

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 1 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

Specification

Requirements Interchange Format (RIF)

Version

1.0

Date

March 16th 2005

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 2 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

Document History

Revision	Modification	Author	Date
0.1	1 st initial Draft	Rupert Wiebel	February 5, 2005
0.5	2 nd Draft after comments Matthias Recknagel	Rupert Wiebel	February 12, 2005
0.8	3 rd Draft rework and consolidation of all information	Steffen Höh	March 4, 2005
0.9	4 st Rework HIS structure integration	Steffen Höh	March 10, 2005
1.0	5 st Rework and integration of comments 1 st -2 nd Chapter Matthias Recknagel	Steffen Höh	March 16, 2005

All trademarks mentioned within this document are unrestricted subject to the regulation of each identified and valid law of identification and as the case may be the law of seizure of each registered owner.

We state explicitly that all mentioned notations, trademarks and logos are in the ownership of each company and/or people and are subject to the copyright.

Table of Contents

Document History	2
1 Overview.....	4
1.1 Introduction	4
1.2 Who should read this document	4
1.3 Structure of this Document / How to read this document.....	4
1.4 Why do we need an Interchange Format for Requirements	4
1.5 RIF Partners.....	7
2 Use Cases Supported by RIF	7
3 RIF – Format and Interfaces	7
3.1 The Interchange Format.....	7
3.1.1 RIF Modeling	7
3.1.1.1 Introduction: Why modeling the RIF?.....	8
3.1.1.2 Information Elements and Information Types.....	8
3.1.1.3 Description of the RIF-model	10
3.2 Tool Interfaces	28
3.2.1 Typical Scenarios for importing/exporting.....	28
3.2.1.1 Guide to implement a user interface.	28
3.2.1.2 Sample export process:.....	29
3.2.1.3 Sample import process:	29
3.2.1.4 Warnings and error handling.....	30
Appendix A: Glossary	30
3.3 Glossar: RIF-Project	30
3.3.1 list of abbreviations	30
3.3.2 Object-Words	32
3.3.3 Roles/ Actors	43
3.3.4 Tools	44
3.3.5 Trade marks	44
Appendix B: RIF Schema	45
3.4 RIF XML Schema (rif.xsd)	45
3.5 RIF integrated XHTML Schema	55
Appendix C: Mapping Table.....	71
Appendix D: RIF Test data	76
3.6 RIF XML File (RIF_EF_example.xml).....	76

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 4 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

1 Overview

1.1 Introduction

Requirements Management has been an integral part of the development process in various industries (especially in the Mil/Aero or the Medical Device industry) for years. Other industries have been adopting these techniques recently.

In the automotive industry, where the initiative for this interchange format started, the term “Requirements Management” (RM) found its entry around 1999. Up to then car manufacturers and the supplier industry already took care of their requirements, but did not make use of the advantages of modern Requirements Management. As Requirements Management spread in the automotive industry, more and more car manufacturers (OEM’s) and suppliers have been applying Requirements Management and making use of Requirements Management tools. Large improvements have been made in these organizations. Now with a modern RM in place on both sides (OEM’s and supplier), it became obvious that RM should not stop at company borders but should bridge this gap.

For technical and organizational reasons OEM and supplier will not be able to work on the same RM database. Because of that, a smart exchange mechanism needed to be found to exchange RM data between OEM and supplier without losing the advantage of modern Requirements Management at the organization borders. RIF was created to support such an exchange of information to bridge the gap.

1.2 Who should read this document

This document is created to inform

- persons interested in exchanging RM data between organizations that do not have a possibility to share the same RM database
- RM-tool vendors who want support RIF with export and import interfaces for their RM tools

1.3 Structure of this Document / How to read this document

Chapter 1 of this document gives an overview, why the RIF is needed, who worked on it and how it was created.

Chapter 2 describes the Use Cases that shall be supported by RIF. These Use Cases led to requirements against RIF.

Chapter 3 gives details about RIF itself. This chapter is thought for all people who have to implement interfaces for writing and reading RIF files.

1.4 Why do we need an Interchange Format for Requirements

Gathering and documenting the requirements for the development of a system is an integral part in each system development process. The traditional approach to this task is quite simple: just write down the requirements into a document and call it “specification”. Experience shows that doing the things this way lead to trouble in the later phases of the project. Incomplete requirements often cause changes in the deliverables. These changes are hard to do and expensive because they often do not fit to the initial concepts. Again, these changes are poorly documented ...

Requirements engineering (RE) and requirements management (RM) offer methods and tools to overcome this problem. In the beginning, they were used to develop large and safety critical systems like aircrafts or power plants. The increasing complexity of other products like software systems or cars makes these methods more and more attractive for the development process of these products. Tools that were developed to support the RE/RM process focused on the management of all requirements for a project in special databases. All partners who worked on the requirements were thought to have direct access to this database. For documentation purposes, it was sufficient to export the requirements into a document that can be easily archived.

Increased complexity of the products and specialization of the partners requires closer collaboration of the partners and leads to more and more specialized knowledge on the development tasks for all partners. If one partner also collaborates with competitors of the other partner, a common database is no longer sufficient for the

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 5 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

interests of the partners. An obvious solution for this problem is, that all partners run their own RE processes and manage the different levels of requirements in their own databases. Doing the things this way makes it necessary to interchange the requirements between the partners. Considering the features of current RM tools, this is mostly done by exchange of documents.

Traditional interchange process for requirements in automotive industry starts with the user requirements that are documented by the OEM and delivered to the supplier. In most cases, the exchange of the requirements across company borders is only possible using one of the following mechanisms:

- print off the relevant information from the RM tool and send it to their partners (worst scenario)
- export the relevant information from the RM tool into a text processor readable format (most likely Microsoft® Word)
- if both sides use the same RM tool: hope that the RM tool vendor supports an offline exchange of RM data and use this one.

None of the above is satisfying.

The supplier reads the user requirements, evaluates them and produces the system requirements as an answer to the user requirements of the OEM. If user and system requirements are exchanged as documents, working with these documents is often very time consuming, as the document structures may be completely independent. Comparing the system requirements with the user requirements is done by manually marking corresponding information. Due to the fact that the result of this work is mostly written into the documents, it is very hard to reuse it and identify the differences of new versions of the documents that might appear during this process. It is evident to say that doing things this way not satisfying for all partners and not adequate to the opportunities modern tools might offer. Objectives of RIF

The fact that more and more OEMs and suppliers use RM tools to manage their requirements leads to a situation where a smart handling of requirements across company and tool borders should be possible. The most important thing that is missing is an exchange format that is independent from a special RM tool and supported by all vendors of RM tools. The Requirements Interchange Format (RIF) defines such a tool independent exchange format. With the help of RM tools supporting RIF it will be possible to bridge the gap between companies that cannot share a common RM database:

- The collaboration between the partners will be improved in a way that they can benefit from RM methods across company borders in the coordination of user requirements and system requirements. Both partners do not have to use the same RM tool.

Figure 1: Exchange of requirements between two RM databases using RIF

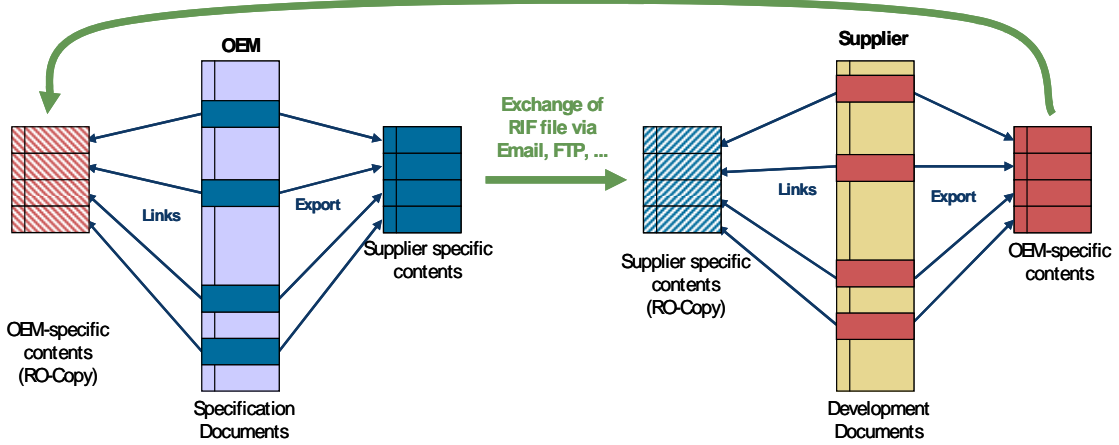
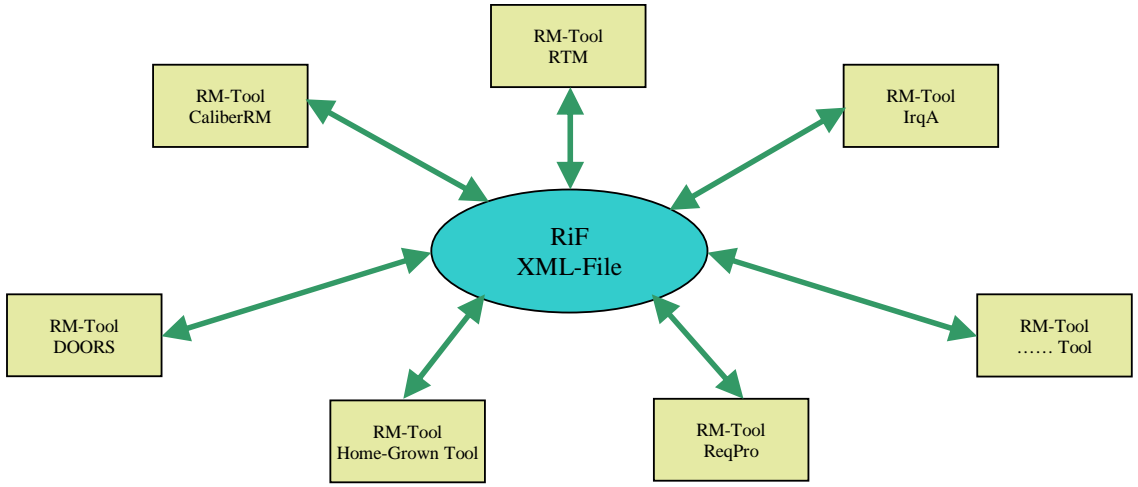


Figure 1 shows the exchange of requirements between two RM tools using the RIF for the transport of information. The situation shown in the figure assumes that the OEM identifies and marks the requirements relevant for the supplier in his RM database. These requirements are copied in a separate repository within the OEM database keeping the identifiers of the requirements with the requirements. This supplier specific repository is then exported to a RIF file that can be sent to the supplier using traditional file transfer tools (email, ftp...). The supplier imports the RIF file into his database, starts to identify the corresponding system requirements in his database and adds new system requirements where necessary. The OEM specific system requirements also copied to a separate repository within the suppliers database that can be exported as an RIF file. The RIF file with the system requirements is sent back to the OEM and imported into his requirements database. If a copy of the initial identifiers of the user requirements in the supplier's database is included in the imported repository, the links between corresponding user and system requirements can be generated automatically.

The complete process can be done several times using an update mode for the repository that keep the imported information. Dependant on the opportunities of the RM tools, changes in the imported information can be made visible easily.

Figure 2: Use of RIF for the interchange of requirements between different tools



HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 7 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

Figure 2 shows the usage of RIF for the interchange of information between different tools. One of the basic ideas of RIF is to offer the opportunity to store all information that can be handled by each of the tools. This way of proceeding makes it necessary to have an additional mapping table that defines how the different tools store their information in the RIF data objects. The disadvantage of having a format definition *and* a mapping table is compensated by the advantage of keeping the format itself stable in case of changes of the interfaces of the tools that support RIF. In principal, the definition of an interchange format cannot solve the problem that different tools handle different types of information. The approach of HIS makes it possible that the exchange of information between different installations of the same tool can be done with a standardized format and that the same format can be used for exchange of information between different tools. In this case the mapping table with becomes important. The mapping describes how information is handled between the different tools. This information can be important to assess the impact of the restrictions of data interchange on the RM information model for the projects.

1.5 RIF Partners

The initial work on RIF was done by the members of the HIS group and additional partners that were associated for this project. The HIS group is the association of the vehicle manufacturers Audi, BMW, DaimlerChrysler, Porsche, and Volkswagen to bundle their activities for standard software modules, process maturity levels, software test, software tools and programming of control units. The common goal is to achieve and use joint standards. The group that is working on the initial release of RIF consists of the HIS members and Adam Opel AG, Conti Temic microelectronic GmbH, Carmeq GmbH and the HOOD Group.

2 Use Cases Supported by RIF

RIF is designed to support two main use cases:

UC1 external creation of a complete specification

UC2 coordination of user and system requirements

The first use case (UC1) describes the situation where a specification is completely written by an external partner. In this case, all information can be modified after the module was exchanged.

UC2 focuses the coordination of the specified content between two partners. In this process, the user requirements usually are specified by the OEM (partner 1) and sent to the supplier (partner 2). The supplier assesses the requirements in order to either agree the requirements or give detailed feedback on individual topics. This results in a commented version of the specification that is sent back to the OEM who again assesses the feedback and documents his opinion in the module. This process can be done several times until a common opinion on the requirements is reached. In this process, each partner has its own information objects where requirements and the feedback are stored. It is important that each partner be allowed to modify only the information objects he is responsible for.

3 RIF – Format and Interfaces

3.1 The Interchange Format

3.1.1 RIF Modeling

This chapter describes

- information on the methodology used for modeling the RIF exchange file format that is necessary to understand the model,
- the RIF-model itself
- the workflow of importing and exporting RIF files that are necessary to implement RIF interfaces for RM tools

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 8 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

The RIF-model is the basis for generating an XML-Schema that defines valid RIF exchange files. The conversion of the model into an XML-Schema and other XML related issues are addressed in Chapter 3.1.1.3

3.1.1.1 Introduction: Why modeling the RIF?

The data structures needed for storing and transferring requirements together with additional information (e.g. data types, links, access rights etc.) are complex due to many different relationships among the various data within the exchange file. A format that wants to enable its users to transport such complex data and relations needs adequate structures to depict this information. A means of handling this complexity is the adoption of graphically modeling the information elements and their relationships. The graphical RIF model is based on UML class diagrams and thus straight-forward to understand. Using appropriate tools, it is possible to automatically convert the graphical RIF model into an XML-Schema that can subsequently be used by RE/RM tool vendors to validate the correctness of an RIF exchange file.

3.1.1.2 Information Elements and Information Types

An exchange file (EF) that is in accordance with the RIF Exchange File Format (EFF) contains a large amount of different data and information. From the point of view of an RE/RM tool, the atomic unit of this data and information is the *information element* (e.g. a requirements text, an attribute value, a relation that links two requirements).

Information elements have certain common characteristics and can thus be categorized or typed. An *information type* is a category of information elements with the same properties in terms of e.g. element attributes or relationship to other information elements. An *information element* can also be seen as an instantiation of an *element type*. Please note that there can be multiple instantiations of *information elements* from the same *information type* that differ from each other in the concrete values of element attributes and/or in the concrete relationships to other information elements.

The RIF model is located on the same abstraction level as *information types*. It specifies and describes the different kind of *information types* as well as their relationships among each other.

As known from UML classes, an *information type* (IT) has the following characteristics:

- It has a name (the *IT name*).
- It is either *abstract* or *concrete*.
- It can have an arbitrary number of (*IT*) *attributes*.
- It can have additional *tags*.

The *IT name* is the identity of the information type and is unique within the RIF model.

The property of being *abstract* means that information elements of this information type never appear in the RIF exchange file. *Abstract* information types are used internally for better structuring of the model. Information elements that can appear in the RIF exchange file are instantiated from *concrete* (non-abstract) information types.

Please note that in the graphical RIF-model, *abstract* information types are indicated by writing their names with *italic* letters.

IT attributes (not to be confused with *requirement attributes*!) contain data of a specified type (e.g. integer, real, string) that is associated with the information type.

Tags specify special properties of an information type with regard to the automatic XML-Schema generation process. They are only relevant for certain XML related issues. Which are described in a following chapter.

Relationships between Information Types

In the RIF-model, relationships can occur between exactly two *information types*. There are no n-any relationships. A relationship between two *information types* means that two *information elements* that are instantiated from these two *information types* can have the specified relationship.

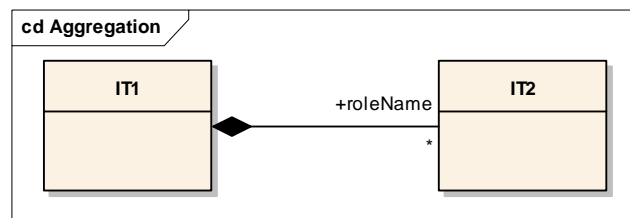
These relationships are one of the following:

- Aggregation
- Generalization
- Association

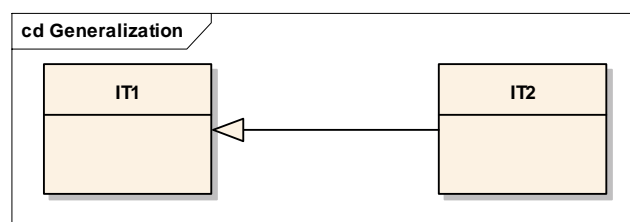
In the following, the explanation of the possible relationships is additionally given on the level of concrete *information elements* in order to simplify the understandability.

Please assume that IT1 and IT2 are two different *information types*, and IE1 and IE2 are *information elements* that have been instantiated from IT1 and IT2, respectively.

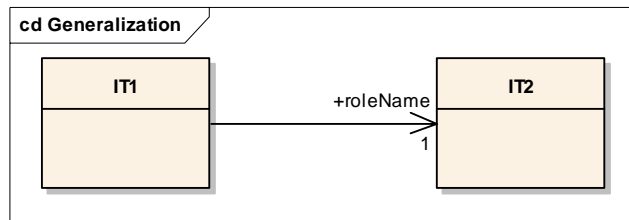
The expression “IT1 *aggregates* IT2” means that IT2 is contained in IT1. IE2 is thus part of IE1. The existence of IE2 is dependent on the existence of IE1.



The expression “IT1 *generalizes* IT2” is equivalent to “IT2 *inherits* from IT1”. This implicates that IT2 receives the attributes, aggregated elements and associations from IT1.



Associations are always unidirectional. An *association* from IE1 to IE2 allows the navigation from IE1 to IE2 while the location of IE1 and IE2 is arbitrary. That means IE1 contains an unambiguous reference to IE2. Note that due to being uni-directional, the reverse navigation from IE2 to IE1 is not possible in this case. On the *information type* level, *associations* are further specified with *cardinalities* on both origin and target side. The cardinality on the origin side IT1 indicates the possible number of associations. The cardinality on the target side IT2 specifies the possible number of instances of *information elements* from the associated *information type* IT2 that can be referenced.



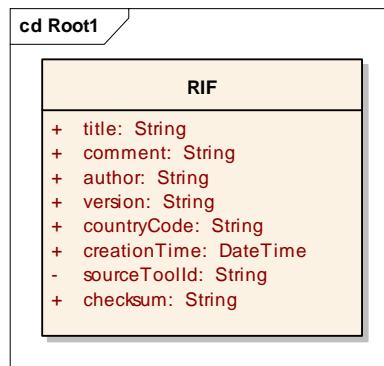
Associations with an *abstract* target information type are in fact associations to concrete information types that inherit from the *abstract* information type.

3.1.1.3 Description of the RIF-model

The concepts described in the previous section are the basis on which the RIF exchange file format has been modeled and specified. The following sub-sections describe the RIF model. It is intended that the level of details in this description together with the RIF-Schema is sufficient to enable RE/RM tool vendors to implement RIF importer and exporter functionality in their RE/RM tools.

General structure

The information type “RIF” is the root of the RIF-model. There must be exactly one instantiation of the root information type “RIF” per RIF exchange file.



