration Values which have been set to fixed values by the implementation.
Rationale:	Pre-configured ECU Configuration Values contain values which cannot be altered by the BSW module integrator, since they are fixed by the implementation. These pre-configured ECU Configuration Values shall be copied into the ECU Configuration Values of the actual BSW module as part of the base module configuration once the module implementation has been chosen [10].
Use Case:	A value may be fixed for different reasons. E.g. all pre-compile parameters are fixed in object code deliveries.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.16 [BSWMD0032] Recommended ECU Configuration Values

Initiator:	WP Methodology and Templates
Date:	07.02.2006
Short Description:	Recommended ECU Configuration Values
Type:	New
Importance:	Low
Description:	The BSWMD template SHALL allow specifying a set of ECU Configuration Values which are recommended by the implementation.
Rationale:	These Recommended ECU Configuration Values may hold the ECU Configuration Values recommended by the implementer and may be copied into the ECU Configuration Values of the BSW module as base, once the BSW module implementation has been chosen. The Recommended ECU Configuration Values are more flexible than default values, since they allow defining several container instances with different ECU Configuration Parameter values in each container [10].
Use Case:	Allow BSW vendors to deliver a partial or complete ECU Configuration Values file of the BSW module together with the implementation. This eases the work of the integrator who only needs to fill in the missing ECU Configuration Values.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.17 [BSWMD0035] Provide Standardized Module Definition

Initiator:	WP Methodology and Templates
Date:	19.09.2006
Short Description:	Provide Standardized Module Definition
Type:	Changed (02.07.2009)
Importance:	High
Description:	The BSWMD template SHALL allow the specification of the module's Standardized Module Definition.
Rationale:	The Standardized Module Definition is the base for the configuration of the BSW module. Out of the Standardized Module Definition the Vendor-Specific Module Definition is derived.
Use Case:	Provide information which Standardized Module Definition is used with a certain BSW module implementation.
Dependencies:	BSWMD0048
Conflicts:	--
Supporting Material:	--

4.1.18 [BSWMD0050] Allow vendor-specific modification of Standardized Module Definition

Initiator:	WP Methodology and Templates
Date:	29.10.2007
Short Description:	Allow vendor-specific modification of Standardized Module Definition
Type:	Changed (02.07.2009)
Importance:	High
Description:	The BSWMD template SHALL allow modification to the Standardized

	Module Definition in order to support implementation specific adaptations.
Rationale:	The Standardized Module Definition does specify the superset of possible values per configuration parameter. A certain implementation may restrict the actual applicable characteristics of individual elements in the Standardized Module Definition.
Use Case:	The BlockId of the NvRam manager may either be 8 or 16 bit. The standardized parameter has a min value of 1 and a max value of 65535. An implementation may choose to support only the 8 bit values, so the max value has to be adapted to be 255.
Dependencies:	BSWMD0035
Conflicts:	--
Supporting Material:	--

4.1.19 [BSWMD0027] Provide Vendor-Specific Module Definition

Initiator:	WP Methodology and Templates
Date:	26.07.2005
Short Description:	Provide Vendor-Specific Module Definition
Type:	New
Importance:	High
Description:	The BSWMD template SHALL allow the definition of the module's Vendor-Specific Module Definition.
Rationale:	Additional configuration parameters are necessary because the Standardized Module Definition does not contain all relevant configuration parameters for certain module implementations. Due to the fact that the configuration parameters are hardware dependent the specific implementation does require additional configuration definition. The Vendor-Specific Module Definition specifies which configuration parameters and ranges are actually supported by a concrete implementation of the BSW module.
Use Case:	Vendor-specific configuration parameters may be added in order to allow the configuration of the module's behavior (applicable for all BSW modules), and/or to support the configuration of specific hardware elements, like special register settings (applicable to hardware dependent modules only).
Dependencies:	For the definition of Vendor-specific parameters the ECU Configuration Parameter Definition template SHALL be used [10]. BSWMD0048
Conflicts:	--
Supporting Material:	--

