

Document Title	Template Modeling Patterns
Document Owner	AUTOSAR GbR
Document Responsibility	AUTOSAR GbR
Document Version	2.0.2
Document Status	Final
Part of Release	2.1
Revision	0014

Document Change History			
Date	Version	Changed by	Change Description
07.02.2007	2.0.2	AUTOSAR Administration	<ul style="list-style-type: none">• Legal disclaimer revised• “Advice for users” revised• “Revision Information” added
27.06.2006	2.0.1	AUTOSAR Administration	Layout Adaptations
19.05.2006	2.0.0	AUTOSAR Administration	Second release
18.10.2005	1.0.0	AUTOSAR Administration	Initial release

Page left intentionally blank

Disclaimer

Any use of these specifications requires membership within the AUTOSAR Development Partnership or an agreement with the AUTOSAR Development Partnership. The AUTOSAR Development Partnership will not be liable for any use of these specifications.

Following the completion of the development of the AUTOSAR specifications commercial exploitation licenses will be made available to end users by way of written License Agreement only.

No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

The word AUTOSAR and the AUTOSAR logo are registered trademarks.

Copyright © 2004-2006 AUTOSAR Development Partnership. All rights reserved.

Advice to users of AUTOSAR Specification Documents:

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	Introduction	5
1.1	Origins and Goals	5
1.2	Terminology	5
2	Common Patterns	7
2.1	Primitive Types	7
2.2	Identifiable Properties	7
2.3	Byte Order	9
2.4	AUTOSAR Properties	9
3	References	10
3.1	Normative References	10
3.2	Informative References	10

1 Introduction

1.1 Origins and Goals

AUTOSAR attempts to allow for a very flexible yet stable and reliable software engineering lifecycle through precise and formal description of all relevant aspects of a distributed system of embedded controllers and the corresponding executed software units.

The descriptions range from high level requirements on interfaces of software components to low level constraints on certain bits of a specific bus message. Various work packages in AUTOSAR determine the information that needs to be captured in the different descriptions.

For instance, in [4] it is defined how AUTOSAR software components need to be described, e.g. when a requirements specification is exchanged between OEM and supplier. The collection of attributes required specifying various AUTOSAR relevant artifacts like software components, ECUs and so on is called an *AUTOSAR template*. Once information is available a template is said to be filled out, leading to an *AUTOSAR description*.

The AUTOSAR template modeling guide [3] provides the formal means to define an AUTOSAR template. The various templates in AUTOSAR typically require a certain set of common features (e.g. a common byte order definition). Those features are collected in this document.

Therefore the common patterns for template model can be understood as an AUTOSAR standard library of modeling elements and patterns.

1.2 Change Process

Due to the nature of the document it is expected that not for each and every addition to the set of known patterns a full milestone cycle needs to be passed. Instead, new features may be requested for inclusion through AUTOSAR change requests.

1.3 Terminology

In this specification the key words **MUST**, **MUST NOT**, **REQUIRED**, **SHALL**, **SHALL NOT**, **SHOULD**, **SHOULD NOT**, **RECOMMENDED**, **MAY**, and **OPTIONAL**, when **EMPHASIZED** are to be interpreted as described in RFC 2119 [1].

The attributes of the classes introduced in this document are listed in form of class tables. They have the form shown in the example of the top-level element *AUTOSAR*:

Class	AUTOSAR			
Package	AUTOSAR Templates::GenericStructure::Infrastructure			
Class Description	Root element of an AUTOSAR description, also the root element in corresponding XML documents.			
Base Class(es)	--			
Attribute	Datatype	Mul.	Link Type	Attribute Description
topLevelPackage	ARPackage	0..*	aggregation	Top level packages of a description.

The headers in the table have the following meaning:

Class: The name of the class as defined in the UML model.

Package: The UML package the class is defined in. This is only listed to help locating the class in the overall metamodel.

Class Description: The comment the modeler gave for the class.

Base Classes: If applicable, the list of direct base classes.

Attribute: The name of an attribute of the class. Note that AUTOSAR does not distinguish between class attributes and owned association ends.

Datatype: The datatype of an attribute of the class.

Multiplicity: The assigned multiplicity of the attribute, i.e. how many instances of the given datatype are associated with the attribute.

Link Type: Specifies, whether the attributes is part of the class (aggregation) or just referenced by it (reference). Instance references are also indicated in this field (refer to [3] for details).

Attribute Description: The comment the modeler gave for the class attribute.