An instantiation of the root type, i.e. a *root element*, contains element attributes that store descriptive information:

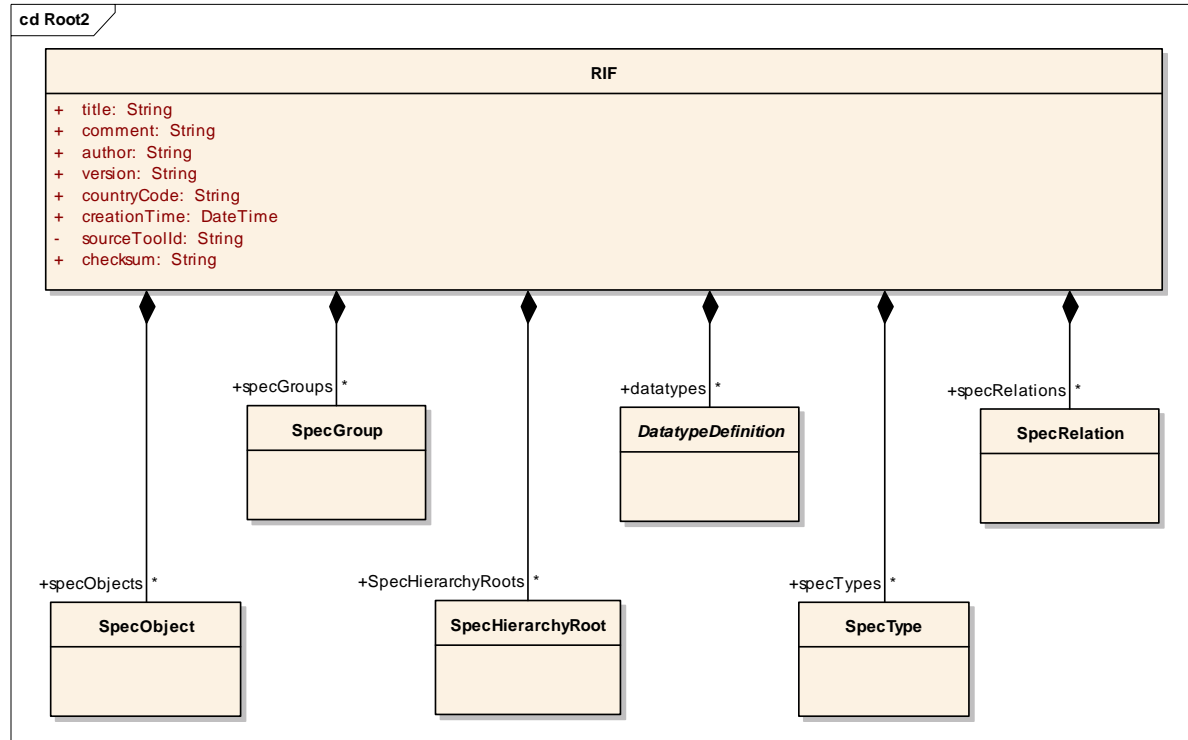
Attribute in <i>RIF</i>	Description
title	Title of the RIF exchange file
comment	Additional comment on the RIF exchange file
author	Author(s) of the RIF exchange file
version	Version of the RIF exchange file
countryCode	Code of the country to which the RIF exchange file is related.
creationTime	Time of creation of the RIF exchange file in the format of the XML Schema data type “dateTime” which specifies the time format as <i>CCYY-MM-DDThh:mm:ss</i> with optional time zone indicator as a suffix <i>±hh:mm</i> . Example: 2005-03-04T10:24:18+01:00 (MET time zone).
sourceToolId	Identifier of the RE/RM tool that has exported the RIF exchange file

checksum	Checksum on the RIF exchange file. See Appendix for details.
----------	--------------------------------------------------------------

Except for the checksum attribute, each of these element attributes contains an arbitrary string. The checksum attribute contains a checksum on the whole RIF exchange file that is generated as specified in Appendix.

Furthermore, the *root element* includes information elements of type

- TimeStamp,
- DatatypeDefinition,
- SpecType,
- SpecObject,
- SpecRelation,
- SpecGroup, and
- SpecHierarchyRoot.



While *TimeStamp* must occur exactly once within the *root element*, all other information elements of a type from the list above can occur with an arbitrary number (including zero) inside the *root element*.

The following table gives a short description on the *information types* that are aggregated by the root type "RIF":

Information type	Short description of information elements of the given type
TimeStamp	Contains the absolute time of generation of the RIF exchange file.
SpecObject	Constitutes an identifiable requirements object that can be associated with various attributes. This is the smallest granularity by which requirements are referenced.

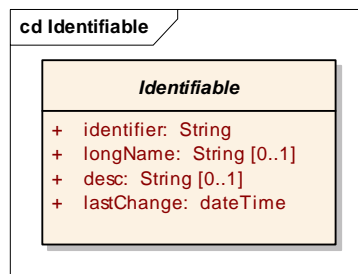
	The <i>SpecObject</i> instance itself does not carry the requirements text or any other user defined content. This data is stored in <i>AttributeValue</i> instances which are associated to the <i>SpecObject</i> instance.
SpecGroup	Defines a group (i.e. a set) of <i>SpecObject</i> instances.
SpecHierarchyRoot	Contains the root node of the tree that hierarchically structures <i>SpecObject</i> instances.
DatatypeDefinition	Contains definitions of user-defined data types such as integer with a certain range or an enumeration data type. Basic types have also to be defined as instances of <i>DatatypeDefinition</i> so that they can be referenced by instances of <i>AttributeDefinition</i> .
SpecType	Contains a set of attribute definitions. Any information element that inherits from <i>ElementWithUserDefinedAttributes</i> (i.e. instances of <i>SpecObject</i> , <i>SpecGroup</i> , <i>SpecHierarchyRoot</i> , and <i>SpecRelation</i>) has to be associated with exactly one <i>SpecType</i> instance.
SpecRelation	Contains relations (links) between two <i>SpecObject</i> instances.

Before describing the information types from the table above in more detail, the concept of *Identifiable* and *SpecElementWithUserDefinedAttributes* are explained first. These concepts are generic and applied to different information types.

Identifiable

Within the RIF exchange file, information elements are able to refer to other information elements using the association relationship. This reference mechanism is realized by generating unique identifiers for each information element that must be referable. These unique identifiers are created by the exporter functionality of the RE/RM tool and are “lifetime identifiers” for the specification elements, i.e. they must not be altered by other tools that import the specification elements and later re-export them. In this way, the tool from which the specification element originates is able to identify the re-imported specification elements that may have been changed.

In the RIF model, all information elements that must be referable in the way described above inherit from the abstract information type *Identifiable*:



Additionally to the identifier, the abstract super-class *Identifiable* contains other element attributes that contain specific information on an information element. The following table describes the element attributes:

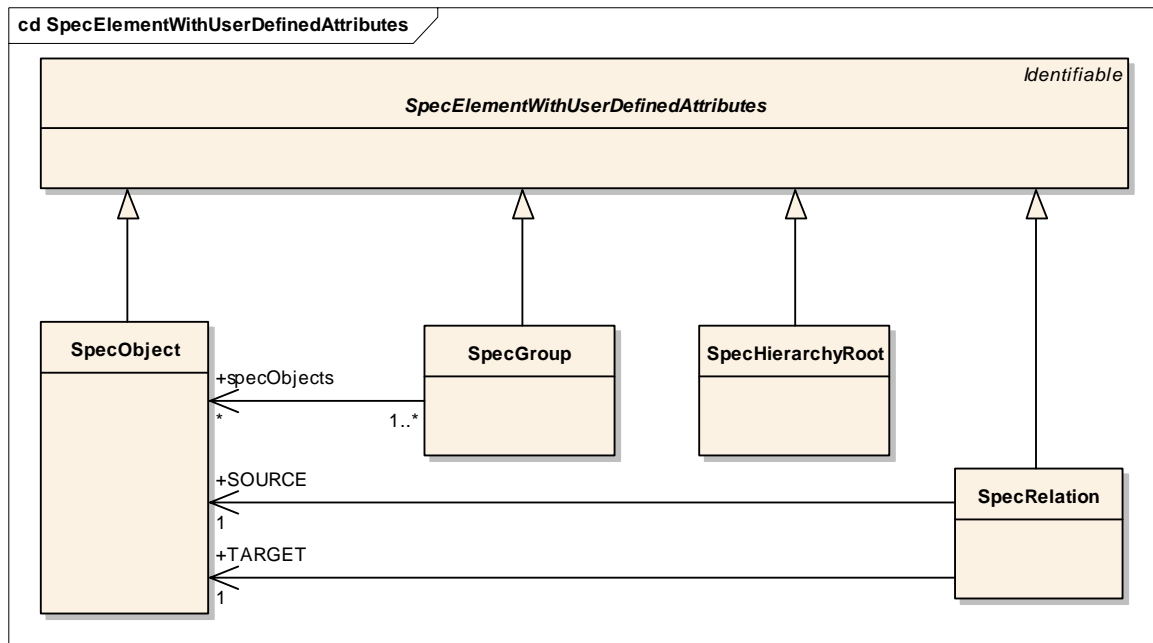
Attribute in <i>Identifiable</i>	Description
identifier	The unique identifier created by the RE/RM tool where the information element originates from during the export.

longName (<i>optional</i>)	The human readable name for the information element. As an example, for <i>AttributeDefinition</i> instances, <i>longName</i> contains the column header that usually specifies the meaning of the attribute definition.
desc (<i>optional</i>)	Additional description for the information element.
lastChange	The date and time of the last change of the information element. This includes the creation of the information element. <i>lastChange</i> is of the XML Schema data type “dateTime” which specifies the time format as <i>CCYY-MM-DDThh:mm:ss</i> with optional time zone indicator as a suffix <i>±hh:mm</i> . Example: 2005-03-04T10:24:18+01:00 (MET time zone).

Please note that RIF does not prescribe how the unique identifier is generated. The only requirement on the identifier is that it must be globally unique, i.e. not only unique within the exchange file but also between all RIF exchange files that might exist. This ensures the possibility of uniquely identifying each specification element in situations where many different RE/RM tools are employed to generate requirements in a distributed manner. These requirements can subsequently be collected into one RE/RM tool without running into problems of identification.

SpecElementWithUserDefinedAttributes

SpecElementWithUserDefinedAttributes is an abstract information type and is a generalization from the information types *SpecObject*, *SpecGroup*, *SpecHierarchyRoot* and *SpecRelation*.



The relationships between the four information types that inherit from *SpecElementWithUserDefinedAttributes* are described later and can be ignored at this point.

From a requirements management point of view, a *specification element* is an information element that carries content relevant for the specification of the system in scope. Thus, not only the requirement itself is a *specification element* but also relations (links) between requirements, the hierarchical structure of requirements and the information to which group a requirement belongs to make up a specification document and are therefore *specification elements*.

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 14 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

For requirements, it is well-known that they can be enriched with additional *requirement attributes* that are defined by the user. In RIF, this concept has been generalized and applied to relations, to the specification group and to the specification hierarchy as well.

Specification elements are usually not associated with arbitrary combinations of attribute elements. For example, specification objects that belong to the same specification group carry the same combination of attribute types. Therefore, it makes sense to define these combinations of attribute types so that they can be easily referred to within the RIF exchange file.

A combination of attribute types is defined by an instance of the information type *SpecType*. *SpecType* aggregates the abstract class *AttributeDefinition* which is a super-class for the different concrete types of *attribute definitions* (please refer to Section 0 on data types in the RIF exchange file format). Note that an *attribute definition* can also be seen as the instantiation of an attribute type for a specific specification element. The attribute definition entitles the attribute, associates it with a data type and makes it recognizable so the importing tool knows how to put the attribute values into the different “columns”. (In some RE/RM tool, an *attribute definition* for specification objects corresponds to a column carrying user-defined attribute values.)

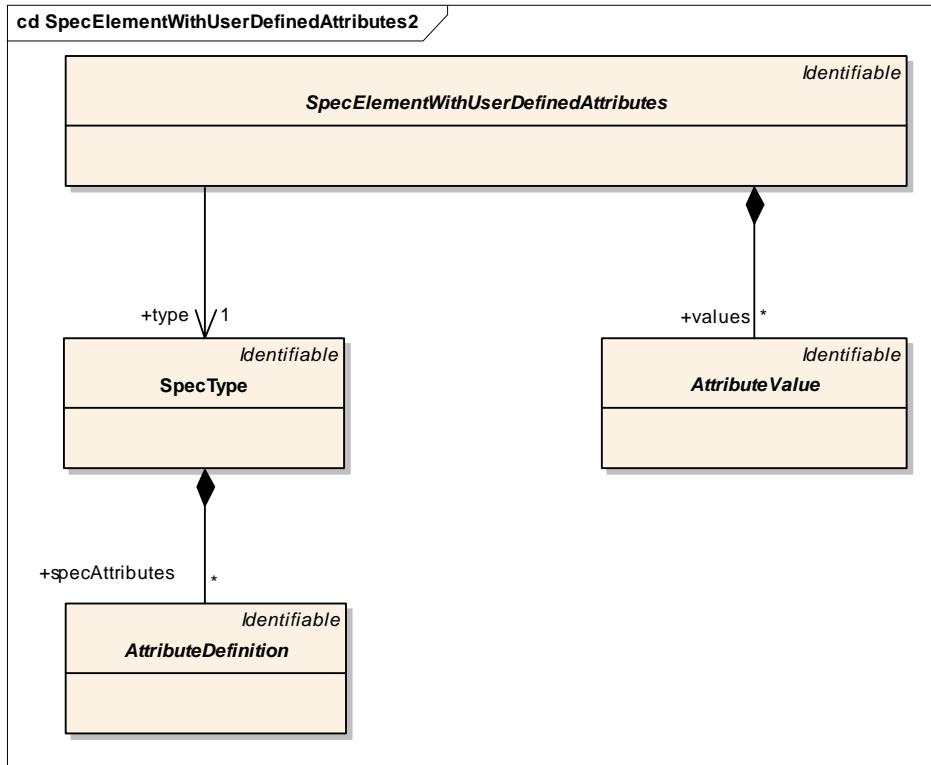
The combination of attribute definitions is specified by the information type *SpecType* that aggregates an arbitrary number of *AttributeDefinitions*.

For the information type *SpecElementWithUserDefinedAttributes*, it is now sufficient to refer to a *SpecType* (i.e. association with a single *SpecType*) in order to be associated with a combination of *attribute definitions*. (For formal reasons, this holds true also if the number of associative attribute definition instances is zero).

At last, *SpecElementWithUserDefinedAttributes* aggregates an arbitrary number of *AttributeValues*. Instances of *AttributeValues* carry concrete values of the type defined by the associated *AttributeDefinition*. This is further explained in Section 0 on data types.

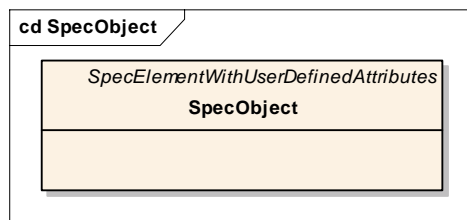
Please remember that *SpecElementWithUserDefinedAttributes* is abstract and is thus only used to be inherited from. The association to a *SpecType* as well as the instances of attribute values are therefore located inside the four specification elements that inherit from *SpecElementWithUserDefinedAttributes*.

Please note also that it is not possible to check on the XML level (e.g. by an XML parser) whether the *AttributeValue* instances inside a *SpecObject* instance correctly refer to *AttributeDefinition* instances that belong to the *SpecType* instance that is associated with the *SpecObject* instance. It is also not possible to check on the XML level whether the contents in *AttributeValue* instances really conform to the type definition associated with the attribute. These validity checks have to be performed on a level where this knowledge from the RIF model is available. In future versions of the RIF model, these consistency conditions will be specified using OCL (Object Constraint Language) to enable consistency checks on the XML level.



SpecObject

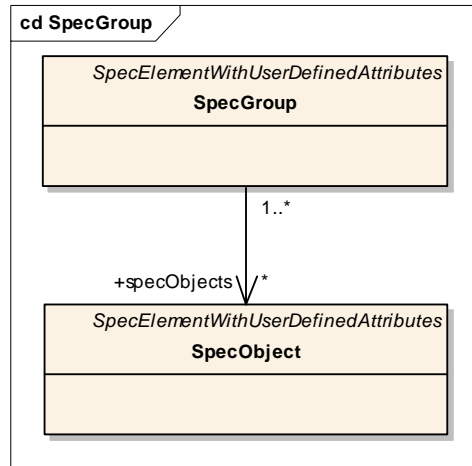
Instances of the information type *SpecObject* constitute individually identifiable requirements. An instance of *SpecObject* is “empty” by itself and therefore contains no data. Data that is associated with a *SpecObject* instance is stored in instances of *AttributeValue* which are aggregated by *SpecObject* (through inheritance from *SpecElementWithUserDefinedAttributes*).



In practical use, there are usually at least two instances of *AttributeDefinition* that are carried by all *SpecObject* instances: The requirement identifier from the source tool and the *requirement text* (i.e. the verbal description of the requirement).

SpecGroup

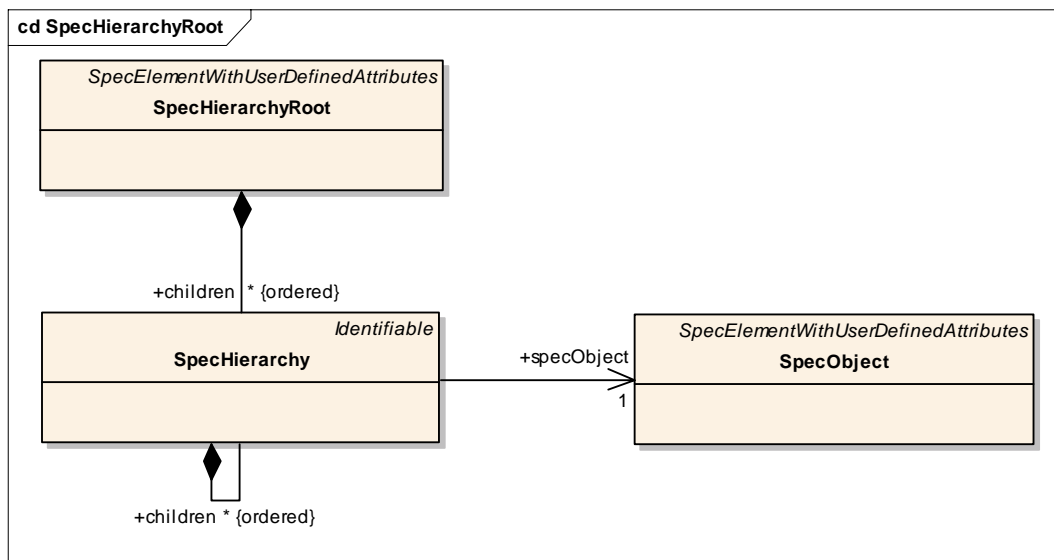
Multiple instances of *SpecObject* can be grouped together. Such a group is described by an instance of *SpecGroup* which aggregates instances of *SpecObject*.



SpecHierarchyRoot

The specification hierarchy describes the hierarchical structure of instances of *SpecObject*. This hierarchical structure can be seen as a tree with one instance of *SpecHierarchyRoot* as its root node and an arbitrary number of instances of *SpecHierarchy* as tree nodes. Links between the nodes of the tree are established by the “recursive” aggregation of instances of *SpecHierarchy* inside other instances of *SpecHierarchy*.

The mapping of this tree structure to actual requirements is realized by a reference to a *SpecObject* instance within each instance of *SpecHierarchy*.



DatatypeDefinition

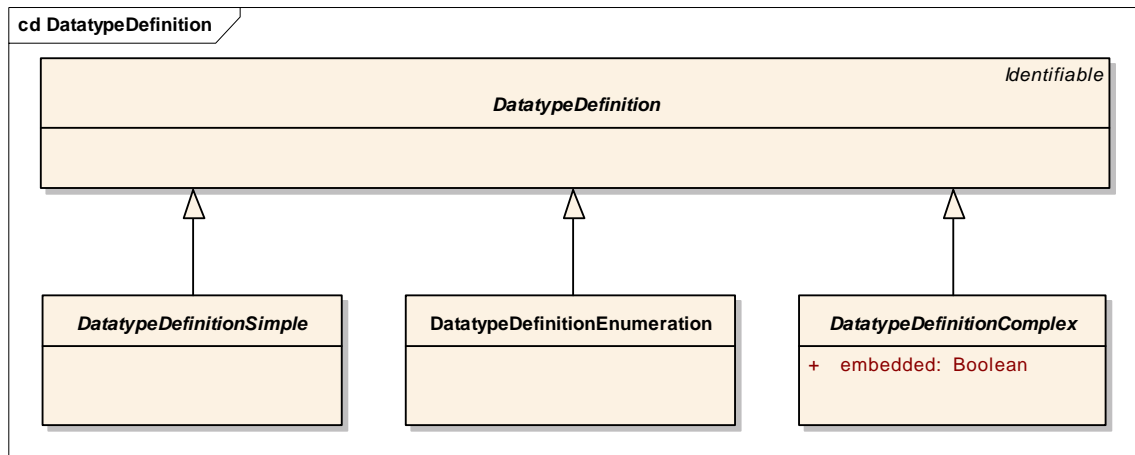
In RIF, there are three kinds of data types:

1. Simple data types (i.e. Integer, Real, Boolean, String)
2. Enumeration data types (both single and multiple enumerations)

3. Complex data types

Complex data types are in fact data files that can be either binary or have an XML based content. Although there are partially large differences in terms of data concepts between these kinds of data types, a common modeling approach has been realized as far as reasonable.

The abstract super-class for the three kinds of data types is *DatatypeDefinition*:

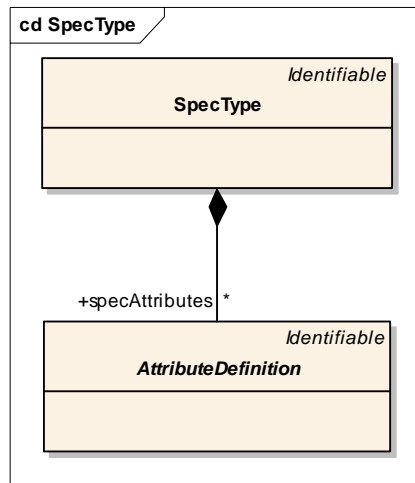


Concrete information types with data type definitions inherit from the abstract information types *DatatypeDefinitionSimple*, *DatatypeDefinitionEnumeration* or *DatatypeDefinitionComplex*. They are described in more detail in Section 0.

A concrete information type defining a data type is referenced by an instance of *AttributeDefinition*. An *AttributeDefinition* is thus typed by this association.