4.1.20 [BSWMD0007] Provide vendor-specific published information

Initiator:	WP Methodology and Templates
Date:	04.04.2005
Short Description:	Provide vendor-specific published information
Type:	New
Importance:	Low
Description:	BSWMD template SHALL support the definition of vendor-specific published information.
Rationale:	Vendors may want to publish proprietary information for usage in their tool chain.
Use Case:	Describe vendor-specific published information on the actual implementation

	of the module based on the implemented target. Avoid proprietary means to provide such extensions by specifying a standardized way of description.
Dependencies:	BSWMD0048, BSWMD0024
Conflicts:	This is diluting the standard; however there is a need for such extensions.
Supporting Material:	ECUC0002 [9]

4.1.21 [BSWMD0048] Tagging of Vendor-Specific Module Definition

Initiator:	WP Conformance Test
Date:	30.05.2007
Short Description:	Tagging of Vendor-Specific Module Definition
Type:	New
Importance:	High
Description:	It SHALL be possible to distinguish between Standardized and Vendor-Specific Module Definitions.
Rationale:	Since it is possible for vendors to add vendor specific ECU Configuration Parameters to the Standardized Module Definition those additions need to be distinguished from the standardized ECU Configuration Parameters.
Use Case:	In order to check the Vendor-Specific Module Definition for conformance it needs to be described which ECU Configuration Parameters are standardized and which are vendor-specific.
Dependencies:	BSWMD0035, BSWMD0027, BSWMD0007
Conflicts:	--
Supporting Material:	--

4.1.22 [BSWMD0047] Modeling of call-chain dependencies between BSW Modules

Initiator:	WP Methodology and Templates
Date:	23.01.2007
Short Description:	Modeling of call-chain dependencies between BSW Modules
Type:	New
Importance:	Medium
Description:	It SHALL be possible to describe which other API functions are invoked by a function.
Rationale:	Needed when the OS is configured since OS resources must be mapped to the tasks that utilize them.
Use Case:	Derive which OS resources are used when a main function is invoked and this main function is invoking another API function, and so on.
Dependencies:	BSWMD0046
Conflicts:	--
Supporting Material:	--

4.1.23 [BSWMD0049] Describe optional and required elements

Initiator:	WP Conformance Test
Date:	01.07.2008
Short Description:	Describe optional and required elements
Type:	New
Importance:	High

Description:	The BSWMD template SHALL provide means to describe optional and required elements of a BSW module implementation.
Rationale:	Due to the high configurability of the BSW modules – originating from the AUTOSAR specifications – there SHALL be means to describe the actually supported elements of one BSW module implementation.
Use Case:	Statement which optional elements a BSW module implementation supports. Shipment of a BSW module description for production use containing optional elements, which shall be selected later by the integrator. Using a BSW module description which describes the standard including mandatory and optional elements as a reference for conformance checks of other BSW descriptions.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.24 [BSWMD0044] Description of generated artifacts

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Description of generated artifacts
Type:	New
Importance:	High
Description:	Support the description of which artifacts a generation tool will create.
Rationale:	The knowledge on which artifacts (header- and c-files, documentation) are generated by the BSW module's generation tool does support integration and build process.
Use Case:	Generate the make-file based on the information from BSWMD0025 and the generated artifacts.
Dependencies:	BSWMD0025
Conflicts:	--
Supporting Material:	[BRF00106] Generic Build System [15]

4.1.25 [BSWMD0051] Description of libraries

Initiator:	WP Methodology and Templates
Date:	07.04.2008
Short Description:	Description of libraries
Type:	New
Importance:	High
Description:	It SHALL be possible to describe libraries and their implementations.
Rationale:	Libraries are used to share code between several users in the BSW and in Application SW-Components. There SHALL be support for the selection and integration of libraries.
Use Case:	Describe the provided APIs of libraries Describe the required APIs of libraries Describe the resource needs of libraries.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.26 [BSWMD0052] Description of the generated RTE

