SIEMENS

Requirements Management at RI

RM Process Manual - Application Rules

	Name	Department	Date	Manual signature
Released by	Timo Franz	SMO RI BE	01.09.2022	sgd. FRANZ_TIM1
Formally checked by	Ralf Numrich	SMO RI BE QM	2022-09-01	sgd. NUM-RICH_RAL
Checked by	Christian Benkert	SMO RI MT PE PMO	2022-09-01	sgd. Christian Benkert
Author	Stefan Jung	SMO RI BE QM	2022-08-18	sgd. JUNG_STE

Status: freigegeben FRANZ_TIM1 01.09.2022
DCC: EBD022 OKZ Prod: TS_S

SIEMENS

Document history

Version Ext. ID	Version Int. ID	Date	Author	Sections changed	Change
---	-	2013-08-07	Stefan Jung	all	New document
---	A	2014-04-25	Stefan Jung	all all 4.2.2.2 5.4 7.1, 7.3, 7.4 7.5	New RA document template CR CFX00184351 implemented Consideration of R&D RM Interface description Processing in Dev. Projects Update procedure added
---	B	2016-05-02	Stefan Jung	all	Update according to organisa-tional changes
---	C	2022-09-01	Stefan Jung	all	Adaption to new RM process for project execution

Siemens Mobility GmbH
Postfach 33 27
38023 Braunschweig
Germany
www.siemens.com

SIEMENS

Contents

1 Introduction

1.1 Scope and Application

This manual defines the process for requirements management for definition of application rules and their processing and tracking in generic SSC projects (following a defined standard system configuration), customer projects or development projects according to the standard [EN_50129]. Tailoring of the process is possible.

The manual is written in the main tool-neutrally. Tool specific handling instructions are described in separate documents which are referred to in Appendix A IT Tool specific Manuals.

This manual includes it´s placement (chapter 2) and an introduction to Requirements Management (RM) (chapter 3).

1.2 Validity and Update

The present document is valid

1. for the Requirements Management (RM) of product development projects for elicitation of application rules and
2. for the Requirements Management (RM) of all projects (customer, generic SSC and development projects) at the design phase for processing of all application rules of all used components and systems.

This manual is valid and applicable for the business unit SMO RI worldwide.

The process management for RM is responsible for editing and updating the present document. The head of BE, who is the process owner, releases this document.

The present document is changed and updated when required. All changes are documented in the change list and identified by means of version update.

1.3 References and Abbreviations

Referenced documents are listed in chapter 11. Abbreviations used in this document are explained in the project glossary in section 9.3.

2 Regulation for Requirements Management

2.1 Siemens Mobility Company

The chapter 6.1.8 of the Siemens Mobility PM Guideline [SMO_PM_Guideline] defines the RM framework based to be implemented by business units and segments.

2.2 Business Unit Rail Infrastructure

This manual describes the implementation of RM for application rules in the business unit SMO RI based on the process definition in the [RI_RM_Process_in_BIC].

Basis for the project execution is the RM@RI process manual [RM_Process_Manual_PE] respectively the R&D guideline for RM [GL_RM_RD].

3 Requirements Management General

3.1 RM Policy

RM is the systematic approach to support

- identification of all requirements
- tracing of all requirements from identification or definition to their verification and validation
- administration of all information that affect and/or supplement the single requirement
- the quality of delivery in projects (bid and execution)
- reduction of potential risks and overall project costs

RM requires a defined process and an efficient tool support. The RM tasks will be integrated into the PM process whereas certain results of such tasks are mandatory for passing certain PM and development milestones / Q-Gates.

The detailed approach and implementation will be defined in a project specific RM plan together with standard tool support.

3.2 Requirements

The identification, evaluation and documentation of all requirements are important for the success of a project.

Requirements can be obtained by interviews, surveys, statements, technical and non-technical documentation, laws and regulations. Besides from tender, contract documents and other external sources, requirements may also be derived internally to specify a complete system.