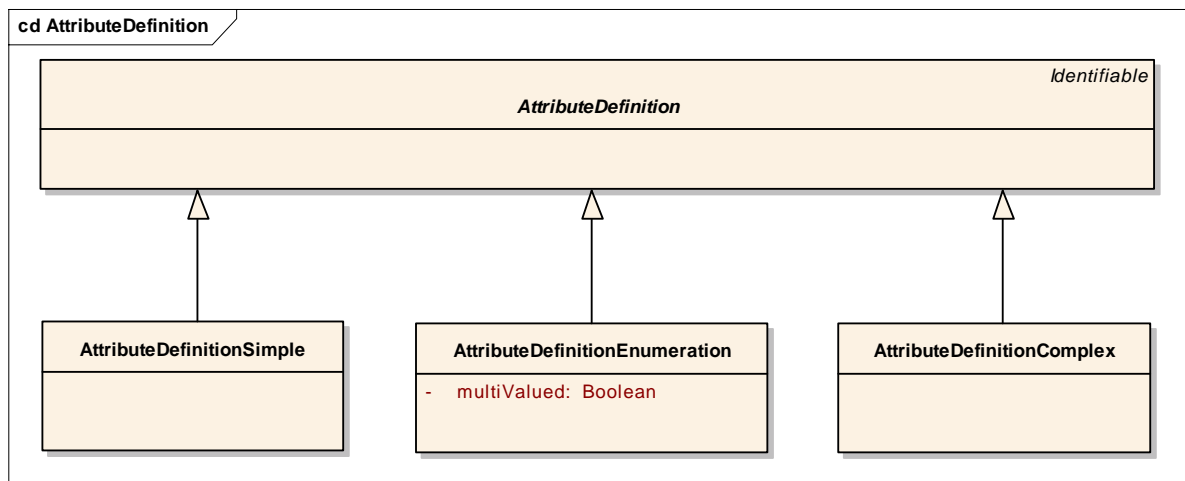
SpecType

An instance of *SpecType* defines a combination of *AttributeDefinition* instances. This is realized by aggregating an arbitrary number of instances of *AttributeDefinition* within a *SpecType* instance.



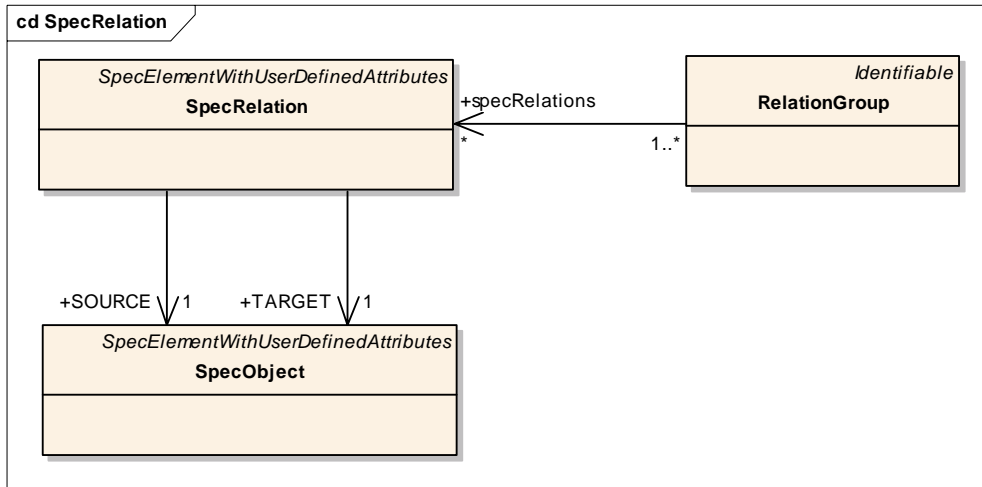
SpecType instances are referenced by instances of specification elements (i.e. instances of *SpecObject*, *SpecGroup*, *SpecHierarchyRoot* and *SpecRelation*). In this way, a combination of attribute definitions is associated with a specification element.

The information type *AttributeDefinition* itself is an abstract super-class for three concrete attribute definitions based on each of the three kinds of data types:



SpecRelation

An instance of *SpecRelation* indicates a relation between two instances of *SpecObject*:



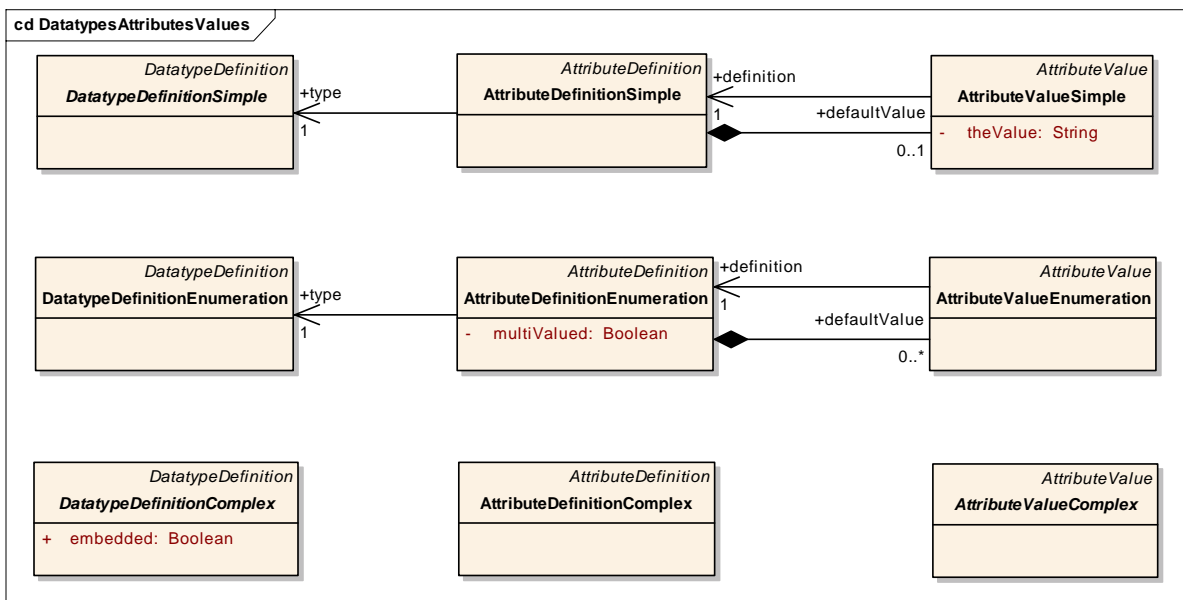
This relation is a uni-directional reference realized by the associations of a *SpecRelation* instance with the role names “source” and “target” to two distinct instances of *SpecObject*.

Instances of *SpecRelation* are themselves grouped together to form types of relations (e.g. “realizes”, “is associated with” etc.). Relation groups are also used for access policies. Groups of *SpecRelation* instances are formed by an instance of *RelationGroup* which has an association to one or multiple instances of *SpecRelation*.

RIF data types

As already mentioned, there are three basic kinds of data types in RIF:

1. Simple data types
2. Enumeration data types
3. Complex data types



Each of these three data type categories has a representation as

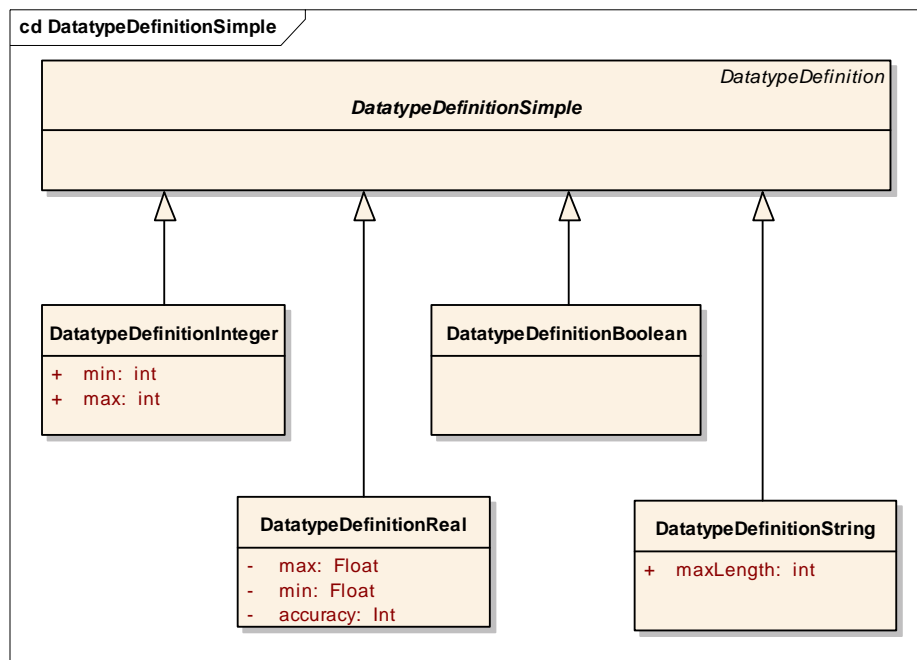
- data type definition,
- attribute definition and
- attribute value.

For simple and enumeration data types, the attribute definition is typed by a data type definition through the respective association. Concrete instances of attribute values have a reference to their respective attribute definition. Furthermore, an attribute definition can have a default value and thus may aggregate an attribute value instance of respective type.

Complex data types are modeled slightly differently from simple and enumeration data types. There are no relations between data type definition, attribute definition and attribute value for the complex data type category. Please refer to Section 0 for the rationale of this concept.

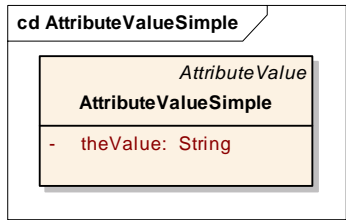
Simple data types

A simple data type is one of Integer, Real, Boolean or String. The concrete information types *DatatypeDefinitionInteger*, *DatatypeDefinitionReal*, *DatatypeDefinitionBoolean* and *DatatypeDefinitionString* therefore inherit from the abstract super-class *DatatypeDefinitionSimple*:



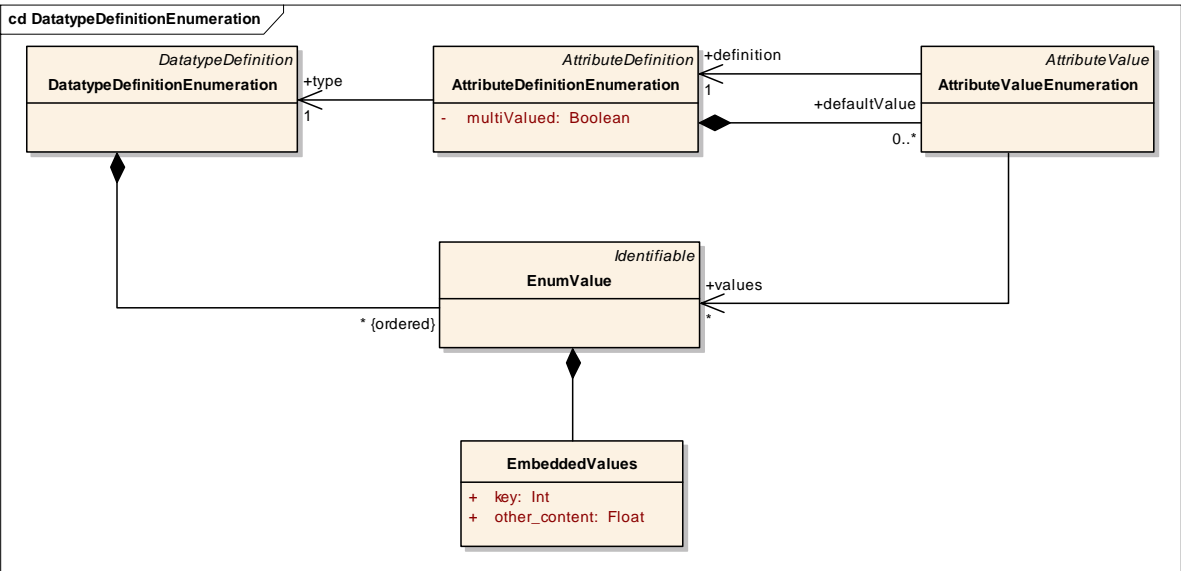
Instances of the concrete data type definitions are further specified by additional element attributes (e.g. “min” and “max” to indicate ranges).

Concrete values of a simple data type are stored within an instance of *AttributeValueSimple* as an element attribute (“theValue”) of type String. In this way, the modeling of data types and attribute values has been simplified but on the other side, no data type checking for attribute values can be performed on the XML-Schema level. This disadvantage is acceptable because the import functionality of an RE/RM-tool has to perform data type checking for all data type categories anyway.



Enumeration data types

RIF supports both single enumeration and multi-value enumeration data types. The enumeration type is indicated by the element attribute “multiValued” inside an information element of type *AttributeDefinitionEnumeration*.



Both single and multiple enumerations types are based on the same set of enumeration values which are instances of *EnumValue*. An enumeration data type is made up of a set of enumeration values. Therefore, an instance of *DatatypeDefinitionEnumeration* aggregates an arbitrary number of *EnumValue* instances.

Concrete enumeration values are stored in an instance of *AttributeValueEnumeration* as associations to the respective *EnumValue* instances. An *AttributeValueEnumeration* instance is defined by an instance of *AttributeDefinitionEnumeration* which also indicates whether the enumeration type is “single” or “multi-value”. Within an instance of *AttributeDefinitionEnumeration*, the set of possible enumeration values is defined by the reference to a *DatatypeDefinitionEnumeration* instance. Furthermore, an instance of *AttributeDefinitionEnumeration* can have a default value which is indicated by the aggregation of an *AttributeValueEnumeration* instance.

Information elements of type *EnumValue* aggregate *EmbeddedValues* instances which contain additional data for the enumeration value (e.g. a key). Please note that the information type *EmbeddedValues* is likely to be extended in the next version of RIF.

Complex data types

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 22 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

Complex data types contain in principle arbitrary structured data. With the exception of XHTML-based data, the deeper meaning and interpretation of complex data is not apparent for the RIF importer and exporter functionality. Complex data is handled on the application level and therefore contains additional information (e.g. file format, associated application type) to enable the intended processing of the complex data.

RIF distinguishes three types of complex data:

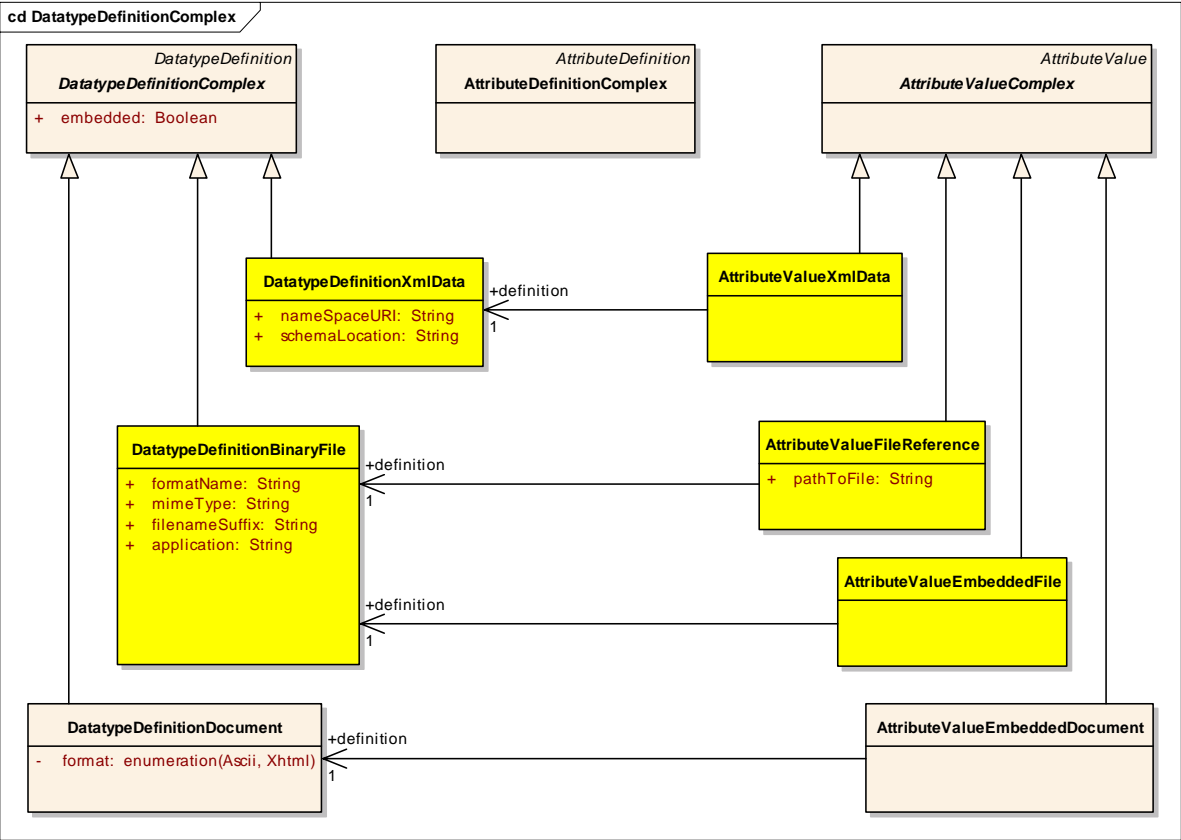
- Document data (i.e. XHTML based data)
- XML based data
- Binary data¹

Although XHTML is in fact a type of XML data, RIF treats XHTML as a complex data type of its own since certain functionality of RIF (e.g. text formatting, object embedding) is realized through XHTML with rules on the use of specific XML elements of XHTML (see Section 0 for further details).

XML based data is “typed“ by an XML-Schema and is embedded within the RIF exchange file in order to simplify its handling.

Binary data can be any data, and thus the structure of binary data is not known to RIF. Binary data is either embedded in an encoded format (e.g. BASE64) within the RIF exchange file or it is stored externally and referenced from within the RIF exchange file.

¹ Please note that for the first release of RIF, the generic XML based data type and the binary data type are not included (information types with yellow background are left out).



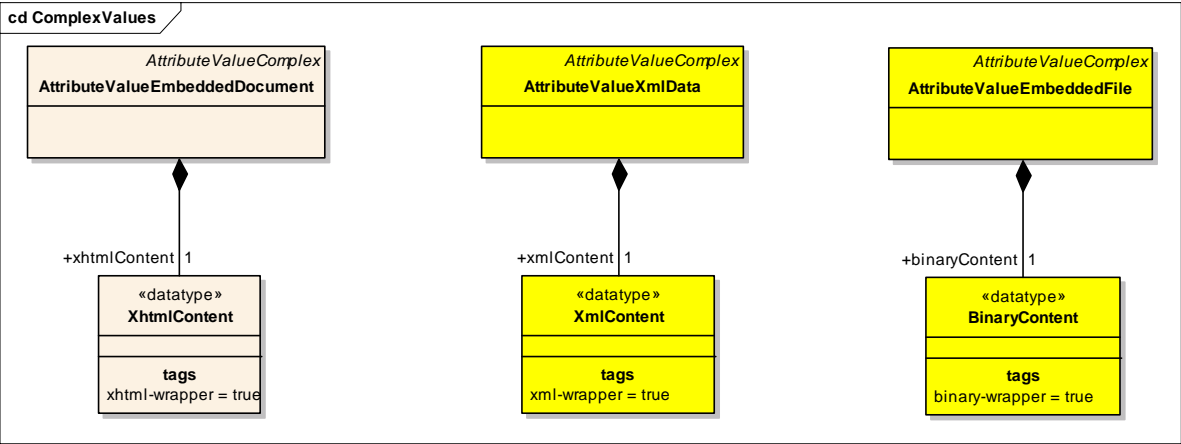
Different from simple and enumeration data types, the attribute definition for the complex category is not typed any further. A complex attribute definition that is associated with a specification object is intended to carry any type of complex data (e.g. pictures, sound files, test cases). Therefore, complex attribute values directly refer to their data type definition.

A complex data type can be either document data, XML data or a binary file. Thus, the information types *DatatypeDefinitionDocument*, *DatatypeDefinitionXmlData* and *DatatypeDefinitionBinaryFile* inherit from the abstract super-class *DatatypeDefinitionComplex*. Instances of these information types have element attributes that further characterize the complex data type:

- For XML based data types, the reference to the XML-Schema has to be specified.
- For binary data, the format name, the MIME type, the file name suffix and the associated application type has to be specified.