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Description of the generated RTE
Type:	New
Importance:	High
Description:	It shall be possible to describe the attributes of the generated RTE.
Rationale:	The RTE generator is capable of taking many decisions which are affecting the integration of the RTE on an ECU.
Use Case:	Describe the used memory sections of the generated RTE. Describe the generated VFB tracing functions. Describe the resource needs of the actual RTE.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.1.27 [BSWMD0061] Support for Debugging of variables

Initiator:	WP Debugging
Date:	02.10.2008
Short Description:	Support for Debugging of variables
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe the variables of the BSW Module which can be debugged using the AUTOSAR Debugger Module.
Rationale:	For the configuration of the Debugger BSW Module the “to be debugged” Module shall provide information what actually can be debugged.
Use Case:	Provide a list of internal variables of the BSW Module which can be observed by the debugger.
Dependencies:	The API calls from any BSW Module towards to Debugger Module can be described with BSWMD0040
Conflicts:	--
Supporting Material:	[BRF00152] BSW Variables becomes accessible by external Debuggers [15]

4.1.28 [BSWMD0062] Provide Measurement and Calibration Support

Initiator:	WP Methodology and Templates
Date:	17.03.2009
Short Description:	Provide Measurement and Calibration Support
Type:	New
Importance:	High
Description:	The BSWMDT shall contain a support format for the description of measurement and calibration data, which can be used (together with linker generated information) by external tools to generate the data descriptions required by calibration and measurement tools.
Rationale:	• For the code generated by the RTE, the contained calibration and measurements data are described in several "upstream" artifacts. An external tool shall be able to do the further processing out of a more simple artifact which contains only the relevant information.

	• An external tool must be able to determine memory addresses of measurement & calibration data. For this, the actual linker symbols of variables and parameters must be made available in the support format. • Information from the ECU configuration (e.g. calibration method by the RTE) must be made available too.
Use Case:	• The RTE generates support data for the measurement and calibration data as part of its own BSWMD, e.g. for data elements in ports. • The RTE (or another tool) generates support data for measurement and calibration data which are declared locally in a module. The BSWMD is updated with the generated support data.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.2 BSW Scheduling

4.2.1 [BSWMD0053] Cyclic time based scheduling of BSW Main Functions

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Cyclic time based scheduling of BSW Main Functions
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe the cyclic time based scheduling requirements of BSW Main Functions.
Rationale:	The RTE Generator creates the scheduling for the whole ECU. Many BSW Modules rely on the cyclic time based call of their main functions in order to fulfill their functionality. The RTE Generator shall be able to implement the cyclic time based call based on the stated requirements.
Use Case:	Call of the function “Com_MainFunctionTx()” from the generated RTE to achieve periodic sending of IPdus.
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00020] Integration of existing BSW Scheduling into the RTE

4.2.2 [BSWMD0054] Mode Switches for BSW modules shall be supported

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Mode Switches for BSW modules shall be supported
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe Mode Switches for BSW modules and the requirements on the scheduling.
Rationale:	Conditional scheduling of BSW Main Functions dependent on different operating modes of the ECU. BSW Main Functions are scheduled dependent on modes activated by entering or exiting a mode

	activated at particular mode transitions
Use Case:	Initialization and finalization phases (Mode provided by the EcuM) Different communication modes (Mode provided by the ComM)
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00020] Integration of existing BSW Scheduling into the RTE [BRF00260] Support of at Runtime dynamically schedulable BSW Modules

4.2.3 [BSWMD0055] Simultaneous Mode transitions

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Simultaneous Mode transitions
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to specify the required simultaneous switching of a Mode controlling BSW Modules and Application Software Components.
Rationale:	Synchronized behavior during a mode transition controlling AUTOSAR BSW Modules and Application Software Components
Use Case:	ECU global Initialization and finalization phase
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00020] Integration of existing BSW Scheduling into the RTE

4.2.4 [BSWMD0056] API for Mode switch notification of BSW modules

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	API for Mode switch notification of BSW modules
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe that a particular BswModuleEntity communicates modes.
Rationale:	The code generator for the BSW Scheduler shall generate the Mode Switch API used by the BSW module service as Mode Manager.
Use Case:	EcuM communicates the ECU's operating state to all BSW modules via BSW Scheduler.
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00020] Integration of existing BSW Scheduling into the RTE