2 Common Patterns

The nature of the common patterns given below is similar to the standard library of a compiler: a set of predefined structures and elements to be used in an AUTOSAR template model.

2.1 Primitive Types

In addition to the fundamental primitive types given in [3], more content related types are offered for usage in template models. They are:

DateTime: This datatype represents a timestamp. While the actual representation of its values depends on the platform used (e.g. XML, database systems, ...) the definitions of timestamps defined in ISO 8601 [6] SHOULD be followed.

2.2 Identifiable Properties

The base class *Identifiable* defined in [3] has further content related attributes, which are shown in Figure 1:

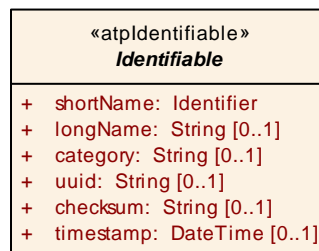


Figure 1: Base class *Identifiable* and its content relevant attributes.

The attributes are given in the following class table:

Class	Identifiable (abstract)			
Package	AUTOSAR Templates::GenericStructure::Infrastructure			
Class Description	Instances of this class can be referred to by their identifier (while adhering to namespace borders).			
Base Class(es)	--			
Attribute	Datatype	Mul.	Link Type	Attribute Description
adminData	AdminData	0..1	aggregation	Administrative data "AdminData" providing information about intended document fragmentation, versioning, used languages, product data management.
category	String	0..1	aggregation	A category allows specifying the particular nature of the object in question. In most cases, this categorization ends up in a subset of information which is appropriate for an object of the category in question.
checksum	String	0..1	aggregation	checksum calculated from the * attributes * aggregations and aggregated non-identifiables * references
desc	String	1	aggregation	desc represents a general but brief description of the object in question.
longName	String	0..1	aggregation	A human readable name of the meta class instance.
shortName	Identifier	1	aggregation	A machine readable name of the meta class instance. As defined for the M2 datatype Identifier, the shortName needs to be unique within the namespace it is declared in.
timestamp	DateTime	0..1	aggregation	the timestamp of the creation or modification of an instance, its attributes, references or aggregated non-identifiables.
uuid	String	1	aggregation	The purpose of this attribute is to provide a globally unique identifier for an instance of a metaclass. The values of this attribute should be globally unique strings prefixed by the type of identifier. For example, to include a DCE UUID as defined by The Open Group, the UUID would be preceded by "DCE:". The values of this attribute may be used to support merging of different AUTOSAR models. The form of the UUID (Universally Unique Identifier) is taken from a standard defined by the Open Group (was Open Software Foundation). This standard is widely used, including by Microsoft for COM (GUIDs) and by many companies for DCE, which is based on CORBA. The method for generating these 128-bit IDs is published in the standard and the effectiveness and uniqueness of the IDs is not in practice disputed. If the id namespace is omitted, DCE is assumed. An example is "DCE:2fac1234-31f8-11b4-a222-08002b34c003".

2.3 Byte Order

When more than one byte is stored in the memory the order of those bytes may differ depending on the architecture of the processing unit. If the least significant byte is stored at the lowest address this architecture is called little endian and otherwise it is called big endian.

ByteOrder is very important in case of communication between different PUs or ECUs.

If specifying a byte-order, template classes MUST use the following enumeration:



Figure 2: Enumeration to indicate byte order of a processing unit.

2.4 AUTOSAR Properties

The concept of properties is currently not an official common pattern. It is expected that future versions of this document will integrate a solution to address the requirement of model variants.

3 References

In this section the references used in this specification are listed. They are separated into normative and non-normative (informative) references.

3.1 Normative References

- [1] Key words for use in RFCs to Indicate Requirement Levels. Network Working Group, S. Brandner, Harvard University, 1997.
<http://www.ietf.org/rfc/rfc2119.txt>
- [2] Guide to the SI, with a focus on usage and unit conversions: NIST Special Publication 811, 1995 Edition, by Barry N. Taylor. Guide for the Use of the International System of Units (SI).
- [3] Template UML Profile and Modeling Guide
https://svn.autosar.org/repos/10Releases/AUTOSAR_TemplateModelingGuide.pdf.

3.2 Informative References

- [4] Software Component Template,
https://svn.autosar.org/repos/10Releases/AUTOSAR_SoftwareComponentTemplate.pdf
- [5] MSR-SW Element Attribute Documentation V2.2.2.
<http://www.msr-wg.de/medoc/download/>
- [6] ISO 8601:2004, Representation of dates and times, published ISO standard,
<http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=40874>