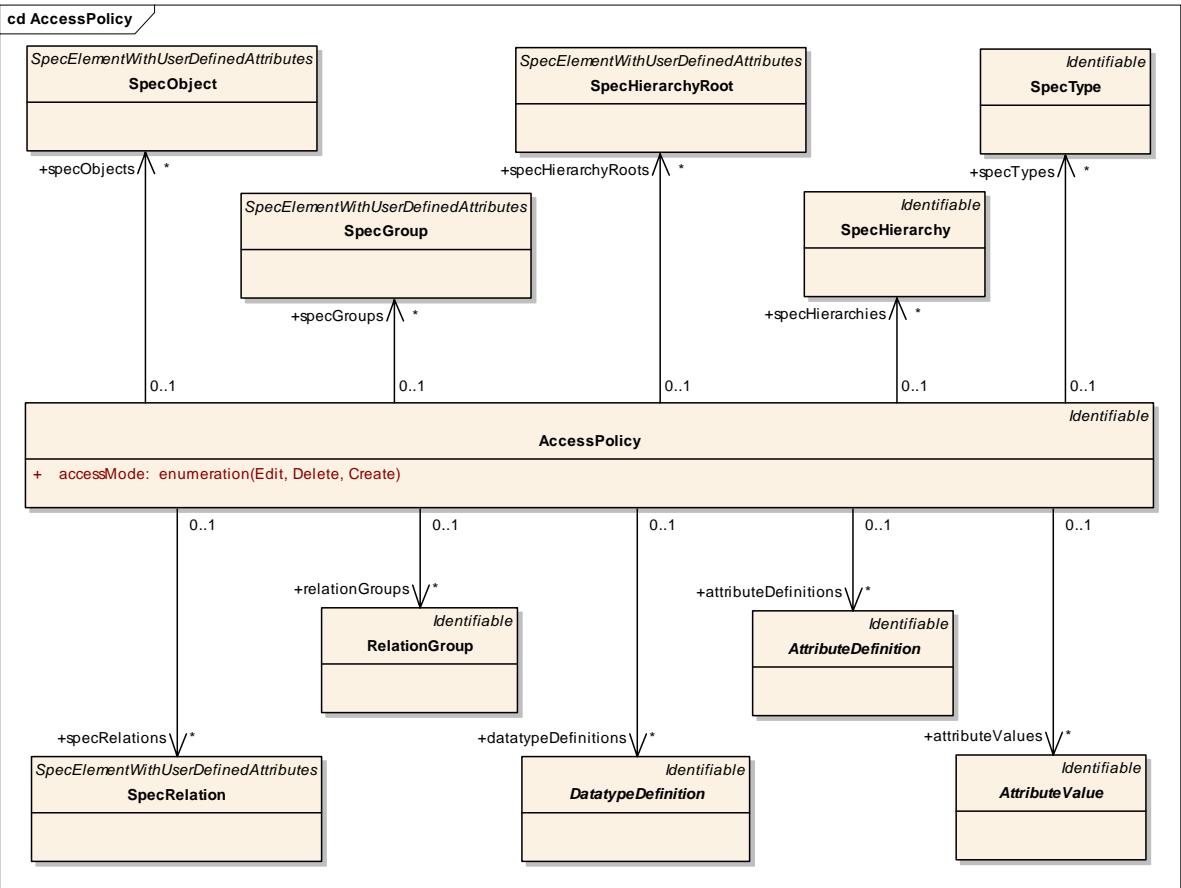
Finally, complex attribute values are instantiated from an information type of either *AttributeValueEmbeddedDocument*, *AttributeValueXmlData*, *AttributeValueFileReference* or *AttributeValueEmbeddedFile* depending on the type (document, XML or binary) and location (embedded or external) of the complex data.

Please note that instances of *AttributeValueDocument*, *AttributeValueXmlData* and *AttributeValueEmbeddedFile* are in principle wrappers for embedded document, XML or binary content, respectively. The storage of these contents is out of scope of the RIF model. This is indicated to the conversion scripts using the tags “xhtml-wrapper”, “xml-wrapper” and “binary-wrapper”.



Access policies

RIF supports access policies that are valid for the receiver of the RIF exchange file. In RIF, many types of information elements can be associated with a definition of access rights (i.e. an instance of *AccessPolicy*). Please note that in the diagram below, the relationships among the information types that are referenced by *AccessPolicy* are hidden due to simplification reasons.



HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 25 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

Dependent on the type of the information element, the associated access policy that is defined as alternatively “Edit”, “Delete” or “Create” have different interpretations and semantics.

A generic rule for access policies is that they are inherited along aggregations but **not** along associations. In this way, an access policy is also valid for the information elements that are located within the information element that is associated with the access policy.

This inherited access policy is valid unless a different access policy overrides it. (This concept is equivalent to the scoping rules of local and global variables in modern programming languages.)

The following table specifies the application of the access policy for the relevant information types:

Information type	Application of the associated access policy	Access policy is inherited by
SpecObject	<ul style="list-style-type: none"> Renaming of the SpecObject instance Changing of the associated SpecType instance 	<ul style="list-style-type: none"> AttributeValue instances for SpecObject
SpecGroup	<ul style="list-style-type: none"> Renaming of the SpecGroup instance Changing of the associated SpecType instance Adding/deleting of SpecObjects instances to/from the SpecGroup instance 	<ul style="list-style-type: none"> AttributeValue instances for SpecGroup RelationGroup instances
SpecHierarchyRoot	<ul style="list-style-type: none"> Renaming of the hierarchy structure Modification of headers of the main chapters (1st level instances of SpecHierarchy) 	<ul style="list-style-type: none"> AttributeValues instances for SpecHierarchyRoot 1st level instances of SpecHierarchy
SpecHierarchy	<ul style="list-style-type: none"> Restructuring, adding and deleting of sub-chapters within the chapter for which the access policy is defined 	<ul style="list-style-type: none"> SpecHierarchy instances (recursively)
SpecType	<ul style="list-style-type: none"> Renaming of the SpecType instance Adding and deleting of AttributeDefinition instance to/from the SpecType instance 	<ul style="list-style-type: none"> AttributeDefinition instances
SpecRelation	<ul style="list-style-type: none"> Renaming of the SpecRelation instance Changing of the associated SpecType instance Changing of the source and target associations to SpecObject instances 	<ul style="list-style-type: none"> AttributeValue instances for SpecRelation
RelationGroup	<ul style="list-style-type: none"> Renaming of the RelationGroup instance Changing of the associated SpecType instance Adding/deleting of SpecRelation instances to/from the RelationGroup instance 	
DatatypeDefinition	<ul style="list-style-type: none"> Renaming of the DatatypeDefinition instance Modification of the DatatypeDefinition instance 	
AttributeDefinition	<ul style="list-style-type: none"> Renaming of the AttributeDefinition instance Changing the reference to the associated DatatypeDefinition instance 	
AttributeValue	<ul style="list-style-type: none"> Modification of the content of the AttributeValue instance 	

XHTML name space

As already mentioned previously, XHTML is a complex data type on its own with rules on the use of certain XHTML elements. The term “XHTML name space” is used here because for data based on the complex XHTML data type, the name space within the exchange file is temporarily switched from the RIF name space to the XHTML name space. This allows validating against different XML-Schemas (i.e. against the RIF-Schema and against an XHTML-Schema) resulting in more independency between the different XML-Schemas.

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 27 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

For RIF, the XHTML name space is based on the XHTML specification version 1.1 – also known as “Module-based XHTML”. The module based approach allows for integrating only those parts of XHTML that are necessary for the desired functionality.

Text styles

For RIF, the functionality for describing text styles is needed. Therefore, the XHTML name space in RIF includes the “Text Module” and the “Edit Module” of XHTML 1.1.

The following tables give an overview over the tags specified in the “Text Module” and in the “Edit Module” of XHTML 1.1:

Text Module:

Tag	Comments
<abbr>	The abbreviation tag is simply a descriptive element. For most browsers it has no affect on style or semantic structure.
<acronym>	Same as above.
<address>	This tag is a descriptive element used to supply contact information for a document or a major part of a document such as a form. It's often used at the beginning or end of a document.
<blockquote>	This tag is used for quoted text - or more precisely for long quotations (block-level content) that require paragraph breaks. Most browsers display blockquote content with margin indentations.
 	This tag is used to force a line-break.
<cite>	This tag is useful to mark a quotation.
<code>	This tag is a descriptive element, which causes most browsers to present the text in a font type with a fixed font width.
<dfn>	The definition tag is a descriptive element and most browsers display this in an italic style.
<div>	This tag, in conjunction with the id and class attributes, offer a generic mechanism for adding structure to documents and defines content to be block-level.
	The emphasis tag is a descriptive element and most browsers display this in an italic style.
<h1>	This tag is used to mark the largest size of a header.
<h6>	The tag is used to mark the smallest header size.
<kbd>	The keyboard tag emulates old typing machine font style with fixed character width.
<p>	This tag is used to mark a paragraph.
<pre>	When a specific content is to be presented <i>exactly</i> as it is, this tag is very useful.
<q>	The quotation tag is purely a descriptive element to mark a quotation, especially intended for short quotations (inline content).
<samp>	The sample tag is a descriptive element, which most browsers display with a fixed

font width.

This tag, in conjunction with the id and class attributes, offers a generic mechanism for adding structure to documents and defines content to be inline.

This tag is a descriptive element and most browsers display the content in a bold style.

<var>

The variable tag is a descriptive element and most browsers display this in an italic style.

Edit Module:

Tag	Comments
	DEL is used to mark-up sections of the document that have been deleted with respect to a different version of a document.
<ins>	INS is used to mark-up sections of the document that have been inserted with respect to a different version of a document.

The tags from both of these modules define logical mark-ups for the text and not directly its physical presentation. For the physical presentation, the common implementation (e.g. as realized in Microsoft's Internet Explorer) must be adopted.

Object embedding in the XHTML name space

With RIF, it is also possible to embed arbitrary objects (e.g. pictures) within text. This functionality is based on Microsoft's clipboard and OLE (Object Linking and Embedding) technology and is realized within the XHTML name space by including the "Object Module".

The following table gives an overview of the XML element and attributes from the Object Module that are used for embedding objects:

Element	Attributes	Attribute types	Minimal Content Model
Object	classid data height name type width	URI URI Length CDATA ContentType Length	(PCDATA Flow param)*

3.2 Tool Interfaces

3.2.1 Typical Scenarios for importing/exporting

3.2.1.1 Guide to implement a user interface.

This is a proposal to guide the user through a workflow of dialogs and information how to set parameters for the transfer process.

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 29 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

A transfer process is divided into export and import whereas both sub-processes must be repeatable in several ways. An export is repeatable as a re-export of the same set of requirements elements and attribute subset as a new version containing the current modifications. A re-export of old versions is not supported by RIF. Exporting old versions must be covered by the tool feature to restore older versions from the history data.

An import process is directed at a creation, restore or update transaction to generate a (new) version of requirements data within the target tool.

To manage this repeatability it is necessary for both involved tools to save additional configuration settings for each transfer process.

3.2.1.2 Sample export process:

1. A project member (sender) runs an Export Wizard
2. The sender selects a set of requirement documents containing the requirements and the subset of attributes to interchange. The sender sets access policies and access rights to allow the recipient to edit certain attributes In case of a re-export the sender can load an existing configuration of all settings from file or database and continue with step 6 if desired
3. The sender selects a committed character set
4. The sender sets RIF-file based information e.g. comments, document title, validation date
5. In extension, the sender sets re-usable project or organisational meta information stored in the tool to structure and organize different sets of interchange configurations
6. The sender sets destination path and RIF file name
7. The tool generates its source Tool-ID to support enhanced importing features on the recipient's tool
8. The tool generates SO Interchange-Ids for all data elements for a first interchange or re-uses existing SO Interchange-ID's in case of a re-export
9. Save settings including SO Interchange-Ids in the tool database or a file at a user-defined location to be able to re-export the same set of requirement data later
10. Start export and save requirement data into the RIF file

3.2.1.3 Sample import process:

1. A project member (recipient) runs an Import Wizard
2. Select a name and path to the RIF file to be imported
3. The tool verifies the file check-sum to assure completeness of the RIF file
4. The tool verifies consistency of the RIF file according to the XML schema
5. The importing tool checks the source tool-ID to switch on enhanced importing features it may support
6. For the first import: A decision how and where to integrate the RIF file content within the target tool element structure is made by the user
7. In case of re-import: The tool compares the scope of the RIF file with the one defined in the stored setting file to identify new or deleted entries.
8. In case of re-import: The tool evaluates the changes between the versions of the requirement content in the RIF file and the internal database. It indicates violations of the access rights defined in the setting file which was used for the first export/import. Optional, changes made by the sender are accepted or rejected with a new version
9. Execute import (errors are handled)
10. Store the scope and settings of the imported RIF file in the tool database or a file at a user-defined location to re-use it for a later re-export.

11. Create a log file with all settings, warning, and error messages

3.2.1.4 Warnings and error handling

Special export warnings and errors

- Versioning: Re-export of an already old version
- Modified export configurations
- Empty attributes must be provided with default values

Special import warnings and errors

- Versioning: Re-import an already integrated (old) version
- Logical content:
 - Changes made to the exported set of access policies
 - Missing default values for all transferred elements and attributes
 - Deleted requirements since previous transfer process

Appendix A: Glossary

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
3.3 Glossar: RIF-Project	n.a.	-
3.3.1 list of abbreviations	Hint	-
AD	Attribute Definition	-
AO	Attribute Object (outdated)	Homonym: - AV.
AP	Access Policy	-
AS	Answer Specification	-
AT	Attribute Type	-
AV	Attribute Value	-
AccPol	Access Policy	-
AnSpec	Answer Specification	-
AttDef	Attribute Definition	-
AttVal	Attribute Value	-
CRS	Customer Requirements Specification	-
DB	Database	-
DT	Data Type	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
DTD	Document Type Definition	-
EF	Exchange File	-
EFF	Exchange File Format	-
EP	Exchange-Process	-
ID	Identity	-
InfoElem	Information Element	-
InfoType	Information Type	-
InitSpec	Initial Specification	-
NFR	Non-Functional Requirement	-
OrigSpec	Original Specification	-
OEM	Original Equipment Manufacturer	-
PP	Project Partner	-
PreSpec	Previous Specification	-
RD	Requirement Data	-
RO	Relation Object (outdated)	Homonym: - SpecRel.
RT	Requirement Type	-
RelGroup	Relation Group	-
RelType	Relation Type	-
Req	Requirement	-
ReqSpec	Requirement Specification	-
ReqSpecDoc	Requirement Specification Document	-
RIF	Requirements interchange Format	-
SA	Specification Attribute	-
SE_A	Specification Element with User Attributes	-
SG	Specification Group	-
SH	Specification Hierarchy	-
SHO	Specification Hierarchy Object	-
SHR	Specification Hierarchy Root	-
SO	Specification Object	-
SOIID	Specification Object Interchange-ID (SO-Interchange-ID)	-
SR	Specification Relation	-
ST	Specification Type	-
SpecElem_A	Specification Element with User Attributes	-
SpecGroup	Specification Group	-
SpecHierarchy	Specification Hierarchy	-
SpecHierarchyRoot	Specification Hierarchy Root	-
SpecObject	Specification Object	-
SpecRelation	Specification Relation	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
SpecType	Specification Type	-
SuD	System under Development	-
SysRS	System Requirements Specification	-
ToC	Table of Content	-
UpSpec	Updated Specification	-
WP	Work Package	-
3.3.2 Object- Words	Hint	-
<u>Access Policy</u>	<p>The AccPol defines</p> <ol style="list-style-type: none"> Access Rights and Modification Rights to (parts of) the EF (see separate definitions). <p>For this reason, AccPol is related to 10 classes of the EFF:</p> <ol style="list-style-type: none"> SpecHierarchy, SpecHierarchyRoot, SpecObject, SpecRelation, SpecGroup, SpecType, Relation Group, Attribute Definition, XML Schema, Attribute Value. <p>AccPol is an <i>optional part</i> of the EFF, while the default will be "read only".</p>	<p>see also:</p> <ul style="list-style-type: none"> - Access Right, - Modification Right, - SH, - SHR, - SO, - SR, - SG, - ST, - RelGroup, - AD, - XML Schema, - AV, - EFF. <p>Synonym:</p> <ul style="list-style-type: none"> - Authorisation.
Access Right	<p>The EFF must support that only authorised person have Access- and Modification Rights as defined in the Contract.</p> <p>The Access Policy is divided into 2 logical divisions:</p> <ol style="list-style-type: none"> Access Right, Modification Right. <p>If a PP is allowed to <i>view/see/read</i> the EF, this PP must have Access Rights.</p> <p>Access Rights can be assigned to Groups.</p> <p>Access Rights are inherited along aggregations, but <i>not</i> along associations; this inheritance is overridden by new Access Policy entries deeper inside the tree of XML elements.</p>	<p>see also:</p> <ul style="list-style-type: none"> - AccPol, - User Group. <p>Dissociate from:</p> <ul style="list-style-type: none"> - Modification Right.
Answer Specification	<p>Within an Exchange-Process, an AnSpec is produced by the Supplier(s), while its origin is the InitSpec produced by the Customer(s).</p> <p>An AnSpec consist of additional information like, for instance, comments, questions and implementation details and is related to its InitSpec.</p> <p>After re-importing the Answer Specification into the Customer(s)' RM-Tool, the changes between Initial Specification and Answer Specification can be viewed separately, while the re-imported Answer Specification becomes an Initial Specification of incremented Version in the RM-Tools's DB.</p>	<p>see also:</p> <ul style="list-style-type: none"> - Initial Specification, - Exchange-Process. <p>Synonym:</p> <ul style="list-style-type: none"> - AS, - Updated Specification. .
Attribute	An Attribute is assigned to an Object (for instance an SO a Group) and enlarges the information that is available to its assigned Object.	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
<u>Attribute Definition</u>	An AttDef. is valid for a (group of) SE_A. For this (group of) SE_A(s), an Attribute Definition defines 2 things: 1. The name of the Attribute it defines, and 2. The Attribute Type. Any AttDef. encloses a default value for any AT. Each AD is aggregated by <i>exactly one</i> SpecType. Each AD is associated with <i>exactly one</i> XML-Schema (here: AT). Each AD aggregates <i>one or zero</i> (0..1) Attribute Value. Each AD is associated by <i>exactly one</i> Attribute Value.	see also: - SE_A, - AT, - ST, - AV.
Attribute Type	Each AO is an instance of an AT.	see also: - AO, - DT. Dissociate from: - AttDef.
<u>Attribute Value</u>	The information a AO represents, while the format of this information is defined by it's AT. An AV can give information of a value or its reference if the AV is used for expressing Relations.	see also: - AT, - AO, - Attribute. Homonym: - Attribute Object (outdated).
Authorisation	This term is identified as being too unspecific; please refer to a) Access Right b) Modification Right instead!	NOTE: s. CustomerRequirements, XRMEML_Bs262.
Baseline	A Baseline is a means to a) group and b) version Requirements within RM-Tools.	-
Basic Subset	(1) The <i>Basic Subset</i> describes a subset of the EFF. It brings out the basic functionality of the EFF that every (meaningful) RM-Tool shall be able to support. (2) A <i>SubSet</i> defines a group of Requirements. The EFF supports re-use of such SubSets for another Export-Process.	see also: - SubSet, - Export Profile.
Change History	A function by a RM-Tool that monitors changes of Requirements over time. As a minimum, it enables to recapitulate <i>which changes</i> have been performed <i>by whom</i> at <i>which point-in-time</i> . Comments on Requirements are monitored explicitly.	-
Character Set	The EFF supports the definition of Character Set(s) to be used for displaying the Requirements included in an EF like UTF8, UTF16, ISO8859, ANSI, Unicode for instance.	see also: - EFF, - EF. Homonym: - CharSet.
Class	See "Requirement Type".	Homonym:

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
		- Requirement Type.
Class "Identifiable"	A special Specification Type with the purpose to identify all Specification Objects that are suitable for the transformation process. Note: Some of its Attributes are "LongName" and "ShortName", while the latter is probably destined to be deleted or altered by Carmeq, soon.	-
Coding	Means to define the representation of information.	-
Comment	Each comment against a RS decides, whether the commenting side accepts, refuses, counter-proposes, comments or questions a Requirement of the RS. A Comment can be made by Customer(s) as well as Supplier(s). A Comment must be linked-up to its corresponding Object.	-
Consolidation-Process	If a Requirement is reviewed by more than one Reviewer, all the single comments need to be integrated to one, consolidated comment and/or requirement.	-
Country Code	The EFF shall enable to add a Country Code to any EF, for instance for the sake of displaying numeric values or dates in an unambiguous, familiar way.	-
Contract	Binding agreement between Customer(s) and Supplier(s) which is basis for the development process of the SuD.	-
Customer Requirements Specification	The CRS is part of the RS.	see also: - RS.
Database	-	Datenbasis.
Data Type	The Data Type of each Specification Attribute defines the values a Spec Attribute might have.	Synonym: - Type.
Due Date	Specified point-in-time, when a certain WP has to be completely processed.	see also: - WP.
Exchange-Process	An Exchange-Process is defined as the combination of an Export-Process, a Transfer-Process and an Import-Process. The exchanging parties must be defined within the Contract. The subject of exchanging is a RSD, which is called Initial Specification on Customer(s)'s side, and Answer Specification on Supplier(s)'s side. If the Answer Specification is re-imported on Customer(s)'s side, it is an Initial Specification of incremented Version. An Exchange-Process between more than two partners is divided into multiple, bi-lateral exchanges.	-
Exchange File	Means of transferring (parts of the) Requirement Specification Document(s) via an Export- and Import-Process between the RM-Tools of the Customer(s) and the Supplier(s) to exchange Requirement Specifications, while each EF can contain information of multiple origins. Each EF meets the Requirements of the EFF.	see also: - Exchange File Format, - SHR.
Exchange File Format	The format of the Exchange File(s). The EF can be regarded as an instantiation of the EFF. The EFF specifies the data structure for storing requirements together with e.g. comments, answers, links and history. While the format extensively specifies all necessary data structures, an EF according to the EFF may contain only a part of the structures defined in the	see also: - Exchange File, - exchange, - Country Code - Character Set.