4.2.5 [BSWMD0057] Triggering of BSW Main Functions by Triggered Events

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Triggering of BSW Main Functions by Triggered Events
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe the triggering of

	BSW Main Functions by the occurrence of Triggered Events. Particular BSW Main Functions in BSW dependent from Triggered Events shall be executed after occurrence of the event. The occurrence of the Triggered Event is either reported via API to the BSW Scheduler or by means of the OS (e.g. expiration of an OS Alarm). Restriction: This is only applicable for intra-ECU usage.
Rationale:	Sporadic and non timing based periodic activation of BSW Main Functions in different BSW Modules.
Use Case:	Angle periodic triggering of the ignition for a combustion engine.
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00031] Triggered Event [BRF00020] Integration of existing BSW Scheduling into the RTE

4.2.6 [BSWMD0058] Synchronized Triggering by Triggered Events

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Simultaneous Triggering by Triggered Events
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to specify the required synchronized triggering of Runnable Entities and BSW Main Functions by shared Triggered Events.
Rationale:	Synchronized activation of routines in AUTOSAR BSW modules and Application Software Components.
Use Case:	Angle periodic triggering of the routines in Application Software Components and Complex Device Drivers for a combustion engine.
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00031] Triggered Event [BRF00020] Integration of existing BSW Scheduling into the RTE

4.2.7 [BSWMD0059] API for Triggering BSW modules by Triggered Events

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	API for Triggering BSW modules by Triggered Events
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to describe that a particular BswModuleEntity raises Triggered Events.
Rationale:	The code generator for the BSW Scheduler shall generate the Trigger API used by the BSW module capturing the source of the Triggered Events.
Use Case:	Angle periodic triggering of the routines in Application Software Components and Complex Device Drivers for a combustion engine.
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00031] Triggered Event [BRF00020] Integration of existing BSW Scheduling into the RTE

4.2.8 [BSWMD0060] Support exclusive areas in BSW Modules and Application Software Components

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Support exclusive areas in BSW Modules and Application Software Components
Type:	New
Importance:	High
Description:	The BSWMD template SHALL provide means to define exclusive areas used by particular BswModuleEntitys to allow priority determination for preventing simultaneous access to shared resources. The exclusive areas shall be defined with a name and the accessing BswModuleEntitys. Exclusive areas shall only protect module internal data.
Rationale:	Decouple the module implementation from applying data consistency mechanisms. The code generator for BSW Scheduler shall provide APIs to enter or exit exclusive areas for BSW Modules.
Use Case:	Provide data consistency for a data buffer shared between a BswSchedulableEntity and a BswInterruptEntity.
Dependencies:	--
Conflicts:	--
Supporting Material:	[BRF00020] Integration of existing BSW Scheduling into the RTE

4.3 Resources

4.3.1 [BSWMD0005] Description of the memory needs of the BSW Module implementation

Initiator:	WP Methodology and Templates
Date:	23.03.2005
Short Description:	Description of the memory needs of the BSW Module implementation
Type:	New
Importance:	Low
Description:	The BSWMD template SHALL support the description of the memory needs of an implementation of a BSW module. Also specification of the quality (e.g. estimate, measurement, analysis) of these values SHALL be supported. The memory requirements of the defined memory sections SHALL be described individually.
Rationale:	Resource estimations/measurements are needed to design & configure the ECU.
Use Case:	The ROM utilization of BSW modules delivered as object code is typically fixed and can be stated in the BSWMD. In most cases the memory needs are dependent on the actual ECU Configuration Parameter Values and can only be estimated.
Dependencies:	BSWMD0031
Conflicts:	--
Supporting Material:	--