Requirements are divided into

- technical requirements (e.g. system requirements)
- non-technical requirements (e.g. commercial, legal, sales related)

Within the system lifecycle, each single requirement has to be

- identified
- captured
- documented
- classified
- assigned to a responsible domain expert
- detailed and apportioned to the applicable system levels

- fulfilled / answered
- closed
- verified
- validated

For further information on the identification and extraction of requirements and how to transform them into well-formed requirements for assessment please refer to [Guideline_Requirements].

3.2.1 Definition of Requirements

In [CMMI], the Software Engineering Institute of the Carnegie Mellon University Pittsburgh defines a requirement as:

1. a condition or capability needed by a user to solve a problem or achieve an objective
2. a condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document
3. a documented representation of a condition or capability as in a.) or b.)

3.2.2 Characteristics of Requirements

To simplify the specification of a single requirement, requirements should follow the SMART criteria described below (see also [Requirements_Properties]):

- **S**pecific
  A requirement must say exactly what is required. Specificity actually comprises several areas as follows:
  - clarity i.e. that there is no ambiguity;
  - consistency i.e. that the same terminology has been used throughout the specification to describe the same system element or concept;
  - simplicity i.e. avoid double requirements e.g. X and Y;
  - an appropriate level of detail.
- **M**easurable
  In the context of requirements engineering, measurable means it is possible, once the system has been constructed, to verify that this requirement has been met. In some software engineering methodologies, the Requirements Engineer is instructed to determine the tests which must be performed in order to satisfy the requirement. If a test is the intended verification method, the requirement should always be expressed in a way allowing for positive testing.

- **A**ttainable
  An attainable requirement can physically be exhibited by the system under the given conditions. Some requirements may be beyond the bounds of human knowledge. Others may have theoretical solutions but be beyond what is currently achievable. This kind of requirements is not atypical. The consequence of attempting to meet these requirements is that the system will never be accepted or be prohibitively expensive or both. In general terms the following guidelines are recommended:
  - Is there a theoretical solution to the problem?
  - Has it been done before? If not, why not?
  - Has a feasibility study been done?
  - Is there an overriding constraint which prohibits this requirement?
  - Are there physical constraints on the size of the memory, processor or peripherals?
  - Are there environmental constraints such as temperature, compressed air?

It is often the case that the attainable and realizable criteria are often considered in parallel. This does not however make them synonymous.

- **R**ealizable
  In the context of system and software requirements, realizable means it is possible to achieve this requirement given what is known about the constraints under which the system and the project must be developed.
- **T**raceable
  Requirements Traceability is the ability to trace (forwards and backwards) a requirement from its conception through its specification to its subsequent design, implementation and test. This is important for the following reasons:
  - so that we can know and understand the reason for each requirement's inclusion within the system.
  - so that we can verify that each requirement has been implemented.
  - so that necessary modifications can be made without endangering the consistency and completeness of the system.

3.2.3 Types of Requirements

As well as all RM process manuals this manual is dealing with different basic types of requirements. In the following the types of requirements are listed which can be influenced by them in a customer project or generic SSC project.

- **Application Rules**
  Application rules are requirements which have to be considered by the customer or the project using the respective component or subsystem. These rules shall be fulfilled to ensure a safe and dependable usage of each component in the system.
- Stakeholder Requirements
  - Requirements explicitly defined or indirectly communicated by a stakeholder (e.g. customer)

- Assumptions
  are taken to support the evaluation of a requirement and need to be clarified and agreed with the customer before signing the contract. The assumptions may address
  - implicit requirements which are not expressed by the customer
  - key issues which are essential for the system design but are not defined by the customer
  - surrounding conditions regarding the system environment (climate, EMC, timetables, capability of adjacent/included (sub-)systems etc.)
- Internally Derived Requirements
  Requirements derived internally to clarify and describe the full functioning system. They are only internally relevant and do not need to be forwarded to the customer (e.g. cross-project environmental specification of SMO RI).

If assumptions, internally derived requirements and application rules are of importance for a specific project, they are not mixed with customer requirements in one document or module. They shall be kept separate and managed separately.