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	<p>EFF.</p> <p>Actually it is intended that the EFF shall support accessibility/visibility, ownership and changeability information, which includes the necessity to support user-roles as well.</p> <p>The EFF supports Country Codes as well.</p> <p>The EFF supports the definition of Character Set(s).</p> <p>The EFF supports means to avoid any data contradiction.</p> <p>The EFF must be easily convertible to other appropriate formats and structures, for instance using XSLT.</p> <p>The method how the EFF will be developed must derive from the AutoSAR method.</p>	
Export-Process	<p>A process to extract Requirement Specification Document(s) from the RM-Tool to a file in EFF.</p> <p>To generate this document, Sequence Information and Mark-Up-Information is available.</p>	<p>see also:</p> <ul style="list-style-type: none"> - Import-Process - Partially Export - Incrementally Export.
Export Interface	<p>The Export Interface must support Export-Processes that create EF in EFF.</p> <p>Vendors of RM-Tools are invoked to implement a suitable interface that fulfils the RIF-Requirements.</p>	-
Export Profile	<p>The EFF must support SubSet information of all objects included in an EF.</p>	<p>see also:</p> <ul style="list-style-type: none"> - SubSet.
Frequency	<p>How often a UseCase is expected to occur.</p>	-
Homonym	<p>A word, standing for more than one meaning.</p> <p>Aristotle defined it this way: Two things are homonym if they have the same notation and different definitions.</p> <p>http://de.wikipedia.org/wiki/Homonym</p>	-
Identity	<p>Each ID is related to a Requirement (or any other Specification Object of an EF) and represents a means to identify the assigned Requirement in an unequivocal way.</p> <p>IDs must be available for Attributes and Types of Attributes.</p>	-
Import-Process	<p>A process to include Requirement Specification Document(s) into a RM-Tool's DB.</p> <p>Counterpart of the Export-Process. After importing ReqSpec in EFF, the ReqSpec is in the DB of the importing party's RM-Tool and mapped to its content.</p> <p>Main goal of the import part of an Exchange-Process is to enable the importing contractual party to comment, question and answer Requirements and Comments of any other partner while traceability is ensured, simultaneously.</p>	<p>see also:</p> <ul style="list-style-type: none"> - importing - Export-Process
Import Interface	<p>The Import Interface must support Import-Processes that create EF in EFF.</p> <p>Vendors of RM-Tools are invoked to implement a suitable interface that fulfils the RIF-Requirements.</p>	-
Incremental Export Process	<p>Iterative workflow to extract and transform data from a source into another format.</p>	-
Information Element	<p>Information Elements are all the Objects enclosed in an EF. Every InfoElem is an instance of a InfoType.</p>	<p>see also:</p> <ul style="list-style-type: none"> - InfoType.
Information Type	<p>Every Information Element is an instance of a Information Type.</p>	<p>see also:</p> <ul style="list-style-type: none"> - InfoElem.

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
Inheritance	Classes inherit state and behavior from their superclasses. Inheritance provides a mechanism for organizing and structuring software programs.	-
Initial Specification	An exported Specification is called Initial Specification prior supplementation of the receiver which makes it an Answer Specification. When the Answer Specification is re-imported by the original sender, the Answer Specification becomes an Initial Specification of an incremented Version. All changes between Initial Specification and Answer Specification must be visible within all PP's RM-Tools without destroying probably previously made comments and links.	see also: - Answer Specification, - Exchange-Process. Synonym: - Original Specification, - Previous Specification (when re-imported by initial sender).
Integration-Process	A (distributed) Exchange-Process is usually followed by an Integration-Process, where the various EF ("Answer Specifications") are finally re-imported by the original sender for integration purpose. Various Requirements and origins of the Requirements, Comments, Links, changes, additions, deletions, structural changes between CRS and the various SysRS are integrated here, while no information is allowed to be lost during integration process (see Merging-Process). After the Integration-Process, the various Answer Specifications become one Initial Specification of incremented Version. An Integration-Process needs a minimum of two PPs.	see also: - Exchange-Process, - EF, - RSD, - Merging-Process.
Interface	Means of a RM-Tool to export and import EF.	see also: - Export Interface, - Import Interface.
Layout	A Layout defines, how the Requirements are arranged in any kind of coherent Document, for instance for a print-out. Layout information have their own Version.	See also: - Style.
Link	A means to explicitly communicate logical or structural dependencies among Requirements.	Homonym: - Relation.
Mapping-Process	Between the exporting and the importing RM-Tool a mapping for the individual attributes must be specified. The Mapping-Process is part of the Merging-Process and related to the technical aspect of merging data. To ensure this process runs automatically, a Mapping Table exists.	see also: - Mapping Table, - Exchange-Process, - Merging-Process.
Mapping Table	A Mapping Table defines for each Object within an EF where it must be stored in the DB of the importing RM-Tool and vice versa. This mapping must be configurable.	Synonym: - Mapping Tabelle.
Mark-Up	A type of data, not yet consolidated.	-
Mark-Up Information	A type of content, not yet consolidated.	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
Merging-Process	A Merging-Process needs exactly two PP, while there's one sender and one receiver; the latter merges the receiving Answer Specification to his Initial Specification within the DB of his RM-Tool. A Change History will document the origin of each Requirement and Comment in both versions including all it's Attributes, changes, additions, deletions, structural changes. This is valid for the merging result as well. The Merging-Process includes the Mapping-Process.	see also: - Initial Specification, - Answer Specification, - Mapping-Process, - Integration-Process.
Meta Information	Additional Information on Specification Objects needed for processing by the RM-Tools and/or the EF. This kind of information is invisible for the User by default.	-
Modification Right	The EFF must support that only authorised person have Access- and Modification Rights as defined in the Contract. The Access Policy is divided into 2 logical divisions: 1. Access Right, 2. Modification Right. If a PP is allowed to <i>create, edit and delete</i> Objects within the EF, this PP must have Modification Rights. Modification Rights can be assigned to SpecGroup(s). Modification Rights are inherited along aggregations, but <i>not</i> along associations; this inheritance is overridden by new Access Policy entries deeper inside the tree of XML elements.	see also: - AccPol, - SpecGroup, - EFF. Dissociate from: - Access Right.
Non-Functional Requirement	(here: for instance Performance, ...)	-
Object	Objekt	see also: - AO, - RO, - SHO, - SO.
Original Specification	This is a Synonym for Initial Specification and defined accordingly.	Synonym: - Initial Specification, - Previous Specification.
Ownership	s. CustomerRequirements, XRMEML_Bs262.	-
Performance	The maximum duration a UseCase or Import- or Export-Process should take.	-
Previous Specification	This is a Synonym for Initial Specification and defined accordingly.	Synonym: - Initial Specification. see also: - Updated Specification, - Exchange-Process.
Primary Actor	The contractual party who wants to have the specification written. Usually, this is the Customer(s).	-
Priority	How critical a Requirement is to the SuD.	-
Proof Reader	Reviser	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
Relation	A means to explicitly communicate logical or structural dependencies among Requirements. A RT defines for its instances, the RO's, which Attributes the Relation has.	Homonym: - Link. see also: - RT, - RO.
<u>Relation Group</u>	A RelGroup defines whether it is allowed to create and add new Relations to the Requirements in the corresponding RelGroup. A RelGroup restricts the creation of Traces <i>starting</i> at the parent elements of their SpecGroup they belong to. A RelGroup restricts the <i>targets</i> of Relations in addition. A RelGroup restricts the SpecType of Objects in the RelGroup. Every SpecObject belongs to a <i>minimum of one</i> (1..n) SpecGroup. Every SpecRelation belongs to a <i>minimum of one</i> (1..m) RelationGroup. A RelGroup assigns <i>exactly one</i> SpecGroup, <i>exactly one</i> SpecType, and 0 to n (0..n) SpecRelations. The RelGroup is aggregated by <i>exactly one</i> SpecGroup.	see also: - SO, - SG, - SR, - ST.
Relation Type	Each Relation Object is assigned to a Relation Type. A Relation Type contains information about its Cardinality. Multiple Relation Types can be used for different kinds of Relations. Any RelType is realised by a SpecType.	see also: - SpecType.
Requirement	This term constitutes a fundamental, logical element of Requirements Engineering (see www.hood-group.com for details). Requirements are collected in Requirement Specifications.	see also: - Requirement Specification.
Requirement Data	The Specification(s) are organised in exchangeable RM-Tool data.	-
Requirement Type	Each SO is an instance of a RT. A multitude of RT shall be able to define for the EFF. Each RT contains 3 things: 1. its Name, 2. its ID, 3. the defined Attributes per instantiated SO. Any ReqType is realised by a SpecType.	Synonym: - Class. see also: - SO, - SpecType.
<u>Requirements Interchange Format</u>	This term is a Homonym to EFF. While "EFF" referred to the technical aspect of name-giving to the SuD of this project, the term "RIF" satisfies, for instance, marketing aspects as well. "EFF" can be seen as a placeholder-term during the development process, while "RIF" must be seen as the final, consolidated term.	Homonym: - EFF. see also: - SuD.
Reviewed Specification	A Original Specification, send from Customer(s) to Reviewer(s), supplemented with review comments.	-
RM-Tool	A SW-Tool for specifying and managing Requirements, explicitly with the possibility to exchange Requirements. A RM-Tools does not need to be available at all contracting parties, but the functionalities are mandatory. The RM-Tool is built upon and around a DB where all the Specifications and parts of Specifications are edited and finally integrated. This integration is done within the RM-Tool's DB on Customer's side.	-
Roll-Back Point	A Roll-Back Point can be defined or set for the sake of later reuse, for instance for handling errors.	Homonym: - Fall-Back Point.
<u>Root Element</u>	A Root Element is the structural and logical basis of every EF. It consists of the following 7 parts :	see also: - EF,

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	1. TimeStamp(s), 2. XML-Schema(s), 3. SpecType(s), 4. SpecObject(s), 5. SpecRelation(s), 6. SpecGroup(s), 7. SpecHierarchyRoot(s).	- TimeStamp, - XML-Schema, - ST, - SO, - SR, - SG, - SHR.
RelaxNG	see http://www.lexikon-definition.de/RELAX-NG.html	-
Release	A Release is a version of the whole RSD.	-
Requirement Specification	A means to define the goals the SuD has to meet when the development is finished. This goals are defined by single Requirements, organised in a Requirements Specification. For this project, two basic types of Requirements Specifications exist: CRS and SysRS. A ReqSpec usually exists within the DB of RM-Tools, RSD's or printout versions of a RSD, while the latter two can be part of an Exchange-Process.	see also: - SuD, - CRS, - SysRS, - RSD, - Exchange-Process. Synonym: - Specification, - Spec.
Requirement Specification Document	A Document containing Requirement Specifications. Each ReqSpecDoc can exist in two ways: a) as a EF, b) as a printed EF and can be part of an Exchange-Process. Within such a process, a RSD can be of one of the following Specification types: - Initial Specification, - (Original Specification), - (Previous Specification), - Answer Specification, - (Updated Specification), while the terms in brackets refer to Synonyms.	see also: - SysRS, - CRS, - RS, - EF, - Exchange-Process. .
Requirements Group	The EFF supports clustering of Requirements in Groups. Each Group has its own Version-Information.	Synonym Specification Group, - SG.
Review-Process	Customer(s) send Initial Specification(s) to one or more Reviewers. A Reviewer must be part of the PP. Every Reviewer then comments the received Specification on basis of single Requirements. After that, the Reviewers send this Answer Specification back to the Customer(s). If there is more than one Reviewer involved, each Reviewer's comments must be visible at the Customer's RM-Tool to support the succeeding Integration-Process. The Review-Process is usually followed-up by an Integration-Process.	see also: - Reviewer, - Initial Specification, - Answer Specification, - Integration-Process.
Sequence Information	For each Requirement included in an EF, the hierarchy- and the Sequence-information is stored in each object while nothing is said weather this information is stored separately or in combination, yet.	see also: - EF, - Hierarchy.
Specification Attribute	A Specification Object has 1 to n Attributes (one for the ID is mandatory). Specification Attributes are based upon the related	Spec Attribute, SpecAttribute.

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	Specification Type of the Specification Object it belongs to. User defined Attributes exist as well as Interface-relevant Attributes.	
<u>Specification Element with User Attributes</u>	The SE_A is a primary object within the EFF. It inherits its (user defined) Attributes to 4 Classes: 1. SpecObject, 2. SpecRelation, 3. SpecGroup, 4. SpecHierarchyRoot. The SE_A associates <i>exactly one</i> SpecType. Each SE_A aggregates 0 to n (0..n) AV(s).	see also: - EFF, - SO, - SR, - SG, - SHR, - ST, - AV.
<u>Specification Group</u>	A SpecGroup is a means to combine SpecObjects in groups. Each Spec Object must be member <i>in a minimum of one</i> (1..n) SpecGroup(s). Each SG can associate 0 to m (0..m) SO(s). Each SG aggregates 0 to x (0..x) RelationGroups, while each RelGroup is aggregated by <i>exactly one</i> SG. Each SpecGroup is associated by 0 to y (0..y), while a RelationGroup associates <i>exactly one</i> SpecGroup. A SpecGroup inherits from SE_A. Each SO has its own Version-Information. Each SpecGroup inherits to Attributes of SpecGroups. Each SpecGroup inherits to Attributes of RelGroups. A SpecGroup does not inherit content of Requirements and its Attributes.	see also: - SO, - SE_A.
<u>Specification Hierarchy</u>	Each SpecHierarchy represents <i>exactly one</i> sub-headline of the ToC of the EF where it is included. A SpecHierarchy contains information, related to the <i>structure</i> of the EF, not related to the <i>content</i> of the EF. It contains information whether the related chapter of the EF is allowed to supplement and/or alter. Any EF has exactly one SHR, while the SHR aggregates 0 to n (0..n) SH(s). Any SH can aggregate further SH(s) (recursive inheritance). Each SH is associated to exactly one SpecObject.	Synonym: - Hierarchy, - Hierarchie, - Spec Hierarchy, - SpecHierarchy, - Tree-Structur, see also: - EF, - SHR, - SH.
Specification Hierarchy Object	One element of a Specification Hierarchy. Each Spec Hierarchy Object associates exactly 1 Spec Object.	Spec Hierarchy Object.
<u>Specification Hierarchy Root</u>	Each EF contains exactly one SHR. A SHR represents the chapter structure of the EF it is enclosed within. A SHR is one out of 7 aggregations by the EFF. Each SpecHierarchyRoot defines whether main-headlines are alterable and whether the name of the hierarchy is alterable. SpecHierarchyRoot inherits to main-headlines and to the Attributes of the Hierarchy. A SpecHierarchyRoot has no impact on the content (Requirements and Attributes) of the EF and on the Attributes of a SpecGroup. A SHR aggregates 0 to n (0..n) SH(s).	see also: - EF, - EFF, - AP, - SG, - SH.
<u>Specification Objekt</u>	Atomic Object of the Exchange Format (and the Source- and Destination-RM-Tool as well due to Exchange-Process!) that is to be referenced by the RM-Tool in question. Each SO owns 5 basic information of: 1. its content, 2. its datatype,	Synonym: - Requirements Object. see also: - AV,

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	<p>3. its structural dependencies, 4. (1..n) related SA's. 5. its Version/History/Time-Stamp.</p> <p>To give an example, a SO can consist of text, picture or an OLE-Object. It is to be discussed whether SO shall support external links (for instance to a Web-Forum via URL). Every SO is instance of a ST, while the ST determines, which Attributes each SO owns. Every SO inherits to Attribute Values. Each SO is associated by <i>exactly one</i> SH, and is associated by a <i>minimum</i> of 1 (1..n) SG(s).</p>	<p>- SO, - SG, - SH, - ST.</p> <p>Dissociate from: - SpecHierarchy Root.</p>
Specification Object Interchange-ID	Special form of ID. For several exchange transactions reusable unique identity. It is created once by the export interface and unique for all following transactions of the same Requirements Specification Document and its Specification Objects and Attributes.	-
<u>Specification Relation</u>	Every Relation between ROs is represented by SpecRelation(s). Each SpecRelation is member in a <i>minimum of one</i> RelationGroup(s). Each SpecRelation associates <i>exactly two</i> SO, while one of the SRs has the role of the <i>source</i> of the relation, and the other has the role of the <i>target</i> of the referred relation. Every SR inherits to Attribute Values. Every SpecRelation inherits from SE_A.	see also: - RO, - SO, - AV, - SE_A.
Specification Structure	The structure of the Specification in question is to be generated in the RM-Tool of the Customer and can be split-up and send due to several exporting on Supplier's side. The Customer(s) generate(s) a Structure for the Specification and sends it to the Specification Writer(s).	SpecStructure, Spec.Structure.
<u>Specification Type</u>	The SpecType defines its own name and is the data basis for any SpecObject. A SpecType inherits to AV(s) and is allowed to add and delete Attribute Definitions. The ST defines the Attributes a SO is allowed to have. Each SO is one instance of a ST. A ST aggregates 0 to n (0..n) Attribute Definition(s). A ST is associated by 0 to m (0..m) Spec Element With User Attribute(s). SpecTypes realise <i>RelTypes</i> and <i>ReqTypes</i> .	Synonym: - Classes.
Style	A Style defines the appearance of characters, for instance bold, italic, underline. Style information have their own Version.	See also: - Layout.
Subordinate UseCase	Name of a UseCase that has one or more Super ordinate UseCases.	-
Superordinate UseCase	Name of a UseCase from which one or more Subordinate UseCases derived. Hence, the Super ordinate UseCase bifurcates into one or more Subordinated UseCases.	-
Synonym	Two words, standing for the same meaning. Aristotle defined it this way: Two things are synonym if they have the same notation and the same definitions. But the latter definition is outdated, today. Today, a Synonym is understood as another word with the same meaning, whereas <i>strict</i> and <i>non-strict</i> Synonyms exist.	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	An example for a strict Synonym: The German word "Apfelsine" is a Synonym for the German word "Orange", while both have just one corresponding expression in English, which is "orange". [http://de.wikipedia.org/wiki/Synonym]	
System Requirements Specification	It describes the whole SuD in a more technical oriented language. It is divided into Sub System Requirements like sub-assembly or component specification.	-
System under Development	<u>general:</u> The (HW- and/or SW-) System or SubSystem the Customer wants to develop or define requirements for. The SuD is defined in the Contract between Customer(s) and Supplier(s). <u>here:</u> The SuD is the EFF, respectively RIF.	-
Table	An ordered columnar display of data.	-
Table of Content	Like in most books, each EF has its own ToC, represented by an SHR. A SHR aggregates 0 to n (0..n) SH(s).	see also: - EF, - SHR, - SH.
<u>Time-Stamp</u>	Every SpecObject and the EF itself has its own TimeStamp, showing the last point-in-time when it has been altered. Any Time-Stamp consists of the following 10 parts : 1. date of creation, 2. time of creation, 3. date of last modification, 4. time of last modification, 5. date of deletion, 6. time of deletion, 7. date of last synchronisation, 8. time of last synchronisation, 9. date of current format, 10. time of current format. Time-Stamped are generated by the exporting interface and cannot be altered by any user.	see also: - Version, - History. Synonym: - TimeStamp.
Transfer-Process	A Transfer-Process is to transfer an EF via an Export- and an Import-Process between Customer(s) and Supplier(s), while each of both actors appears in the role of a sender as well as a receiver. So the Transfer-Process can be seen as the middle part of an Exchange-Process, appearing in both directions.	see also: - EF, - Export-Process, - Import-Process, - Exchange-Process..
Transformation Process	A workflow of converting one input element into another element	-
Trigger	The Primary Actor generates a structure for the Specification and sends it to the Specification Structure.	see also: - Primary Actor.
Type-Definition	Type-Definitions exist for Requirements and Attributes. These Type-Definitions must be part of the EFF. During the Export-Process, the Types must be reducible to individually defined Subsets. Transferred Type-Definitions have their own version. During the Import-Process, the importing Interface must verify whether the value of its included Objects meet their Type-	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	Definition. Additionally, the origin of any Type-Definition must be enclosed in the EFF. Due to the fact that Type-Definitions are supposed to be alterable, those changes must be traceable throughout an EF.	
Unique ID	Each Unique ID consists 4 parts : 1. Time-Stamp, 2. ID of Import-Interface, 3. ID of Export-Interface, 4. Counter/Version.	-
Update-Process	A update workflow	see also: - updating - Update Specification
Updated Specification	This is a Synonym for Answer Specification and defined accordingly.	see also: - Initial Specification, - Export-Process. Synonym: - Answer Specification.
User Group	The EFF has been suggested to support the assignment of Access Rights to single users as well as to User Groups. Nevertheless, this support has been decided to cancel and will not be part of the EFF, yet.	see also: - Access Right.
Version	Each Spec Object, Spec Group and Spec Hierarchy has its own Version, coded as a timestamp according to ISO 8601.	Differentiation to Timestamp and History!
View	A view is a particular way of looking at a set of data. Typically, a view arranges the records in some order and makes only certain fields visible	-
Work Package	A defined task to be fully processed by a defined person until the WP's Due Date.	see also: - Due Date
XML DocBook Format	A special XML dialect for book	-
XML-Schema	An XML-Schema provides a means for defining the structure, content and semantics of XML documents. XML-Schema is used as one representation of the EFF. It can be used to check the validity of EF(s).	Homonym: - Schema.
XSLT	<i>Extensible Style Language Transformation</i> , the language used in XSL style sheets to transform XML documents into other XML documents or dialects.	-
3.3.3 Roles/ Actors	-	-
Customer	The Customer(s) is placing the order to develop the SuD by the Supplier(s). The Customer instantiates the Contract. The Customer(s) is the over-all owner of the Specification(s). The Customer(s) uses a RM-Tool. The final version of the	Synonym: - OEM, - Manufacturer.