4.3.2 [BSWMD0031] Description of used memory section names

Initiator:	WP Methodology and Templates
Date:	25.02.2005
Short Description:	Description of used memory section names
Type:	new
Importance:	high
Description:	Support the description of memory section names which have been used while developing/compiling the BSW module.
Rationale:	With the usage of memory section names it is possible to partition the software into several sections which will be placed into memory sections on the ECU in the ECU Configuration activity.
Use Case:	The ECU State Manager implementation uses the memory section NOINIT to indicate which declared variables SHALL not be initialized during ECU startup. It is up to the ECU Configuration activity to actually map this section in an appropriate memory section on the ECU which satisfies this requirement.
Dependencies:	ECUC0068 [9], BSWMD0005
Conflicts:	--
Supporting Material:	Specification of Memory Mapping [6]

4.3.3 [BSWMD0009] Description of peripheral register usage

Initiator:	WP Methodology and Templates
Date:	04.04.2005
Short Description:	Description of peripheral register usage
Type:	new
Importance:	medium
Description:	The BSWMD template SHALL support ECU Configuration tools in determining conflicts between different BSW modules accessing the same peripheral register. In some cases these needs are dependent on the actual ECU Configuration Parameter Values (no formula SHOULD be provided in that case!).
Rationale:	BSW module implementations from different vendors may use conflicting configuration of peripheral registers. When these BSW modules are integrated in the same ECU then the ECU Configuration tool SHOULD detect these conflicts and alert the user.
Use Case:	Two BSW module implementations both writing to the same microcontroller register but using a different setting. Conflict must be identified.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.3.4 [BSWMD0016] Timing guarantees

Initiator:	WP Methodology and Templates
Date:	23.03.2005
Short Description:	Timing guarantees
Type:	New
Importance:	Low
Description:	The BSWMD template SHALL allow specifying the guaranteed or estimated reaction time of the BSW module functions (main functions and API calls

	incl. callbacks & ISR).
Rationale:	To be able to do timing analysis of Application SW-Components, the BSW needs to define timing guarantees.
Use Case:	With the knowledge of the guaranteed execution time the design of exclusive area access can be optimized, depending on the duration an interrupt block might be possible.
Dependencies:	SW-Component template requirement CONTENT080
Conflicts:	--
Supporting Material:	--

4.3.5 [BSWMD0015] Timing requirements

Initiator:	WP Methodology and Templates
Date:	01.07.2008
Short Description:	Timing requirements
Type:	New
Importance:	High
Description:	The BSWMD template SHALL allow specifying the timing requirements on functions called in other modules such as callback functions.
Rationale:	To be able to do timing analysis of Application SW-Components, the BSW needs to define timing requirements additional to the Timing guarantees BSWMD0016.
Use Case:	In order to fulfill a given timing guarantee the calls to other functions need to be restricted in time.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.3.6 [BSWMD0030] Publish resource needs for the BSW Scheduler

Initiator:	WP Methodology and Templates
Date:	30.05.2007
Short Description:	Publish resource needs for the BSW Scheduler
Type:	New
Importance:	High
Description:	BSWMD template SHALL provide means to describe the resources used by the implementation which need to be provided and integrated by the BSW Scheduler [12].
Rationale:	The BSW Scheduler is used to abstract the usage of concrete OS mechanism from the abstract concepts. The abstract concepts are describe in the BSW Scheduler specification [12]. The BSWMD template SHALL provide means to describe the needs on the BSW Scheduler from the BSW module implementation. But which actual mechanism is used to fulfill these needs is up to the implementation of the BSW Scheduler.
Use Case:	A BSW module is using an exclusive area access in its implementation and need to describe this usage, but it is up to the BSW Scheduler how this exclusive area access is actually implemented (using global interrupt blocking or an OS resource).
Dependencies:	BSWMD0046 [12]
Conflicts:	--
Supporting Material:	--

4.3.7 [BSWMD0046] Publish OS resource usage

Initiator:	WP Methodology and Templates
Date:	23.01.2007
Short Description:	Publish OS resource usage
Type:	new
Importance:	medium
Description:	For every function (main, API, ISR) it SHALL be possible to describe the OS resources used within the function.
Rationale:	To configure the OS correctly the access to OS resources has to be specified for every function. The BSW Scheduler must be able to resolve the task context in which any OS resource may be used.
Use Case:	Configure the OS with the right OS resource accesses.
Dependencies:	BSWMD0030, BSWMD0047
Conflicts:	--
Supporting Material:	--

