Document Title	Requirements on Feature Definition of Authoring Tools
Document Owner	AUTOSAR
Document Responsibility	AUTOSAR
Document Identification No	100
Document Classification	Auxiliary

Document Version	1.0.5
Document Status	Final
Part of Release	3.2
Revision	1

Document Change History

Date	Version	Changed by	Change Description
23.03.2011	1.0.5	AUTOSAR Administration	Legal disclaimer revised
23.06.2008	1.0.4	AUTOSAR Administration	Legal disclaimer revised
31.10.2007	1.0.3	AUTOSAR Administration	• Document meta information extended • Small layout adaptations made
24.01.2007	1.0.2	AUTOSAR Administration	• “Advice for users” revised • “Revision Information” added
28.11.2006	1.0.1	AUTOSAR Administration	Legal disclaimer revised
10.05.2006	1.0.0	AUTOSAR Administration	Initial release

Disclaimer

Advice for users

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

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

Table of Contents

1 About this document

1.1 Introduction

This document defines requirements on the deliverable “Feature definition for AUTOSAR Authoring Tools V1.0”. In addition, some of the requirements have an impact on the meta-modeling profile used within AUTOSAR.

1.2 Terminology

- **A XML schema** means XML validation mechanisms like W3C DTD (“Extensible Markup Language (XML) 1.1, *http://www.w3.org/TR/xml11*”) or W3C XML schema (XML Schema 1.1, *http://www.w3.org/XML/Schema*”)
- **AUTOSAR model** is a generic expression for any kind of representation of instances of the AUTOSAR meta-model. It might be a set of files in a file system, an XML stream, a database or memory used by some running software.
- **AUTOSAR XML description** describes the XML representation of an AUTOSAR model. The AUTOSAR XML description can consist of several fragments. Each fragment must validate successfully against the AUTOSAR XML schema as defined in the AUTOSAR model persistency rules.
- **Authoring tools** are software tools working on any form of AUTOSAR data. The scope of an authoring tool (in terms of AUTOSAR data) is all data that it requires to properly function, especially the data that it can modify.
- **Data inconsistencies** are combinations of elements or attributes of elements that can be described by the modeling language, but models that they are part of are not "valid" in the sense that systems they model would be either impossible to create or they would not work as intended.

1.3 About Requirements

Each requirement has its unique identifier starting with the prefix "ATFD" (meaning **A**uthoring **T**ools **F**eature **D**efinition.) For any review annotations, remarks or questions please refer to this unique ID rather than chapter or page numbers!

1.3.1 Structure

Each requirement is defined as a table. The structure of the tables is as follows:

Initiator:	< number of originating work package, company, etc >
Date:	< date of last change >
Requirement:	< the normative text of the requirement >
Description:	< detailed description of the requirement >
Rationale:	< why is this necessary, what its omission could cause >
Use Case:	< example to a scenario that makes the requirement necessary or useful >
Dependencies:	< reference to depending and depended-on requirements >
Conflicts:	< reference to conflicting requirement >
Supporting Material:	< links to other documents >
Comment:	< additional remarks >

1.3.2 Conventions used

In requirements, the following specific semantics are used (taken from Request for Comment RFC 2119 from 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 described in RFC 2119. Note that the requirement level of the document in which they are used modifies the force of these words.

- MUST: This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- MUST NOT: This phrase, or the phrase "SHALL NOT", means that the definition is an absolute prohibition of the specification.
- 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.)

1.3.3 Guidelines

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

All Requirements shall have the following properties:

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

2 Requirements

This chapter provides a definition of the relevant requirements.

2.1 [ATFD001] Define subset

Initiator:	WP1.2
Date:	02.04.2005
Requirement:	The corresponding deliverable MUST define a subset of AUTOSAR features to serve as a basis for the first implementation of authoring tools.
Description:	The corresponding deliverable should identify crucial parts of the AUTOSAR concept that are of benefit but impose only a limited risk. These will serve as basis for implementation of authoring tools.
Rationale:	The implementation of all AUTOSAR features in a single step requires a so much effort, that it is likely that most concerned parties would define a subset to implement first anyway. By defining a subset centrally, the first implementation of tools can be evaluated in tool-chains as well as in isolation.
Use Case:	The subset can be used to create the first scaled-down versions of AUTOSAR tools and data exchange formats.
Dependencies:	--
Conflicts:	--
Supporting Material:	--
Comment:	Other subsets may be created for other AUTOSAR tooling, eg. BSW. Any Alignment with these is not in scope of Authoring Tools V1.0

2.2 [ATFD002] Identify features

Initiator:	WP1.2
Date:	02.04.2005
Requirement:	The subset MUST contain enough key features to verify and test the AUTOSAR methodology to the extent defined by the use cases described in the corresponding specification document
Description:	
Rationale:	The AUTOSAR meta-model provides a large number of meta-classes. The process of deciding what functionality is implemented in a specific tool is usually based on the identification of features. Therefore, the mapping of meta-classes to features facilitates the discussion about implementation of AUTOSAR Authoring Tools.
Use Case:	--
Dependencies:	--
Conflicts:	--
Supporting Material:	--
Comment:	The features should be self-contained.