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
	Requirements are to be found within the Customers' RM-Tool's DB.	
Lector	Proof-Reader, engaged by the Customer(s). A Proof-Reader is part of the PP.	Homonym: - Proof-Reader.
OEM	Original Equipment Manufacturer	-
Project Partner	Each of the Customer(s) and of the Supplier(s) listed in the Contract related to build the SuD. Within this Contract, the role of every Customer and Supplier in the development process of the SuD is defined. To divide several PP's, a number might be added, for example PP1 and PP2 might be partners involved in an Exchange Process.	see also: - Customer - Supplier - Contract
Primary Actor	The Trigger for a UseCase. The Primary Actor is one defined party of the Customers involved.	see also: - Trigger
Proof-Reader	A Proof-Reader is part of the PP.	Homonym: - Lector. see also: - PP.
Review	An inspection or examination for the purpose of evaluation or improvement.	see also: - Reviewer
Reviewer	One or more people who are involved in the development process of the SuD for the sake of ensuring quality. A Reviewer might be internal (i.e. Customer) or external (i.e. Supplier, Proof-Reader), but must be part of the group of PP.	see also: - Review - reviewing
Specification Writer	Any person on Supplier(s)'s and/or Customer(s)'s side who is authorized to write (parts of) Requirement Specification Documents and is authorised to exchange them with the Customer. Only the Specification Writer(s) on side of the Customer needs a RM-Tool installed.	-
Sub-Contractor	A contractual party, involved in the development process for the SuD.	-
Supplier	One or more company(s) that is involved by the Customer(s) in a project in order to develop or specify parts of the SuD.	Specification Writer, Sub-Contractor.
User	A person or a piece of SW that has access to the RM-Tool and operates it.	-
3.3.4 Tools	Hint	-
Borland - CaliberRM™	http://www.borland.com/caliber/	-
IBM - RequisitePro™	http://www-306.ibm.com/software/de/rational/	-
Microsoft® Word	http://www.microsoft.com/germany/default.aspx	-
TCP IA - IRqA™	http://www.irqaonline.com/	-
Telelogic - DOORS™	http://www.telelogic.com/	-
3.3.5 Trade marks		
DOORS/ERS™	DOORS/ERS™ is a trademark of Telelogic AB	-
IRqA™	IRqA™ is a trademark of TCP Sistemas e Ingeniería S.L. Spain	-

English Terms and Definitions	Meaning / Explanation	Synonym / Homonym
CaliberRM™	CaliberRM™ is a trademark of Borland Software Corporation	-
Microsoft® Word™	Microsoft® Word™ is a trademark of Microsoft Corporation	-
RequisitePro™	RequisitePro™ is a trademark of IBM Corporation	

Appendix B: RIF Schema

The following sample RIF XML Schema file will not be updated within this document. A current version can be downloaded from HIS-Homepage <http://www.automotive-his.de/his-ergebnisse.htm>

3.4 RIF XML Schema (rif.xsd)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:RIF="http://automotive-his.de/schema/rif"
targetNamespace="http://automotive-his.de/schema/rif" elementFormDefault="qualified" attributeFormDefault="qualified">
<!-- Element generated from meta class: ExchangeFile::AccessPolicy -->
<xsd:element name="ACCESS-POLICY">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="ACCESS-MODE" type="RIF:ACCESS-MODE-ENUM" minOccurs="0"/>
<xsd:element name="SPEC-HIERARCHIES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-HIERARCHY-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-TYPES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-TYPE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="RELATION-GROUPS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="RELATION-GROUP-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="ATTRIBUTE-DEFINITIONS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="ATTRIBUTE-DEFINITION-COMPLEX-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-DEFINITION-ENUMERATION-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-DEFINITION-SIMPLE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="DATATYPE-DEFINITIONS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="DATATYPE-DEFINITION-BINARY-FILE-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-BOOLEAN-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-DOCUMENT-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-ENUMERATION-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-INTEGER-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-REAL-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-STRING-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-XML-DATA-REF" type="RIF:REF"/>

```

```

</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="ATTRIBUTE-VALUES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-VALUE-EMBEDDED-FILE-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-VALUE-ENUMERATION-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-VALUE-FILE-REFERENCE-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-VALUE-SIMPLE-REF" type="RIF:REF"/>
<xsd:element name="ATTRIBUTE-VALUE-XML-DATA-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-HIERARCHY-ROOTS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-HIERARCHY-ROOT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-RELATIONS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-RELATION-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-OBJECTS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-OBJECT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-GROUPS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-GROUP-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="ACCESS-MODE-ENUM">
<xsd:restriction base="RIF:STRING">
<xsd:enumeration value="EDIT"/>
<xsd:enumeration value="DELETE"/>
<xsd:enumeration value="CREATE"/>
</xsd:restriction>
</xsd:simpleType>
<!-- Element generated from meta class: ExchangeFile::AttributeDefinitionComplex -->
<xsd:element name="ATTRIBUTE-DEFINITION-COMPLEX">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeDefinitionEnumeration -->
<xsd:element name="ATTRIBUTE-DEFINITION-ENUMERATION">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="MULTI-VALUED" type="RIF:BOOLEAN" minOccurs="0"/>
<xsd:element name="TYPE" minOccurs="0">

```

```
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="DATATYPE-DEFINITION-ENUMERATION-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="DEFAULT-VALUE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ATTRIBUTE-VALUE-ENUMERATION"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeDefinitionSimple -->
<xsd:element name="ATTRIBUTE-DEFINITION-SIMPLE">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="DEFAULT-VALUE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element ref="RIF:ATTRIBUTE-VALUE-SIMPLE"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="TYPE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="DATATYPE-DEFINITION-BOOLEAN-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-INTEGER-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-REAL-REF" type="RIF:REF"/>
<xsd:element name="DATATYPE-DEFINITION-STRING-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeValueEmbeddedDocument -->
<xsd:element name="ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="DEFINITION" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="DATATYPE-DEFINITION-DOCUMENT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="XHTML-CONTENT" minOccurs="0">
<xsd:complexType>
<xsd:choice>
<xsd:any namespace="http://automotive-his.de/schema/rif-xhtml" processContents="strict"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeValueEmbeddedFile -->
<xsd:element name="ATTRIBUTE-VALUE-EMBEDDED-FILE">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 48 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="BINARY-CONTENT" minOccurs="0"/>
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element ref="RIF:BINARY-CONTENT"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="DEFINITION" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="DATATYPE-DEFINITION-BINARY-FILE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeValueEnumeration -->
<xsd:element name="ATTRIBUTE-VALUE-ENUMERATION">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="DEFINITION" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="ATTRIBUTE-DEFINITION-ENUMERATION-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="VALUES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="ENUM-VALUE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeValueFileReference -->
<xsd:element name="ATTRIBUTE-VALUE-FILE-REFERENCE">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="PATH-TO-FILE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="DEFINITION" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="DATATYPE-DEFINITION-BINARY-FILE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeValueSimple -->
<xsd:element name="ATTRIBUTE-VALUE-SIMPLE">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="THE-VALUE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="DEFINITION" minOccurs="0">

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 49 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="ATTRIBUTE-DEFINITION-SIMPLE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::AttributeValueXmlData -->
<xsd:element name="ATTRIBUTE-VALUE-XML-DATA">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="XML-CONTENT" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element ref="RIF:XML-CONTENT"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="DEFINITION" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="DATATYPE-DEFINITION-XML-DATA-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::BinaryContent -->
<xsd:element name="BINARY-CONTENT">
<xsd:complexType/>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionBinaryFile -->
<xsd:element name="DATATYPE-DEFINITION-BINARY-FILE">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="EMBEDDED" type="RIF:BOOLEAN" minOccurs="0"/>
<xsd:element name="FORMAT-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="MIME-TYPE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="FILENAME-SUFFIX" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="APPLICATION" type="RIF:STRING" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionBoolean -->
<xsd:element name="DATATYPE-DEFINITION-BOOLEAN">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionDocument -->
<xsd:element name="DATATYPE-DEFINITION-DOCUMENT">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="EMBEDDED" type="RIF:BOOLEAN" minOccurs="0"/>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 50 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xsd:element name="FORMAT" type="RIF:FORMAT-ENUM" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<xsd:simpleType name="FORMAT-ENUM">
<xsd:restriction base="RIF:STRING">
<xsd:enumeration value="XHTML"/>
<xsd:enumeration value="RTF"/>
</xsd:restriction>
</xsd:simpleType>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionEnumeration -->
<xsd:element name="DATATYPE-DEFINITION-ENUMERATION">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="ENUM-VALUES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ENUM-VALUE"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionInteger -->
<xsd:element name="DATATYPE-DEFINITION-INTEGER">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="MAX" type="RIF:INTEGER" minOccurs="0"/>
<xsd:element name="MIN" type="RIF:INTEGER" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionReal -->
<xsd:element name="DATATYPE-DEFINITION-REAL">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="MAX" type="RIF:FLOAT" minOccurs="0"/>
<xsd:element name="MIN" type="RIF:FLOAT" minOccurs="0"/>
<xsd:element name="ACCURACY" type="RIF:INTEGER" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionString -->
<xsd:element name="DATATYPE-DEFINITION-STRING">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="MAX-LENGTH" type="RIF:INTEGER" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::DatatypeDefinitionXmlData -->
<xsd:element name="DATATYPE-DEFINITION-XML-DATA">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>

```

```

<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="EMBEDDED" type="RIF:BOOLEAN" minOccurs="0"/>
<xsd:element name="NAME-SPACE-URI" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="SCHEMA-LOCATION" type="RIF:STRING" minOccurs="0"/>
<xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::EmbeddedValues -->
<xsd:element name="EMBEDDED-VALUES">
<xsd:complexType>
<xsd:all>
<xsd:element name="KEY" type="RIF:INTEGER" minOccurs="0"/>
<xsd:element name="OTHER_CONTENT" type="RIF:FLOAT" minOccurs="0"/>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::EnumValue -->
<xsd:element name="ENUM-VALUE">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="EMBEDDED-VALUESS" minOccurs="0"/>
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element ref="RIF:EMBEDDED-VALUES"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::RelationGroup -->
<xsd:element name="RELATION-GROUP">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="RELATION-TYPE" minOccurs="0"/>
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-TYPE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="TARGET-GROUP" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-GROUP-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-RELATIONS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-RELATION-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::RIF -->
<xsd:element name="RIF">
<xsd:complexType>
<xsd:all>
<xsd:element name="TITLE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="CHECKSUM" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="COUNTRY-CODE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="CREATION-TIME" type="RIF:STRING" minOccurs="0"/>

```

```
<xsd:element name="COMMENT" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="VERSION" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="AUTHOR" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="SOURCE-TOOL-ID" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="DATATYPES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:DATATYPE-DEFINITION-BINARY-FILE"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-BOOLEAN"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-DOCUMENT"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-ENUMERATION"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-INTEGER"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-REAL"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-STRING"/>
<xsd:element ref="RIF:DATATYPE-DEFINITION-XML-DATA"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="ACCESS-POLICIES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ACCESS-POLICY"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-OBJECTS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-OBJECT"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-GROUPS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-GROUP"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-TYPES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-TYPE"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-HIERARCHY-ROOTS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-HIERARCHY-ROOT"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-RELATIONS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-RELATION"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::SpecGroup -->
<xsd:element name="SPEC-GROUP">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="TYPE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 53 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xsd:element name="SPEC-TYPE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="VALUES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-FILE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-ENUMERATION"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-FILE-REFERENCE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-SIMPLE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-XML-DATA"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SPEC-OBJECTS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element name="SPEC-OBJECT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="RELATION-GROUPS" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:RELATION-GROUP"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::SpecHierarchy -->
<xsd:element name="SPEC-HIERARCHY">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="SPEC-OBJECT" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-OBJECT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="CHILDREN" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-HIERARCHY"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::SpecHierarchyRoot -->
<xsd:element name="SPEC-HIERARCHY-ROOT">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="TYPE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-TYPE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="VALUES" minOccurs="0">

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 54 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-FILE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-ENUMERATION"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-FILE-REFERENCE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-SIMPLE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-XML-DATA"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="CHILDREN" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:SPEC-HIERARCHY"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::SpecObject -->
<xsd:element name="SPEC-OBJECT">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="TYPE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-TYPE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="VALUES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-FILE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-ENUMERATION"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-FILE-REFERENCE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-SIMPLE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-XML-DATA"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::SpecRelation -->
<xsd:element name="SPEC-RELATION">
<xsd:complexType>
<xsd:all>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="IDENTIFIER" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="TYPE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-TYPE-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="VALUES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-EMBEDDED-FILE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-ENUMERATION"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-FILE-REFERENCE"/>
<xsd:element ref="RIF:ATTRIBUTE-VALUE-SIMPLE"/>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 55 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xsd:element ref="RIF:ATTRIBUTE-VALUE-XML-DATA"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="SOURCE" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-OBJECT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
<xsd:element name="TARGET" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0">
<xsd:element name="SPEC-OBJECT-REF" type="RIF:REF"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::SpecType -->
<xsd:element name="SPEC-TYPE">
<xsd:complexType>
<xsd:all>
<xsd:element name="IDENTIFIER" type="RIF:STRING"/>
<xsd:element name="DESC" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LONG-NAME" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="LAST-CHANGE" type="RIF:STRING" minOccurs="0"/>
<xsd:element name="SPEC-ATTRIBUTES" minOccurs="0">
<xsd:complexType>
<xsd:choice minOccurs="0" maxOccurs="unbounded">
<xsd:element ref="RIF:ATTRIBUTE-DEFINITION-COMPLEX"/>
<xsd:element ref="RIF:ATTRIBUTE-DEFINITION-ENUMERATION"/>
<xsd:element ref="RIF:ATTRIBUTE-DEFINITION-SIMPLE"/>
</xsd:choice>
</xsd:complexType>
</xsd:element>
</xsd:all>
<xsd:attribute ref="RIF:ID"/>
</xsd:complexType>
</xsd:element>
<!-- Element generated from meta class: ExchangeFile::XmlContent -->
<xsd:element name="XML-CONTENT">
<xsd:complexType/>
</xsd:element>
<!-- Predefined element and attribute types -->
<xsd:simpleType name="STRING">
<xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:simpleType name="FLOAT">
<xsd:restriction base="xsd:float"/>
</xsd:simpleType>
<xsd:simpleType name="INTEGER">
<xsd:restriction base="xsd:integer"/>
</xsd:simpleType>
<xsd:simpleType name="BOOLEAN">
<xsd:restriction base="RIF:STRING">
<xsd:enumeration value="TRUE"/>
<xsd:enumeration value="FALSE"/>
</xsd:restriction>
</xsd:simpleType>
<xsd:simpleType name="REF">
<xsd:restriction base="xsd:string"/>
</xsd:simpleType>
<xsd:attribute name="ID" type="xsd:string"/>
</xsd:schema>

```

3.5 RIF integrated XHTML Schema

```
<?xml version="1.0" encoding="UTF-8"?>
```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 56 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns="http://automotive-his.de/schema/rif-xhtml"
targetNamespace="http://automotive-his.de/schema/rif-xhtml" elementFormDefault="qualified" version="1.0" xml:lang="en">
<xs:annotation>
<xs:documentation>schema is a subset of the official W3C XHTML Schema
</xs:documentation>
</xs:annotation>
<xs:import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="http://www.w3.org/2001/xml.xsd"/>
<xs:annotation>
<xs:documentation>
===== Character mnemonic entities =====
entity sets are identified by the PUBLIC and SYSTEM identifiers:
"-//W3C//ENTITIES Latin 1 for XHTML//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-lat1.ent"
"-//W3C//ENTITIES Special for XHTML//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-special.ent"
"-//W3C//ENTITIES Symbols for XHTML//EN" http://www.w3.org/TR/xhtml1/DTD/xhtml1-symbol.ent"
</xs:documentation>
</xs:annotation>
<xs:annotation>
<xs:documentation>
===== Imported Names =====
</xs:documentation>
</xs:annotation>
<xs:simpleType name="ContentType">
<xs:annotation>
<xs:documentation>type, as per [RFC2045]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="ContentTypes">
<xs:annotation>
<xs:documentation>separated list of media types, as per [RFC2045]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="Charset">
<xs:annotation>
<xs:documentation>character encoding, as per [RFC2045]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="Charsets">
<xs:annotation>
<xs:documentation>space separated list of character encodings, as per [RFC2045]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="LanguageCode">
<xs:annotation>
<xs:documentation>language code, as per [RFC3066]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:language"/>
</xs:simpleType>
<xs:simpleType name="Character">
<xs:annotation>
<xs:documentation>single character, as per section 2.2 of [XML]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
<xs:length value="1" fixed="true"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="Number">
<xs:annotation>
<xs:documentation>or more digits
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:nonNegativeInteger">
<xs:pattern value="[0-9]"/>
</xs:restriction>
</xs:simpleType>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 57 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xs:simpleType name="tabindexNumber">
<xs:annotation>
<xs:documentation>attribute specifies the position of the current elementthe tabbing order for the current document. This value must
benumber between 0 and 32767. User agents should ignore leading zeros.
</xs:documentation>
</xs:annotation>
<xs:restriction base="Number">
<xs:minInclusive value="0"/>
<xs:maxInclusive value="32767"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="LinkTypes">
<xs:annotation>
<xs:documentation>separated list of link types
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:NMTOKENS"/>
</xs:simpleType>
<xs:simpleType name="URI">
<xs:annotation>
<xs:documentation>Uniform Resource Identifier, see [RFC2396]
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:anyURI"/>
</xs:simpleType>
<xs:simpleType name="UriList">
<xs:annotation>
<xs:documentation>space separated list of Uniform Resource Identifiers
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="Datetime">
<xs:annotation>
<xs:documentation>and time information. ISO date format
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:dateTime"/>
</xs:simpleType>
<xs:simpleType name="Script">
<xs:annotation>
<xs:documentation>expression
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="StyleSheet">
<xs:annotation>
<xs:documentation>sheet data
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="Text">
<xs:annotation>
<xs:documentation>for titles etc.
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string"/>
</xs:simpleType>
<xs:simpleType name="Length">
<xs:annotation>
<xs:documentation>for pixels or nn% for percentage length
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="Pixels">
<xs:annotation>
<xs:documentation>representing length in pixels
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:nonNegativeInteger"/>

```

```

</xs:simpleType>
<xs:annotation>
<xs:documentation>are used for image maps
</xs:documentation>
</xs:annotation>
<xs:simpleType name="Shape">
<xs:restriction base="xs:token">
<xs:enumeration value="rect"/>
<xs:enumeration value="circle"/>
<xs:enumeration value="poly"/>
<xs:enumeration value="default"/>
</xs:restriction>
</xs:simpleType>
<xs:simpleType name="Coords">
<xs:annotation>
<xs:documentation>separated list of lengths
</xs:documentation>
</xs:annotation>
<xs:restriction base="xs:string">
</xs:restriction>
</xs:simpleType>
<xs:annotation>
<xs:documentation>
===== Generic Attributes =====
</xs:documentation>
</xs:annotation>
<xs:attributeGroup name="coreattrs">
<xs:annotation>
<xs:documentation>attributes common to most elements    document-wide unique id    space separated list of classes    associated style info
advisory title/amplification
</xs:documentation>
</xs:annotation>
<xs:attribute name="id" type="xs:ID"/>
<xs:attribute name="class" type="xs:NMTOKENS"/>
<xs:attribute name="style" type="StyleSheet"/>
<xs:attribute name="title" type="Text"/>
</xs:attributeGroup>

<xs:attributeGroup name="attrs">
<xs:attributeGroup ref="coreattrs"/>
</xs:attributeGroup>
<xs:annotation>
<xs:documentation>
===== Text Elements =====
</xs:documentation>
</xs:annotation>
<xs:group name="special.pre">
<xs:choice>
<xs:element ref="br"/>
<xs:element ref="span"/>
</xs:choice>
</xs:group>
<xs:group name="special">
<xs:choice>
<xs:group ref="special.pre"/>
<xs:element ref="object"/>
<xs:element ref="img"/>
</xs:choice>
</xs:group>
<xs:group name="fontstyle">
<xs:choice>
<xs:element ref="tt"/>
<xs:element ref="i"/>
<xs:element ref="b"/>
<xs:element ref="big"/>
<xs:element ref="small"/>
</xs:choice>
</xs:group>
<xs:group name="phrase">
<xs:choice>
<xs:element ref="em"/>
<xs:element ref="strong"/>
<xs:element ref="dfn"/>
<xs:element ref="code"/>

```

```

<xs:element ref="q"/>
<xs:element ref="samp"/>
<xs:element ref="kbd"/>
<xs:element ref="var"/>
<xs:element ref="cite"/>
<xs:element ref="abbr"/>
<xs:element ref="acronym"/>
<xs:element ref="sub"/>
<xs:element ref="sup"/>
</xs:choice>
</xs:group>
<xs:group name="inline.forms">
<xs:choice/>
</xs:group>
<xs:group name="misc.inline">
<xs:choice>
<xs:element ref="ins"/>
<xs:element ref="del"/>
</xs:choice>
</xs:group>
<xs:group name="misc">
<xs:annotation>
<xs:documentation>can only occur at block level
</xs:documentation>
</xs:annotation>
<xs:choice>
<xs:group ref="misc.inline"/>
</xs:choice>
</xs:group>
<xs:group name="inline">
<xs:choice>
<xs:element ref="a"/>
<xs:group ref="special"/>
<xs:group ref="fontstyle"/>
<xs:group ref="phrase"/>
<xs:group ref="inline.forms"/>
</xs:choice>
</xs:group>
<xs:complexType name="Inline" mixed="true">
<xs:annotation>
<xs:documentation>
"Inline" covers inline or "text-level" elements
</xs:documentation>
</xs:annotation>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:group ref="inline"/>
<xs:group ref="misc.inline"/>
</xs:choice>
</xs:complexType>
<xs:annotation>
<xs:documentation>
===== Block level elements =====
</xs:documentation>
</xs:annotation>
<xs:group name="heading">
<xs:choice>
<xs:element ref="h1"/>
<xs:element ref="h2"/>
<xs:element ref="h3"/>
<xs:element ref="h4"/>
<xs:element ref="h5"/>
<xs:element ref="h6"/>
</xs:choice>
</xs:group>
<xs:group name="lists">
<xs:choice>
<xs:element ref="ul"/>
<xs:element ref="ol"/>
<xs:element ref="dl"/>
</xs:choice>
</xs:group>
<xs:group name="blocktext">
<xs:choice>
<xs:element ref="pre"/>