4.3.8 [BSWMD0045] Publish resources needed from AUTOSAR Services

Initiator:	WP Methodology and Templates
Date:	23.01.2007
Short Description:	Publish resources needed from AUTOSAR Services
Type:	new
Importance:	medium
Description:	If a BSW module needs resources from AUTOSAR Services, the needs have to be described.
Rationale:	To allow the ECU Configuration activity of the AUTOSAR Services, the needs from BSW and Application SW Components have to be captured.
Use Case:	When a BSW module requires some NVRAM space it has to provide a description of the attributes this NVRAM has to have.
Dependencies:	--
Conflicts:	--
Supporting Material:	--

4.3.9 [BSWMD0026] Description of supported hardware

Initiator:	WP Methodology and Templates
Date:	26.07.2005
Short Description:	Description of supported hardware
Type:	new
Importance:	medium
Description:	For BSW modules that are hardware dependent (like the drivers) the supported hardware SHALL be described.
Rationale:	Certain software modules can only be integrated on specific hardware.
Use Case:	When the supported hardware is specified a selection of drivers for a certain hardware can be provided.
Dependencies:	The characterization SHOULD be done by referencing the ECU Resource Description.
Conflicts:	--
Supporting Material:	--

4.4 Requirements on the Template

4.4.1 [BSWMD0001] Main source of information on BSW Module ECU Configuration activity and integration

Initiator:	WP Methodology and Templates
Date:	23.03.2005
Short Description:	Main source of information on BSW Module ECU Configuration activity and integration
Type:	new
Importance:	medium
Description:	The BSWMD template SHALL provide means to describe - or reference to - the information needed for ECU Configuration activity and integration of a BSW module or cluster of BSW modules. This description formats SHALL be used for the ECU Configuration activity and integration along with the relevant BSW SWS documents.
Rationale:	By selecting the BSWMD of a BSW module's implementation, the necessary information for the ECU Configuration activity and integration of that module SHALL be available. When delivering several BSW modules in a cluster the BSWMD template SHALL support the integration of this cluster. However this description format might not formalize all aspects needed for taking integration decisions (e.g. scheduling).
Use Case:	To be able to exchange BSW modules from different vendors only the specified information can be used during the integration.
Dependencies:	BSWMD0014
Conflicts:	--
Supporting Material:	--

4.4.2 [BSWMD0008] BSW Module Description SHALL be tool processable

Initiator:	WP Methodology and Templates
Date:	30.05.2007
Short Description:	BSW Module Description SHALL be tool processable
Type:	New
Importance:	High
Description:	Work products based on the BSWMD template SHALL be readable and processable by tools.
Rationale:	The ECU Configuration activity of an ECU SHALL be supported by tools with the BSWMD as one input.
Use Case:	ECU Configuration activity will have to have tool support. The ICS SHALL be extractable from the BSWMD.
Dependencies:	--
Conflicts:	--
Supporting Material:	BSW159 [2]

4.4.3 [BSWMD0028] Development according to the AUTOSAR Generic Structure Template document

Initiator:	WP Methodology and Templates
Date:	29.07.2005
Short Description:	Development according to the AUTOSAR Generic Structure Template

	document
Type:	New
Importance:	High
Description:	The UML representation of the BSWMD template SHALL be developed according to the AUTOSAR Generic Structure Template.
Rationale:	The experience and tools already available for the AUTOSAR Metamodeling SHALL be reused.
Use Case:	The template for the BSWMD is similar to other templates already done with the AUTOSAR Generic Structure Template.
Dependencies:	--
Conflicts:	--
Supporting Material:	AUTOSAR Generic Structure Template [4]

4.4.4 [BSWMD0029] Transformation of BSWMD modeling according to the AUTOSAR Model Persistence Rules for XML