2.3 [ATFD003] Include key features

Initiator:	WP1.2
Date:	24.01.2005
Requirement:	The subset MUST contain enough key features to verify and test the AUTOSAR methodology.
Description:	The subset is a trade-off between effort and risk. The more features that are part of subset, the higher the chance that any intrinsic problems with the respective AUTOSAR templates are identified, but the more work it will take to implement. This requirement states that the subset should contain at least some key elements, otherwise it is pointless to have a subset.
Rationale:	Without this requirement the subset could just be empty. This would conflict with the purpose of having a way of finding as many possible incompatibilities between tools and/or specifications as possible.
Use Case:	This requirement must be considered before deciding that a certain feature is not part of the subset.
Dependencies:	ATFD001
Conflicts:	ATFD004
Supporting Material:	--
Comment:	Note that it is impossible to exactly define when the subset contains enough features; it is up to the WP to decide.

2.4 [ATFD004] Limit risk

Initiator:	WP1.2
Date:	24.01.2005
Requirement:	The subset SHOULD only contain features that are absolutely necessary for the evaluation of various parts of the AUTOSAR methodology.
Description:	The subset is a trade-off between effort and risk. The more features are the part of subset, the higher chance that any possible problems are identified, but the more work it is both to identify the problem and to correct it. This requirement states that the subset should not contain anything unnecessary, as these feature increase the cost in terms of time and effort.
Rationale:	Without this requirement the subset could just contain the whole meta-model. This would conflict with the purpose of evaluating the AUTOSAR methodology with as little effort as possible.
Use Case:	This requirement must be considered when the criteria for subset membership are defined.
Dependencies:	ATFD001
Conflicts:	ATFD003
Supporting Material:	--
Comment:	Note that it is impossible to exactly define which features are absolutely necessary. Just like ATFD003, the fulfillment of this requirement is also up to the judgment of the WP.

2.5 [ATFD005] Declare criteria

Initiator:	WP1.2
Date:	04.02.2005
Requirement:	The subset document SHOULD declare the criteria for membership of features in the subset.
Description:	Some WP decisions about subset membership may have very specific reasons, but most decisions should be based on a few underlying principles. These should be documented.
Rationale:	Without such criteria, the subset may become incoherent, incidental.
Use Case:	These criteria should help in understanding the goal of the subset as well as assist in membership decisions about any possible new features introduced to the meta-model after the subset document is released.
Dependencies:	ATFD001
Conflicts:	--
Supporting Material:	--
Comment:	--

2.6 [ATFD006] Assess priorities

Initiator:	WP1.2
Date:	24.01.2005
Requirement:	The corresponding deliverable SHOULD define the priority of each feature before deciding its membership in the subset.
Description:	--
Rationale:	--
Use Case:	--
Dependencies:	ATFD001
Conflicts:	--
Supporting Material:	--
Comment:	--

2.7 [ATFD007] Formal description of subset

Initiator:	WP1.2
Date:	04.02.2005
Requirement:	The corresponding deliverable MUST define for every element in the meta-model, whether it is part of the subset.
Description:	Every attribute of every meta-class should be identified as either part of the subset or not. This follows from the membership of the features that the element or attribute is necessary for, but the relation must be made explicit, so the different implementations will support exactly the same data structure.
Rationale:	To make data exchange possible between tools.
Use Case:	Validating AUTOSAR descriptions against DTDs or XML Schemas generated from the meta-model is a part of the AUTOSAR concept. Therefore it must be possible to unambiguously create such DTDs.
Dependencies:	ATFD001
Conflicts:	--
Supporting Material:	--
Comment:	--

2.8 [ATFD008] Support structural validation

Initiator:	WP1.2
Date:	04.02.2005
Requirement:	The corresponding deliverable SHOULD define subset membership in a way that allows automatic generation of DTDs or Schemas limited to the subset.
Description:	--
Rationale:	--
Use Case:	--
Dependencies:	ATFD001
Conflicts:	--
Supporting Material:	--
Comment:	Requirement belongs to Meta-Model team