```

```

<xs:element ref="hr"/>
<xs:element ref="blockquote"/>
<xs:element ref="address"/>
</xs:choice>
</xs:group>
<xs:group name="block">
<xs:choice>
<xs:element ref="p"/>
<xs:group ref="heading"/>
<xs:element ref="div"/>
<xs:group ref="lists"/>
<xs:group ref="blocktext"/>
</xs:choice>
</xs:group>
<xs:complexType name="Block">
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:group ref="block"/>
<xs:group ref="misc"/>
</xs:choice>
</xs:complexType>
<xs:complexType name="Flow" mixed="true">
<xs:annotation>
<xs:documentation>
"Flow" mixes block and inline and is used for list items etc.
</xs:documentation>
</xs:annotation>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:group ref="block"/>
<xs:group ref="inline"/>
<xs:group ref="misc"/>
</xs:choice>
</xs:complexType>
<xs:annotation>
<xs:documentation>
===== Content models for exclusions =====
</xs:documentation>
</xs:annotation>
<xs:complexType name="a.content" mixed="true">
<xs:annotation>
<xs:documentation>elements use "Inline" excluding a
</xs:documentation>
</xs:annotation>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:group ref="special"/>
<xs:group ref="fontstyle"/>
<xs:group ref="phrase"/>
<xs:group ref="inline.forms"/>
<xs:group ref="misc.inline"/>
</xs:choice>
</xs:complexType>
<xs:complexType name="pre.content" mixed="true">
<xs:annotation>
<xs:documentation>uses "Inline" excluding big, small, sup or sup
</xs:documentation>
</xs:annotation>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:element ref="a"/>
<xs:group ref="fontstyle"/>
<xs:group ref="phrase"/>
<xs:group ref="special.pre"/>
<xs:group ref="misc.inline"/>
<xs:group ref="inline.forms"/>
</xs:choice>
</xs:complexType>
<xs:complexType name="form.content">
<xs:annotation>
<xs:documentation>uses "Block" excluding form
</xs:documentation>
</xs:annotation>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:group ref="block"/>
<xs:group ref="misc"/>
</xs:choice>
</xs:complexType>

```

```

<xs:complexType name="button.content" mixed="true">
<xs:annotation>
<xs:documentation>uses "Flow" but excludes a, form and form controls
</xs:documentation>
</xs:annotation>
<xs:choice minOccurs="0" maxOccurs="unbounded">
<xs:element ref="p"/>
<xs:group ref="heading"/>
<xs:element ref="div"/>
<xs:group ref="lists"/>
<xs:group ref="blocktext"/>
<xs:group ref="special"/>
<xs:group ref="fontstyle"/>
<xs:group ref="phrase"/>
<xs:group ref="misc"/>
</xs:choice>
</xs:complexType>
<xs:annotation>
<xs:documentation>
===== Document Structure =====
</xs:documentation>
</xs:annotation>
<xs:element name="html">
<xs:complexType>
<xs:sequence>
<xs:element ref="body"/>
</xs:sequence>
<xs:attribute name="id" type="xs:ID"/>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Document Body =====
</xs:documentation>
</xs:annotation>
<xs:element name="body">
<xs:complexType>
<xs:complexContent>
<xs:extension base="Block">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="div">
<xs:annotation>
<xs:documentation>language/style container
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Flow">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Paragraphs =====
</xs:documentation>
</xs:annotation>
<xs:element name="p">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Headings =====

```

are six levels of headings from h1 (the most important)h6 (the least important).

```

</xs:documentation>
</xs:annotation>
<xs:element name="h1">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="h2">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="h3">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="h4">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="h5">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="h6">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>

```

===== Lists =====

```

</xs:documentation>
</xs:annotation>
<xs:element name="ul">
<xs:annotation>
<xs:documentation>list
</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element ref="li" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attributeGroup ref="attrs"/>
</xs:complexType>
</xs:element>

```

```

<xs:element name="ol">
<xs:annotation>
<xs:documentation>(numbered) list
</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element ref="li" maxOccurs="unbounded"/>
</xs:sequence>
<xs:attributeGroup ref="attrs"/>
</xs:complexType>
</xs:element>
<xs:element name="li">
<xs:annotation>
<xs:documentation>item
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Flow">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>lists - dt for term, dd for its definition
</xs:documentation>
</xs:annotation>
<xs:element name="dl">
<xs:complexType>
<xs:choice maxOccurs="unbounded">
<xs:element ref="dt"/>
<xs:element ref="dd"/>
</xs:choice>
<xs:attributeGroup ref="attrs"/>
</xs:complexType>
</xs:element>
<xs:element name="dt">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="dd">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Flow">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Address =====
</xs:documentation>
</xs:annotation>
<xs:element name="address">
<xs:annotation>
<xs:documentation>on author
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>

```

```

<xs:annotation>
<xs:documentation>
===== Horizontal Rule =====
</xs:documentation>
</xs:annotation>
<xs:element name="hr">
<xs:complexType>
<xs:attributeGroup ref="attrs"/>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Preformatted Text =====
</xs:documentation>
</xs:annotation>
<xs:element name="pre">
<xs:annotation>
<xs:documentation>is "Inline" excluding "img|object|big|small|sub|sup"
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="pre.content">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Block-like Quotes =====
</xs:documentation>
</xs:annotation>
<xs:element name="blockquote">
<xs:complexType>
<xs:complexContent>
<xs:extension base="Block">
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="cite" type="URI"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Inserted/Deleted Text =====
/del are allowed in block and inline content, but itsto include block content within an ins elementin inline content.
</xs:documentation>
</xs:annotation>
<xs:element name="ins">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Flow">
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="cite" type="URI"/>
<xs:attribute name="datetime" type="Datetime"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="del">
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Flow">
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="cite" type="URI"/>
<xs:attribute name="datetime" type="Datetime"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== The Anchor Element =====

```

```

</xs:documentation>
</xs:annotation>
<xs:element name="a">
<xs:annotation>
<xs:documentation>is "Inline" except that anchors shouldn't be nested
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="a.content">
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="charset" type="Charset"/>
<xs:attribute name="type" type="ContentType"/>
<xs:attribute name="name" type="xs:NMTOKEN"/>
<xs:attribute name="href" type="URI"/>
<xs:attribute name="hreflang" type="LanguageCode"/>
<xs:attribute name="rel" type="LinkTypes"/>
<xs:attribute name="rev" type="LinkTypes"/>
<xs:attribute name="shape" type="Shape" default="rect"/>
<xs:attribute name="coords" type="Coords"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Inline Elements =====
</xs:documentation>
</xs:annotation>
<xs:element name="span">
<xs:annotation>
<xs:documentation>language/style container
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="br">
<xs:annotation>
<xs:documentation>line break
</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:attributeGroup ref="coreattrs"/>
</xs:complexType>
</xs:element>
<xs:element name="em">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="strong">
<xs:annotation>
<xs:documentation>emphasis
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>

```

```
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="dfn">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="code">
<xs:annotation>
<xs:documentation>code
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="samp">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="kbd">
<xs:annotation>
<xs:documentation>user would type
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="var">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="cite">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 67 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="abbr">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="acronym">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="q">
<xs:annotation>
<xs:documentation>quote
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="cite" type="URI"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="sub">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="sup">
<xs:annotation>
<xs:documentation>
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="tt">

```

```

<xs:annotation>
<xs:documentation>pitch font
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="i">
<xs:annotation>
<xs:documentation>font
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="b">
<xs:annotation>
<xs:documentation>font
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="big">
<xs:annotation>
<xs:documentation>font
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:element name="small">
<xs:annotation>
<xs:documentation>font
</xs:documentation>
</xs:annotation>
<xs:complexType mixed="true">
<xs:complexContent mixed="true">
<xs:extension base="Inline">
<xs:attributeGroup ref="attrs"/>
</xs:extension>
</xs:complexContent>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>

```

===== Object =====

is used to embed objects as part of HTML pages.elements should precede other content. Parametersalso be expressed as attribute/value pairs on theelement itself when brevity is desired.

```

</xs:documentation>
</xs:annotation>
<xs:element name="object">
<xs:complexType mixed="true">
<xs:choice minOccurs="0" maxOccurs="unbounded">

```

```

<xs:element ref="param"/>
<xs:group ref="block"/>
<xs:group ref="inline"/>
<xs:group ref="misc"/>
</xs:choice>
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="declare"/>
<xs:simpleType>
<xs:restriction base="xs:token">
<xs:enumeration value="declare"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
<xs:attribute name="classid" type="URI"/>
<xs:attribute name="codebase" type="URI"/>
<xs:attribute name="data" type="URI"/>
<xs:attribute name="type" type="ContentType"/>
<xs:attribute name="codetype" type="ContentType"/>
<xs:attribute name="archive" type="UriList"/>
<xs:attribute name="standby" type="Text"/>
<xs:attribute name="height" type="Length"/>
<xs:attribute name="width" type="Length"/>
<xs:attribute name="usemap" type="URI"/>
<xs:attribute name="name" type="xs:NMTOKEN"/>
<xs:attribute name="tabindex" type="tabindexNumber"/>
</xs:complexType>
</xs:element>
<xs:element name="param">
<xs:annotation>
<xs:documentation>is used to supply a named property value.XML it would seem natural to follow RDF and support ansyntax where the
param elements are replacedattribute value pairs on the object start tag.
</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:attribute name="id" type="xs:ID"/>
<xs:attribute name="name"/>
<xs:attribute name="value"/>
<xs:attribute name="valuetype" default="data">
<xs:simpleType>
<xs:restriction base="xs:token">
<xs:enumeration value="data"/>
<xs:enumeration value="ref"/>
<xs:enumeration value="object"/>
</xs:restriction>
</xs:simpleType>
</xs:attribute>
<xs:attribute name="type" type="ContentType"/>
</xs:complexType>
</xs:element>
<xs:annotation>
<xs:documentation>
===== Images =====

```

avoid accessibility problems for people who aren't to see the image, you should provide a textusing the alt and longdesc attributes.addition, avoid the use of server-side image maps.that in this DTD there is no name attribute. That only available in the transitional and frameset DTD.

```

</xs:documentation>
</xs:annotation>
<xs:element name="img">
<xs:complexType>
<xs:attributeGroup ref="attrs"/>
<xs:attribute name="src" type="URI" use="required"/>
<xs:attribute name="alt" type="Text" use="required"/>
<xs:attribute name="longdesc" type="URI"/>
<xs:attribute name="height" type="Length"/>
<xs:attribute name="width" type="Length"/>
<xs:attribute name="usemap" type="URI"/>
<xs:annotation>
<xs:documentation>points to a map element which may be in this documentan external document, although the latter is not widely supported
</xs:documentation>
</xs:annotation>
</xs:attribute>
<xs:attribute name="ismap">
<xs:simpleType>
<xs:restriction base="xs:token">
<xs:enumeration value="ismap"/>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 70 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

</xs:restriction>
</xs:simpleType>
</xs:attribute>
</xs:complexType>
</xs:element>
</xs:schema>

Appendix C: Mapping Table

This is a fragmentary overview how to map certain RIF elements to some tool elements. The purpose is to help understanding how to use certain RIF classes in exemplary tool context.

TOOL Mapping RIF				
Structure elements				
Model element (classes) im RIF	DOORS/ERS ²	IRqA ³	CaliberRM ⁴	Comments / Rules
Application-Structur				Strukturelemente, die sich nicht 1:1 vergleichen lassen (z.B. verschiedene Toolfamilien), werden über User-Defined-Attributes im jeweils anderen Tool abgebildet. Strukturelemente die innerhalb einer gleichen Toolfamilie ausgetauscht werden, können über speziell Schlüsselworte als eine Erweiterung des Klassen-Bezeichners erfolgen. Die Identifikation des Senders wird durch die TOOL-ID (als Attribut im Root Element) ermöglicht.
AttributeDefinitionEnumeration	Attribute Definition: Enumeration	Attribute Definition: Enumeration Management Facets	n/a	Definition von Aufzählungsattributen
AttributeDefinitionStructured	n/a	n/a	n/a	Superklasse zur Vererbung
AttributeValueDirectlyCoded	Attribute Value [für String, Real, Integer, Datum, etc.]	Attribute Value [für Text, Real, Integer, Datum, Boolean, Listing]	n/a	Werte von Attributen die vom Benutzer programmiert wurden
AttributeValueEmbeddedDocument	Attribute Value [für Text]	Attribute Value [für Text]	n/a	Werte von Attributen die eingebettete Dokumente enthalten
AttributeValueEnumeration	Attribute Value [für Enumerations]	Attribute Value [für Enumerations und Management Facets]	n/a	Werte von Aufzählungsattributen
AttributeValueNameSpace	n/a	n/a	n/a	Superklasse zur Vererbung
DatatypeDefinitionBinary	n/a	n/a	n/a	Superklasse zur Vererbung
DatatypeDefinitionComplex	n/a	n/a	n/a	Superklasse zur Vererbung

² DOORS/ERS is a trademark of Telelogic AB

³ IRqA is a trademark of TCP Sistemas e Ingeniería S.L. Spain

⁴ CaliberRM is a trademark of Borland Software Corporation

DatatypeDefinitionEnumeration	Type: Enumeration	Type: Enumerate oder Management Facets	n/a	Datentyp Definition von Aufzählungstypen
DatatypeDefinitionRTF	Type: Text	Type: Text	Type: Text (sofern unterstützt)	Datentyp Definition von RTF
DatatypeDefinitionStructured	n/a	n/a	n/a	Datentyp Definition von strukturierten Elementen
DatatypeDefinitionXHTML	Type: Text	Type: Text	Type: Text (sofern unterstützt)	Datentyp Definition für XHTML
EmbeddedValues	Key Value und Color [in der Typ-Definition von Enumerations]	Key Value und Color [in der Typ-Definition von Enumerations]	n/a	Zusammengesetzte Aufzählungstypen z.B. bestehend aus den Artefakten Schlüssel –Wert und -Farbe
RelationGroup	Linkset Pairing	Motiv	n/a	Gruppierung von Beziehungen in einem übergeordneten Kontext
SpecHierarchyRoot	Formal Modul	Block	n/a	Wurzelement einer SpecHierarchy
AccessPolicy	Access Rights	Rights management with Groups and Access Partitions	n/a	Zugriffsrechte für Austauschpartner
DatatypeDefinitionBoolean	Type: Boolean	Type: Boolean	Type: Boolean (sofern unterstützt)	Datentyp Definition für boolesche Attribute
<i>DatatypeDefinition</i>	Type	Type	Type (sofern unterstützt)	Datentyp Definition
DatatypeDefinitionDocument	Type: Text	Type: Text	Type: Text (sofern unterstützt)	Datentyp Definition für Dokumente
EnumValue	Sowohl als Attribute Value als auch innerhalb der Typ-Definition in Enumerations	Sowohl als Attribute Value als auch innerhalb der Typ-Definition in Enumerations	Sowohl als Attribute Value als auch innerhalb der Typ-Definition in Enumerations	Konkrete zugewiesene Werte von Aufzählungen
Identifiable	vor allem Tool-Ids	vor allem Tool-Ids	vor allem Tool-Ids	Unique identifizierbares Element in RIF
DatatypeDefinitionInteger	Type: Integer	Type: Integer	Type: Integer (sofern unterstützt)	Datentyp Definition für integer Typen
DatatypeDefinitionReal	Type: Real	Type: Real	Type: Real	Datentyp Definition für real Typen
DatatypeDefinitionString	Type: String	Type: Text	Type: Text	Datentyp Definition für Zeichenketten Typen
<i>DatatypeDefinitionSimple</i>	Type	Type	Type	Datentyp Definition für einfache Typen
SpecElementWithUserDefinedAttributes	n/a	n/a	n/a	Superklasse zur Vererbung

© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.

AttributeDefinition	Attribute Definitions	New Attribute	n/a	Definition der Merkmale von benutzerdefinierten Attributen
AttributeDefinitionComplex	n/a	n/a	n/a	Superklasse zur Vererbung
AttributeDefinitionSimple	Attribute Definitions	New Attribute	n/a	Attribut Definition für einfache Typen
AttributeValue	Attribute Values	Attribute Values	n/a	Konkret zugewiesener Wert eines Attributs
AttributeValueComplex	n/a	n/a	n/a	Superklasse zur Vererbung
SpecGroup	Formal Modul	Block	n/a	Anordnung von zusammengehörigen Requirements
SpecHierarchy	Formal Modul	Hierarchy	n/a	Damit werden Baumansichten in Form von Parent-Child Beziehungen abgebildet.
SpecObject	Object	Requirements, Test Scenarios, Entities, concepts, services, actors, events, scenarios, processes	n/a	Beispiel: Textuelle, graphische strukturelle oder sämtliche andere Arten von Requirements
SpecRelation	Link	relationship by Motif	n/a	Darstellung von Verknüpfungs- bzw. Link Beziehungen
SpecType	Modul	Blocktype with attributes, Management Facets, Domain Facet, terms and Object Type	Requirements Type	Typisierung von gleichartigen SpecObjects
TimeStamp	Element der Austauschdatei	Element der Austauschdatei	Element der Austauschdatei	Ist der Identifier der Datei und wird von den RM-Tools beim Erzeugen der Datei generiert und in der RIF Datei transportiert.
RIF	Element der Austauschdatei	Element der Austauschdatei	Element der Austauschdatei	Dieses ROOT-Element des Austauschformat muss in jeder Austauschdatei vorhanden sein.
RootClassFromOtherNameSpace	Element der Austauschdatei	Element der Austauschdatei	Element der Austauschdatei	Neben den Schemata Definitionen werden weitere Namespaces benötigt. Daher muss dieses Element mit übertragen werden
SpecGroup	Formale Module	Blocks	Requirement Types	Formal Module, Blocks und Requirement Types sind Gruppierungen von Requirements
SpecHierarchy	Kapitel-Gliederung innerhalb eines Formal Module	Hierarchy (Parent-Child-Links)	Eltern-Kind-Beziehung zwischen Anforderungen	Betrachtet man ein komplettes Spezifikationsdokument im Sinne eines Lasten oder Pflichtenheftes, dann ist i.d.R eine Kapitelstruktur erkennbar. Sofern diese in einem Tool verwaltet wird, entspricht dies genau einer Hierarchie. In Ausnahmefällen kann es innerhalb dieser Hierarchie zu einer weiteren Schachtelung in eingebettete Hierarchien kommen.

© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.

SpecHierarchy (mit Auszeichnung)	DOORS Table	DOORS_TABLE_SpecHierarchy	DOORS_TABLE_SpecHierarchy	Befinden sich innerhalb eines SpecObjects z.B. Tables, deren Zellen wiederum eigene SpecObjects mit eigenen Attributen bilden, dann kann diese Sub-Hierarchy auch als eine spezielle Form einer SpecHierarchy mit ausgezeichnetem SpecType dargestellt werden. Diese Hierarchy muss dann aber per Konvention besonders gekennzeichnet werden (z.B. "DOORS_TABLE"). Grundsätzlich kann der sendende Import-Filter seine eigenen Tables auch ohne spezielle Kennzeichnung erkennen, wenn er sich beim vorhergehenden Export den die hierarchischen Strukturen, Substrukturen und sämtliche Struktur- und SpecObject spezifische System-Attribute gespeichert hat. Ausnahme wenn die übergeordnete Hierarchy nur aus einem Element besteht, welches eine spezielle Tabelle beinhaltet. Die strukturelle Entscheidung, was die Table-Hierarchy ist und was die übergeordnete Gliederung ist, muss der Export-Filter festhalten und für den Import reproduzieren können, Gleiches gilt für die eigenen Table-spezifische System-Attribute. Die Partnerschnittstelle kann die Tables die selbst Hierarchien sind wie folgt behandeln: 1.) als weitere Hierarchy/Gliederung abbilden, wenn mehrere Hierarchien parallel beherrscht werden. 2.) Über Relationen zwischen den Tabellen-Elementen abbilden
SpecType (RelationGroup, Type)	Link Module (LM)	Motive	n/a	Die Unterscheidung zwischen einem Link Container (z.B. LM) und einem SpecObject Container (z.B. FM) über den SpecType kann wie folgt getroffen weil 1.) Objekte der Klasse "SpecRelation" diesen SpecType assoziieren und 2.) RelationGroup Objekte auf diesen SpecType assoziieren. ACHTUNG: Probleme könnten bei leeren Link Container bestehen. Ausweg wäre ein Meta-Attribut, oder zwei Sub-Klassen von SpecType, nämlich eine für Anforderungs-Typen und eine für Relations-Typen.
Relation Group (specRelations)	LinkSet	Könnte in IRqA™ über die Traceability-Matrix abgebildet werden	n/a	RelationGroups im Sinne von Gruppierung von verschiedenen Verknüpfungen bzw. Verknüpfungstypen, die SpecObjects innerhalb verschiedener Pakete und Strukturen miteinander nach definierten Regeln verbinden. Der Austausch sollte kein Problem sein, wenn es von Tools, die keine LinkSets kennen, ignoriert wird. Beim Re-Import in Tools die mit LinkSets arbeiten sollte die Zuordnung auf die Link-Module und Link-Sets sich aus den beim vorangegangenen Export diesen Tools zuvor in der dortigen DB vorhandenen Informationen ergeben. Irritationen kann es nur dann geben, wenn sich beim Austausch mit einem Partner der keine LinkSets kennt zwischen Export und Re-Import in der DB Änderungen der Default-LinkSet Definitionen ergeben, dann kommen die vom Partner erzeugten Links nämlich in das LinkSet nach neuer Definition.
SpecObject	Object (SpecGroup via FM)	Requirements, Test Scenarios, Entities, concepts, services, actors, events, scenarios, processes	Requirements	Beispiel: Textuelle, graphische strukturelle oder sämtliche andere Arten von Requirements
SpecRelation	Link	Relationship	Relationship (simple: one kind)	Gerichtete Verknüpfungen (Links) zwischen Source und Target
AttributeDefinition	Attribute Definition	New Attribute	n/a	Attribut Definition

© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.

DatatypeDefinition	Type Definition	New Typ	n/a	
Sample User-Defined resp. System-Attributes for requirements level	Objects	Requirements	Details (System Attributes)	Alle Attribute werden entweder direkt zum Zieltool abgebildet oder wenn keine 1:1 Mapping möglich ist über ein Präfix mit dem Namen des Partner Tools. Damit können auch Besonderheiten beim Austausch innerhalb der gleichen Toolfamilie im Bezug auf zu übertragende read-only Systemattribute abgedeckt werden. Dazu finden Sie links einige Beispiele. Die grünen Felder zeigt, wie das Partnertool das Attribut des Senders als Präfix im Bezeichner empfängt.
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	Absolut Number	DOORS_Absolut_Number	DOORS_Absolut_Number	Beispiel, wie Absolut Number auf IRqA™ und CaliberRM™ abgebildet wird, das keine Entsprechung in den Partner-Tools hat
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	CaliberRM_Serial Number	CaliberRM_Serial Number	Serial Number	Beispiel, wie ein CaliberRM™ Attribut auf IRqA™ und DOORS™ abgebildet wird
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	Object Number	DOORS_Object_Number	DOORS_Object_Number	Beispiel, wie Object Number auf IRqA™ und CaliberRM™ abgebildet wird, das keine Entsprechung in den Partner-Tools hat
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	Object Heading	Name	Requirement Type	Beispiel für Attribute, die in allen RM Tools direkt gemappt werden können
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	Object Short Text	CaliberRM_Requirement_Name	Requirement name	Beispiel für ein Attribut das zwischen DOORS™ und CaliberRM™ direkt gemappt wird, in IRqA™ aber keine direkte Entsprechung hat. Hier wird ein Austausch zwischen ™ und CaliberRM™ dargestellt. Wenn IRqA™ und DOORS™ austauschen wäre das Attribut für IRqA™ DOORS_Object_Short_Text
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	Object Text	Description	Requirement Description	Beispiel für Attribute die in allen RM Tools ein direktes Pendant haben.
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	Object Level	DOORS_Object_Level	DOORS_Object_Level	Diese drei Zeilen zeigen ein Mapping von 3 fachlich zusammenhängenden, aber nicht identischen Attributen der beteiligten Tools
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	IRqA_Level	Level	IRqA_Level	
SpecType, DatatypeDefinition, AttributeDefinition, AttributeValue	CaliberRM_Hierarchical_Number	CaliberRM_Hierarchical_Number	Hierarchical Number	

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 76 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

Appendix D: RIF test data

The following sample RIF XML file will not be updated within this document. A current version can be downloaded from HIS-Homepage <http://www.automotive-his.de/his-ergebnisse.htm>

3.6 RIF XML File (RIF_EF_example.xml)