3.2.4 Handling of Technical Requirements

Technical requirements are the basis for system design and development and for the preliminary as well as final architecture. The management of technical requirements follows the procedures defined in [RM_Process_Manual_PE].

3.2.5 Handling of Non-Technical Requirements

Although non-technical requirements do not affect the system design directly, their number and complexity require proper management, too. In most cases the use of a tool-based RM for non-technical requirements is reasonable or even decisive for the overall success of the project. The applicable process scale (that may define different procedures for technical and non-technical requirements respectively) depends on the actual project setup.

Non-technical requirements shall be handled according to the procedure as in [RM_Process_Manual_PE] for the project execution.

4 RM Process for Handling Application Rules

The overall RM process for SMO RI is defined in process documentation in BIC Cloud of SMO RI [RI_RM_Process_in_BIC]. The interfaces of the RM process including in- and outputs to the product lifecycle management as well as to the development process are defined there.

During the development and/or selection of subsystems, products, and their parts (modules, platforms, etc.) additional requirements occur and shall be confirmed in the validation or assessment phase. These requirements shall be considered for the use of the concerned subsystems and products.

4.1 Process Overview

The following figure gives a general overview of the overall requirements management process, which is applicable for all types of requirements - technical and non-technical (see section 3.2)

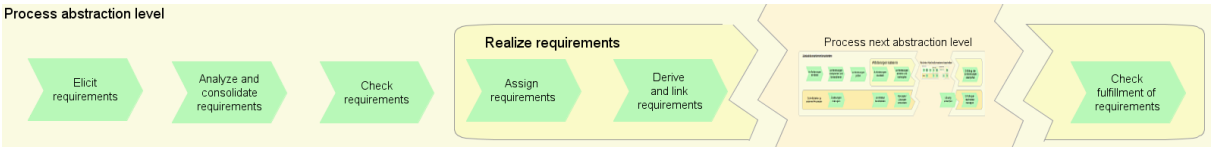

Figure 1 Overview of the Overall RM Process

The process is described in detail in the process definition in BIC Cloud [RI_RM_Process_in_BIC]. The steps of requirements management are

- Elicit requirements
- Analyze and consolidate requirements
- Check requirements
- Assign requirements
- Derive and link requirements
- Check fulfillment of requirements

4.2 Handling and Refinement of Requirements Attributes

Requirements are marked and handled by setting values of the respective attributes.

The setting of attribute values of a requirement needs to be documented with name of the author and date (change history). The same applies for all comments. This is a standard functionality of each commercial RM system.

The attributes to be set are marked as mandatory in the [RI_AR_Standard_Attributes] and in accordingly GUIDE].

For application rules two sets of attribute types (for producing and consuming projects) shall be considered (see Table 1) to cover the 6 steps of the process in Figure 1:

1. For the definition of the application rule (producer) in the steps 1-3.
2. For the handling as a requirement in a project (consumer) in the steps 4-6.

In the following table the process steps are listed with their responsibilities:

Step	Task	Description	Remark	Responsible
1	Elicit requirements	The requirements/rules shall be elicited for the customer project, the generic SSC project or the direct customer depending on the conditions of use, the validation and the assessment		Architect, Safety Manager + Validator of Development Project (R&D)
2	Analyze and consolidate requirements	Based on the reasons for the elicitation of the rules the requirements have to be analyzed by the experts and their rating must be documented in the standard attributes for application rules.	Consideration of the causes for the eliciting: Why is it necessary to consider the rule?	Safety Manager + Validator of Development Project / (PLM)
3	Check requirements	Review of all elicited requirements regarding comprehensibility, completeness and feasibility in projects.	Documented in review protocols	PLM
4	Assign requirements	Assignment of requirements to domain experts in the project	According to recommended responsibility by the originator	SMiP
5	Derive and link requirements	Tracing of the requirements and linking to solutions incl. review	According to RM@RI or GUIDE	Domain Experts / SmiP
6	Check fulfillment of requirements	Validation of the requirement fulfillments according to the linked solutions	According to RM@RI or GUIDE	Validator