Initiator:	WP Methodology and Templates
Date:	26.07.2005
Short Description:	Transformation of BSWMD template modeling according to the AUTOSAR Model Persistence Rules for XML
Type:	New
Importance:	High
Description:	The XML representation for the BSWMD template SHALL be derived from its UML representation according to the AUTOSAR Model Persistence Rules for XML.
Rationale:	The experience and tools already available for the AUTOSAR Modeling SHALL be reused.
Use Case:	The template for the BSWMD is similar to other templates already done with the AUTOSAR Metamodeling Guide.
Dependencies:	--
Conflicts:	--
Supporting Material:	Model Persistence Rules for XML [5]

5 References

[1] Glossary,
AUTOSAR_TR_Glossary.pdf

[2] General Requirements on Basic Software Modules,
AUTOSAR_SRS_BSWGeneral.pdf

[3] Requirements on RTE Software,
AUTOSAR_SRS_RTE.pdf

[4] Generic Structure Template,
AUTOSAR_TPS_GenericStructureTemplate.pdf

[5] Model Persistence Rules for XML,
AUTOSAR_TR_XMLPersistenceRules.pdf

[6] Specification of Memory Mapping,
AUTOSAR_SWS_MemoryMapping.pdf

[7] Methodology,
AUTOSAR_MOD_Methodology.pdf

[8] Technical Overview,
AUTOSAR_TechnicalOverview.pdf

[9] Requirements on ECU Configuration,
AUTOSAR_RS_ECU_Configuration.pdf

[10] Specification of ECU Configuration,
AUTOSAR_TPS_ECUConfiguration.pdf

[11] Layered Software Architecture,
AUTOSAR_EXP_LayeredSoftwareArchitectur.pdf

[12] Specification of RTE,
AUTOSAR_SWS_RTE.pdf

[13] Basic Software Module Description Template,
AUTOSAR_TPS_BSWModuleDescriptionTemplate.pdf

[14] Meta Model-generated XML Schema,
AUTOSAR_MMOD_XMLSchema.xml

[15] Feature Specification of the BSW Architecture and the RTE,
AUTOSAR_TR_BSWAndRTEFeatures.pdf

6 Change History

6.1 Change History for AUTOSAR R4.0.1 against R3.1.5

6.1.1 Changed SRS Items

BSWMD0010	Compiler version and settings
BSWMD0014	Support of BSW Module clusters
BSWMD0025	Support for shipment information
BSWMD0027	Provide Vendor-Specific Module Definition
BSWMD0028	Development according to the AUTOSAR Generic Structure Template document
BSWMD0029	Transformation of BSWMD modeling according to the AUTOSAR Model Persistence Rules for XML
BSWMD0032	Recommended ECU Configuration Values
BSWMD0033	Pre-configured ECU Configuration Values
BSWMD0034	ECU Configuration Editor and Generation supported tool version information
BSWMD0035	Provide Standardized Module Definition
BSWMD0040	Identification of required API and functions
BSWMD0041	Declaration of the provided API argument data types
BSWMD0042	Description of the required API argument data types
BSWMD0043	Support description of common published information
BSWMD0047	Modeling of call-chain dependencies between BSW Modules
BSWMD0048	Tagging of Vendor-Specific Module Definition
BSWMD0050	Allow vendor-specific modification of Standardized Module Definition

6.1.2 Added SRS Items

BSWMD0015	Timing requirements
BSWMD0044	Description of generated artifacts
BSWMD0049	Describe optional and required elements
BSWMD0051	Description of libraries
BSWMD0052	Description of the generated RTE
BSWMD0053	Cyclic time based scheduling of BSW Main Functions
BSWMD0054	Mode Switches for BSW modules shall be supported
BSWMD0055	Simultaneous Mode transitions
BSWMD0056	API for Mode switch notification of BSW modules
BSWMD0057	Triggering of BSW Main Functions by Triggered Events
BSWMD0058	Synchronized Triggering by Triggered Events
BSWMD0059	API for Triggering BSW modules by Triggered Events
BSWMD0060	Support exclusive areas in BSW Modules and Application Software Components
BSWMD0061	Support for Debugging of variables
BSWMD0062	Provide Measurement and Calibration Support