```

<?xml version="1.0" encoding="UTF-8"?>
<RIF xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://automotive-his.de/schema/rif" xmlns:rif-
xhtml="http://automotive-his.de/schema/rif-xhtml" xsi:schemaLocation="http://automotive-his.de/schema/rif rif.xsd http://automotive-
his.de/schema/rif-xhtml rif-xhtml.xsd">
<TITLE>RIF InterfaceRequirements Examples</TITLE>
<COMMENT>This RIF EFF has been manually generated for demonstration purposes.</COMMENT>
<AUTHOR>Stefan Hendrata</AUTHOR>
<VERSION>0.1</VERSION>
<CHECKSUM>T.B.D.</CHECKSUM>
<CREATION-TIME>2005-03-02T12:00:00</CREATION-TIME>
<COUNTRY-CODE>DE</COUNTRY-CODE>
<SOURCE-TOOL-ID>Manually written</SOURCE-TOOL-ID>
<DATATYPES>
<DATATYPE-DEFINITION-STRING>
<DESC>This is the standard string type.</DESC>
<IDENTIFIER>ID_StringType_1</IDENTIFIER>
<LONG-NAME>Standard String Type</LONG-NAME>
<MAX-LENGTH>65535</MAX-LENGTH>
</DATATYPE-DEFINITION-STRING>
<DATATYPE-DEFINITION-DOCUMENT>
<DESC>This is the standard document type for XHTML content.</DESC>
<IDENTIFIER>ID_XhtmlType_1</IDENTIFIER>
<LONG-NAME>Standard XHTML type</LONG-NAME>
</DATATYPE-DEFINITION-DOCUMENT>
</DATATYPES>
<SPEC-TYPES>
<SPEC-TYPE>
<LONG-NAME>SpecType_1</LONG-NAME>
<DESC>This is specification type number 1.</DESC>
<IDENTIFIER>ID_SpecType_1</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:13:24</LAST-CHANGE>
<SPEC-ATTRIBUTES>
<ATTRIBUTE-DEFINITION-SIMPLE>
<DESC>This is the attribute definition for the Requirements-ID</DESC>
<IDENTIFIER>ID_AttributeDefinition_1</IDENTIFIER>
<LONG-NAME>ID</LONG-NAME>
<LAST-CHANGE>2005-03-01T10:24:08</LAST-CHANGE>
<TYPE>
<DATATYPE-DEFINITION-STRING-REF>ID_StringType_1</DATATYPE-DEFINITION-STRING-REF>
</TYPE>
</ATTRIBUTE-DEFINITION-SIMPLE>
<ATTRIBUTE-DEFINITION-COMPLEX>
<DESC>This is the attribute definition for the Requirements Text</DESC>
<IDENTIFIER>ID_AttributeDefinition_2</IDENTIFIER>
<LONG-NAME>RIF Customer Requirements f&uuml;r Interfaces</LONG-NAME>
<LAST-CHANGE>2005-03-01T11:34:27</LAST-CHANGE>
</ATTRIBUTE-DEFINITION-COMPLEX>
<ATTRIBUTE-DEFINITION-SIMPLE>
<DESC>This is the attribute definition for the attribute column 'Author' of type String</DESC>
<IDENTIFIER>ID_AttributeDefinition_3</IDENTIFIER>
<LONG-NAME>Author</LONG-NAME>
<LAST-CHANGE>2005-03-01T11:35:43</LAST-CHANGE>
<TYPE>
<DATATYPE-DEFINITION-STRING-REF>ID_StringType_1</DATATYPE-DEFINITION-STRING-REF>
</TYPE>
</ATTRIBUTE-DEFINITION-SIMPLE>
<ATTRIBUTE-DEFINITION-SIMPLE>
<DESC>This is the attribute definition for the attribute column 'Status' of type String</DESC>
<IDENTIFIER>ID_AttributeDefinition_4</IDENTIFIER>
<LONG-NAME>Status</LONG-NAME>
<LAST-CHANGE>2005-03-01T11:37:13</LAST-CHANGE>
<TYPE>
<DATATYPE-DEFINITION-STRING-REF>ID_StringType_1</DATATYPE-DEFINITION-STRING-REF>
</TYPE>
</ATTRIBUTE-DEFINITION-SIMPLE>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 77 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<ATTRIBUTE-DEFINITION-SIMPLE>
<DESC>This is the attribute definition for the attribute column 'Erstelldatum' of type String</DESC>
<IDENTIFIER>ID_AttributeDefinition_5</IDENTIFIER>
<LONG-NAME>Erstelldatum</LONG-NAME>
<LAST-CHANGE>2005-03-01T11:38:44</LAST-CHANGE>
<TYPE>
<DATATYPE-DEFINITION-STRING-REF>ID_StringType_1</DATATYPE-DEFINITION-STRING-REF>
</TYPE>
</ATTRIBUTE-DEFINITION-SIMPLE>
</SPEC-ATTRIBUTES>
</SPEC-TYPE>
<SPEC-TYPE>
<LONG-NAME>SpecType_2</LONG-NAME>
<DESC>This is specification type number 2.</DESC>
<IDENTIFIER>ID_SpecType_2</IDENTIFIER>
<LAST-CHANGE>2005-03-02T10:41:28</LAST-CHANGE>
<SPEC-ATTRIBUTES>
<ATTRIBUTE-DEFINITION-SIMPLE>
<DESC>This is the attribute definition for the Requirements-ID</DESC>
<IDENTIFIER>ID_AttributeDefinition_1001</IDENTIFIER>
<LONG-NAME>ID</LONG-NAME>
<LAST-CHANGE>2005-03-02T10:24:08</LAST-CHANGE>
<TYPE>
<DATATYPE-DEFINITION-STRING-REF>ID_StringType_1</DATATYPE-DEFINITION-STRING-REF>
</TYPE>
</ATTRIBUTE-DEFINITION-SIMPLE>
<ATTRIBUTE-DEFINITION-COMPLEX>
<DESC>This is the attribute definition for the Requirements Text</DESC>
<IDENTIFIER>ID_AttributeDefinition_1002</IDENTIFIER>
<LONG-NAME>SHO Created 20050215 & Modified 15Feb</LONG-NAME>
<LAST-CHANGE>2005-03-02T11:34:27</LAST-CHANGE>
</ATTRIBUTE-DEFINITION-COMPLEX>
</SPEC-ATTRIBUTES>
</SPEC-TYPE>
</SPEC-TYPES>
</SPEC-OBJECTS>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_97</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_2</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h1>Interface Requirements</rif-xhtml:h1>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_3</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>H&ouml;h_HOOD</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_4</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 78 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>Hintergrundinfo.n/a</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_5</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>22 October 2004</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_2</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_6</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_308</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_7</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h2>Dokument Historie</rif-xhtml:h2>
<rif-xhtml:p>
<rif-xhtml:strong>Status, Version</rif-xhtml:strong>
<rif-xhtml:br/><rif-xhtml:strong>Author</rif-xhtml:strong>
<rif-xhtml:br/>
<rif-xhtml:strong>Datum</rif-xhtml:strong>
</rif-xhtml:p>
<rif-xhtml:p>Draft, V.0.0</rif-xhtml:br/>HOOD Group</rif-xhtml:br/>20.10.2004 </rif-xhtml:p>
<rif-xhtml:p>Draft, V.0.5</rif-xhtml:br/>HOOD Group</rif-xhtml:br/>24.10.2005 </rif-xhtml:p>
</rif-xhtml:div>
</XHTML-CONTENT>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_8</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE/>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_9</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>vorgeschlagen</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_10</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>24 February 2005</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 79 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_3</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_11</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_300</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_12</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h2>Fehlerbehandlung</rif-xhtml:h2>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_13</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>H&ouml;l;h_HOOD</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_14</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>Hintergrundinfo.n/a</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_15</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>26 January 2005</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_4</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_16</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_210</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_17</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 80 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>Beim Import muss &uuml;berpr&uuml;ft werden k&ouml;nne, ob Ownership oder Berechtigungen (Authorisation) verletzt
worden sind. (Anf. an Importer)</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_18</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>Geischer/Maier_BMW</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_19</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>vorl&auml;ufig_best&auml;tigt_23Nov_abends</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_20</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>17 November 2004</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_5</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_21</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_139</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_22</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>Der Anwender soll gewarnt werden, wenn versucht wird eine zu einem fr&uuml;heren Zeitpunkt generierte
Austauschformat-Datei (EF) &uuml;ber eine neuere zu importieren.</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_23</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>Mtg13.Okt</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_24</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>Mtg23Nov_best&auml;tigt</THE-VALUE>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 81 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_25</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>25 October 2004</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_6</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_26</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_291</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_27</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h2>F&uuml;hrung durch den Workflow</rif-xhtml:h2>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_28</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>H&ouml;h_HOOD</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_29</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>Hintergrundinfo,n/a</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_30</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>24 November 2004</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_7</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-01T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_31</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:48</LAST-CHANGE>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 82 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_IR_303</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_32</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>Beispiele f&uuml;r die Benutzerf&uuml;hrung befinden sich in einem eigenen Dokument.</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_33</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:13:58</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_3</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>H&ouml;h_HOOD</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_34</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:05</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_4</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>vorgeschlagen</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_35</IDENTIFIER>
<LAST-CHANGE>2005-03-01T14:14:16</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_5</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>02 February 2005</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1001</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1001</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_1</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1002</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h1>Diese Dokumentation behandelt exemplarischen Workflow f&uuml;r das Austauschformat:</rif-xhtml:h1>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1002</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 83 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1003</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_2</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1004</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:h2>Dokument Historie</rif-xhtml:h2>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1003</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1005</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_114</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1006</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:object data="revision_history_table.xls" name="RevisionHistory" classid="00020820-0000-0000-C000-000000000046"
type="Excel.Sheet"/>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1004</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1007</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_55</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1008</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 84 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h1>Einführung</rif-xhtml:h1>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1005</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1009</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_56</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1010</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h2>Ziele des Dokuments:</rif-xhtml:h2>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1006</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1011</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_58</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1012</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>Beispiele für eine Integration an der Präsentationsschicht.</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1007</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1013</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 85 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_59</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1014</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h1>Workflow -Konfigurierbarkeits / Parametrierungs-Vorschlag</rif-xhtml:h1>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1008</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1015</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_60</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1016</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:object data="workflow_image.bmp" name="Workflow" classid="D3E34B21-9D75-101A-8C3D-00AA001A1652"
type="Paint.Picture"/>
</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1009</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
<VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1017</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_134</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1018</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>
<rif-xhtml:h2>Workflow</rif-xhtml:h2>
</rif-xhtml:div>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 86 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
<SPEC-OBJECT>
<IDENTIFIER>ID_SpecObj_1010</IDENTIFIER>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<LAST-CHANGE>2005-03-02T14:12:56</LAST-CHANGE>
</VALUES>
<ATTRIBUTE-VALUE-SIMPLE>
<IDENTIFIER>ID_AttributeValue_1019</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:48</LAST-CHANGE>
<DEFINITION>
<ATTRIBUTE-DEFINITION-SIMPLE-REF>ID_AttributeDefinition_1001</ATTRIBUTE-DEFINITION-SIMPLE-REF>
</DEFINITION>
<THE-VALUE>RIF_WF_61</THE-VALUE>
</ATTRIBUTE-VALUE-SIMPLE>
<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
<IDENTIFIER>ID_AttributeValue_1020</IDENTIFIER>
<LAST-CHANGE>2005-03-02T14:13:50</LAST-CHANGE>
<DEFINITION>
<DATATYPE-DEFINITION-DOCUMENT-REF>ID_XhtmlType_1</DATATYPE-DEFINITION-DOCUMENT-REF>
</DEFINITION>
<XHTML-CONTENT>
<rif-xhtml:div>Abhängig von der RM-Tool Technologie gibt es einige mögliche Lösungen zur Implementierung einer
Import und Exportsschnittstelle. Ein Austausch auf dem Präsentation Layer könnte auf verschiedene Weise implementiert
werden: Z.B. durch ein vorgeschaltetes (eigenständiges) Tool (z.B. PlugIn/ AddIn Dialog), als Script Dialog, oder eine Abfolge von
Windows Dialogen.</rif-xhtml:div>
</XHTML-CONTENT>
</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT>
</VALUES>
</SPEC-OBJECT>
</SPEC-OBJECTS>
<SPEC-GROUPS>
<SPEC-GROUP>
<IDENTIFIER>ID_SpecGroup_1</IDENTIFIER>
<LONG-NAME>RIF_InterfaceRequirements_Samples</LONG-NAME>
<LAST-CHANGE>2005-03-01T15:01:59</LAST-CHANGE>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<SPEC-OBJECTS>
<SPEC-OBJECT-REF>ID_SpecObj_1</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_2</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_3</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_4</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_5</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_6</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_7</SPEC-OBJECT-REF>
</SPEC-OBJECTS>
<RELATION-GROUPS>
<RELATION-GROUP>
<IDENTIFIER>ID_RelationGroup_1</IDENTIFIER>
<DESC>This is some relation description.</DESC>
<LONG-NAME>Just a relation</LONG-NAME>
<RELATION-TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</RELATION-TYPE>
<SPEC-RELATIONS>
<SPEC-RELATION-REF>ID_SpecRelation_1</SPEC-RELATION-REF>
</SPEC-RELATIONS>
</RELATION-GROUP>
</RELATION-GROUPS>
</SPEC-GROUP>
<SPEC-GROUP>
<IDENTIFIER>ID_SpecGroup_2</IDENTIFIER>
<LONG-NAME>RIF_Workflow_Samples</LONG-NAME>
<LAST-CHANGE>2005-03-02T15:01:59</LAST-CHANGE>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
</SPEC-OBJECTS>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 87 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<SPEC-OBJECT-REF>ID_SpecObj_1001</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1002</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1003</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1004</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1005</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1006</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1007</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1008</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1009</SPEC-OBJECT-REF>
<SPEC-OBJECT-REF>ID_SpecObj_1010</SPEC-OBJECT-REF>
</SPEC-OBJECTS>
<RELATION-GROUPS>
<RELATION-GROUP>
<IDENTIFIER>ID_RelationGroup_2</IDENTIFIER>
<DESC>This is some other relation description.</DESC>
<LONG-NAME>Yet another relation</LONG-NAME>
<RELATION-TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</RELATION-TYPE>
<SPEC-RELATIONS>
<SPEC-RELATION-REF>ID_SpecRelation_1001</SPEC-RELATION-REF>
</SPEC-RELATIONS>
</RELATION-GROUP>
</RELATION-GROUPS>
</SPEC-GROUP>
</SPEC-GROUPS>
<SPEC-HIERARCHY-ROOTS>
<SPEC-HIERARCHY-ROOT>
<IDENTIFIER>ID_SpecHierarchyRoot_1</IDENTIFIER>
<LONG-NAME>RIF_InterfaceRequirements_Samples</LONG-NAME>
<LAST-CHANGE>2005-03-01T15:03:16</LAST-CHANGE>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_2</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_2</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_3</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_3</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_4</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_4</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_5</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_5</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_6</IDENTIFIER>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 88 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_6</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_7</IDENTIFIER>
<LAST-CHANGE>2005-03-01T15:04:02</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_7</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY-ROOT>
<SPEC-HIERARCHY-ROOT>
<IDENTIFIER>ID_SpecHierarchyRoot_2</IDENTIFIER>
<LONG-NAME>RIF_Workflow_Samples</LONG-NAME>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1001</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1001</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1002</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1002</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1003</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1003</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1004</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1004</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1005</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1005</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1006</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1006</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>

```

HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 89 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

```

</CHILDREN>
</SPEC-HIERARCHY>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1007</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1007</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1008</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1008</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1009</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1009</SPEC-OBJECT-REF>
</SPEC-OBJECT>
<CHILDREN>
<SPEC-HIERARCHY>
<IDENTIFIER>ID_SpecHierarchy_1010</IDENTIFIER>
<LAST-CHANGE>2005-03-02T15:03:16</LAST-CHANGE>
<SPEC-OBJECT>
<SPEC-OBJECT-REF>ID_SpecObj_1010</SPEC-OBJECT-REF>
</SPEC-OBJECT>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY>
</CHILDREN>
</SPEC-HIERARCHY-ROOT>
</SPEC-HIERARCHY-ROOTS>
<SPEC-RELATIONS>
<SPEC-RELATION>
<IDENTIFIER>ID_SpecRelation_1</IDENTIFIER>
<LAST-CHANGE>2005-03-02T16:25:10</LAST-CHANGE>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_1</SPEC-TYPE-REF>
</TYPE>
<SOURCE>
<SPEC-OBJECT-REF>ID_SpecObj_7</SPEC-OBJECT-REF>
</SOURCE>
<TARGET>
<SPEC-OBJECT-REF>ID_SpecObj_1010</SPEC-OBJECT-REF>
</TARGET>
</SPEC-RELATION>
<SPEC-RELATION>
<IDENTIFIER>ID_SpecRelation_1001</IDENTIFIER>
<LAST-CHANGE>2005-03-02T16:25:10</LAST-CHANGE>
<TYPE>
<SPEC-TYPE-REF>ID_SpecType_2</SPEC-TYPE-REF>
</TYPE>
<SOURCE>
<SPEC-OBJECT-REF>ID_SpecObj_1010</SPEC-OBJECT-REF>
</SOURCE>
<TARGET>
<SPEC-OBJECT-REF>ID_SpecObj_7</SPEC-OBJECT-REF>
</TARGET>
</SPEC-RELATION>
</SPEC-RELATIONS>
<ACCESS-POLICIES>
<ACCESS-POLICY>
<DESC>This access policy allows editing of the hierarchy structure and of the requirements text of all requirements.</DESC>
<IDENTIFIER>ID_AccessPolicy_1</IDENTIFIER>
<ACCESS-MODE>EDIT</ACCESS-MODE>
<SPEC-HIERARCHIES>
<SPEC-HIERARCHY-REF>ID_SpecHierarchy_1</SPEC-HIERARCHY-REF>
<SPEC-HIERARCHY-REF>ID_SpecHierarchy_1001</SPEC-HIERARCHY-REF>
</SPEC-HIERARCHIES>

```


HIS	Requirements Interchange Format (RIF)	Version 1.0	Page 91 of 91
© 2005- HIS Members and Partners. Distribution allowed for all HIS members and partners Adam Opel AG, Conti Temic GmbH, Carmeq GmbH & HOOD Group. Copyright protection enforced.			

<ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1002</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1003</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1004</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1005</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1006</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1007</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1008</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1009</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1010</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1011</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1012</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1013</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1014</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1015</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1016</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1017</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1018</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1019</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 <ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>ID_Attribute_Value_1020</ATTRIBUTE-VALUE-EMBEDDED-DOCUMENT-REF>
 </ATTRIBUTE-VALUES>
 </ACCESS-POLICY>
 </ACCESS-POLICIES>
 </RIF>