Table 1 Process Steps for Handling of Application Rules

4.2.1 Attributes for the Elicitation and Definition

For a comprehensive and homogeneous usage, the following attribute values have to be defined for each application rule during the elicitation and definition phase (Step 1 – 3 of Table 1):

No.	Attribute Name	Description	Type
1	STD Object Type	Possible values: DEF: Application rule Prose: Prose text for headings and explanations	single-enu-meration
2	STD Lifecycle Status	Possible values for the status: • In Creation • In Review • In Revision • Approved • Cancelled • Deleted	single-enu-meration
3	STD Lifecycle Explanation	Explanation to the current lifecycle status of the requirement	text
4	STD Release Number	Current release number of the requirement according to the respective document	single-enu-meration
5	STD Pass Fail Criteria	Defined pass fail criteria for the method of verification	single-enu-meration
6	STD Safety related	Flag for safety relevance with following, possible values: • No • Safety Functional • Safety Integrity (or Yes if there shall be no different classification)	single-enu-meration
7	STD RAM related	Flag for RAM relevance with following, possible values: • No • Yes	single-enu-meration
8	STD Security related	Flag for security relevance with following, possible values: • No • Yes	single-enu-meration
9	STD Functional	Type of the application rule: • technical_functional • technical_non_functional • non_technical	single-enu-meration
10	STD Proposed Responsibility	Responsibility for the application rule proposed by development project: • Customization • Engineering • Verification / Test • Safety • Validation • Operations • Maintenance • Installation • Integration	multiple-enu-meration

No.	Attribute Name	Description	Type
11	STD Related Product Requirements	Traces to the related product or subsystem requirements out of the development project	links / string

Table 2 Attributes for Elicitation of Application Rules

4.2.2 Attributes for the Handling, Tracing and Validation in a Project

The following two subsections display the attributes for handling, tracing and validation in a customer or generic SSC project as well as in development project for the process steps 4 - 6.

4.2.2.1 Customer or Generic SSC Projects

The following attributes have to be processed for each application rule transferred/ imported to the customer or generic SSC RM project. During the project execution phase (Step 4-6 of Table 1) the application rules must be processed according to [RM_Process_Manual_PE] and treated there as requirements:

No.	Attribute Name	Description	Type
1	REQ Type	used to clearly mark requirements (DEF) and informative text only (Prose)	single-enumeration
2	REQ Progress	Progress status of the requirement	single-enumeration
3	REQ Forwarding	forwarding target(s) for the requirement	multi-enumeration
4	REQ Statement	statement to the requirement	text
5	REQ Pass Fail Criteria	provision of a criteria as method for verification and/or validation	text
6	REQ Origin	information about the origin of the requirement	text / single-enumeration
7	REQ Safety_related	indicator if the requirement is safety related	single-enumeration
8	REQ RAM_related	indicator if the requirement is RAM related	single-enumeration

Table 3 Attributes for Processing of Application Rules in Customer or Generic SSC Projects

4.2.2.2 Development Projects following GUIDE

The following attributes have to be processed for each application rule transferred/ imported as requirement to the RM development project. During the project phase the application rules have to be processed according to GUIDE] and treated there as requirements:

No.	Attribute Name	Description	Type
1	REQ Type	used to clearly mark the referred requirements and informative text only	single-enumeration
2	REQ Imported ID	ID of origin in the central source document/module (e.g. external document)	string
3	REQ Status	life-cycle status of the requirement	single-enumeration
4	REQ Comment	annotations to the requirement	text
5	REQ Release	assignment of the requirement for a release	text
6	REQ Origin	information about the origin of the requirement	text / single-enumeration
7	REQ SafetyRelevant	indicator if the requirement is safety relevant	single-enumeration
8	REQ Subsystem	indicates the subsystem of the system architecture to which the requirement is assigned	multiple-enumeration

Table 4 Attributes for Processing of Application Rules in Development Projects

4.3 Central Provision of Application Rules for a Product

Access must be ensured for all projects to the application rules and their attributes of all products integrated to their system via links of the used RM system. According to the RM@RI process manual for the project execution phase [RM_Process_Manual_PE] (section 7.3) all application rules must be integrated in the customer or generic SSC RM project on system level. From there the application rules can be tracked and solved via processing the project V model. This proceeding follows the RM rules for project execution.

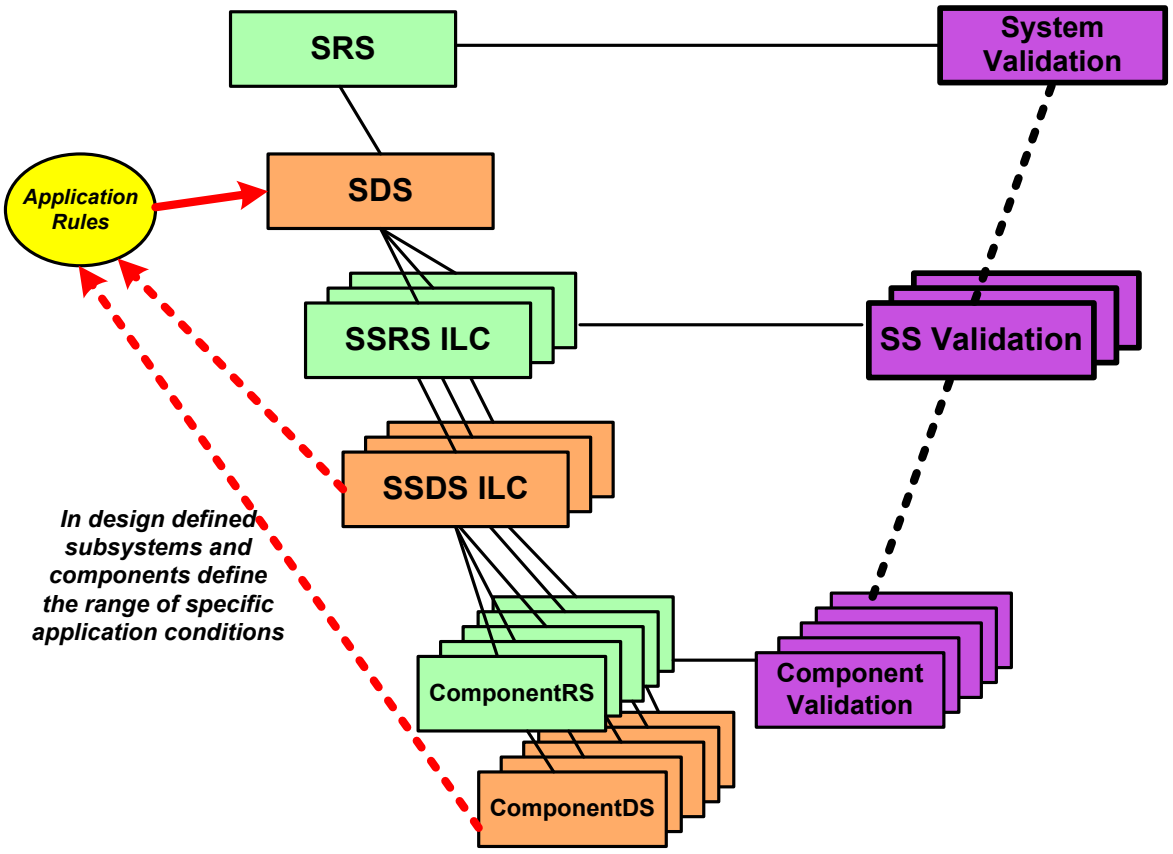

Figure 2 Integration of Application Rules in the RM of Project Execution

SIEMENS

5 Implementation of Application Rules in RM

5.1 Central Storage Location for Application Rules

The application rules with all necessary versions have to be provided on a central storage location in the RM database system by the responsible person of each product. The structure inside the RM database system follows the product hierarchy of the product portfolio management.

The folder structure in the central storage location considers the standard system configuration (SSC)/ project types (PT). In the current RM system of SMO RI (located in the root folder /Products/RA Application Conditions) the products are grouped by their product groups:

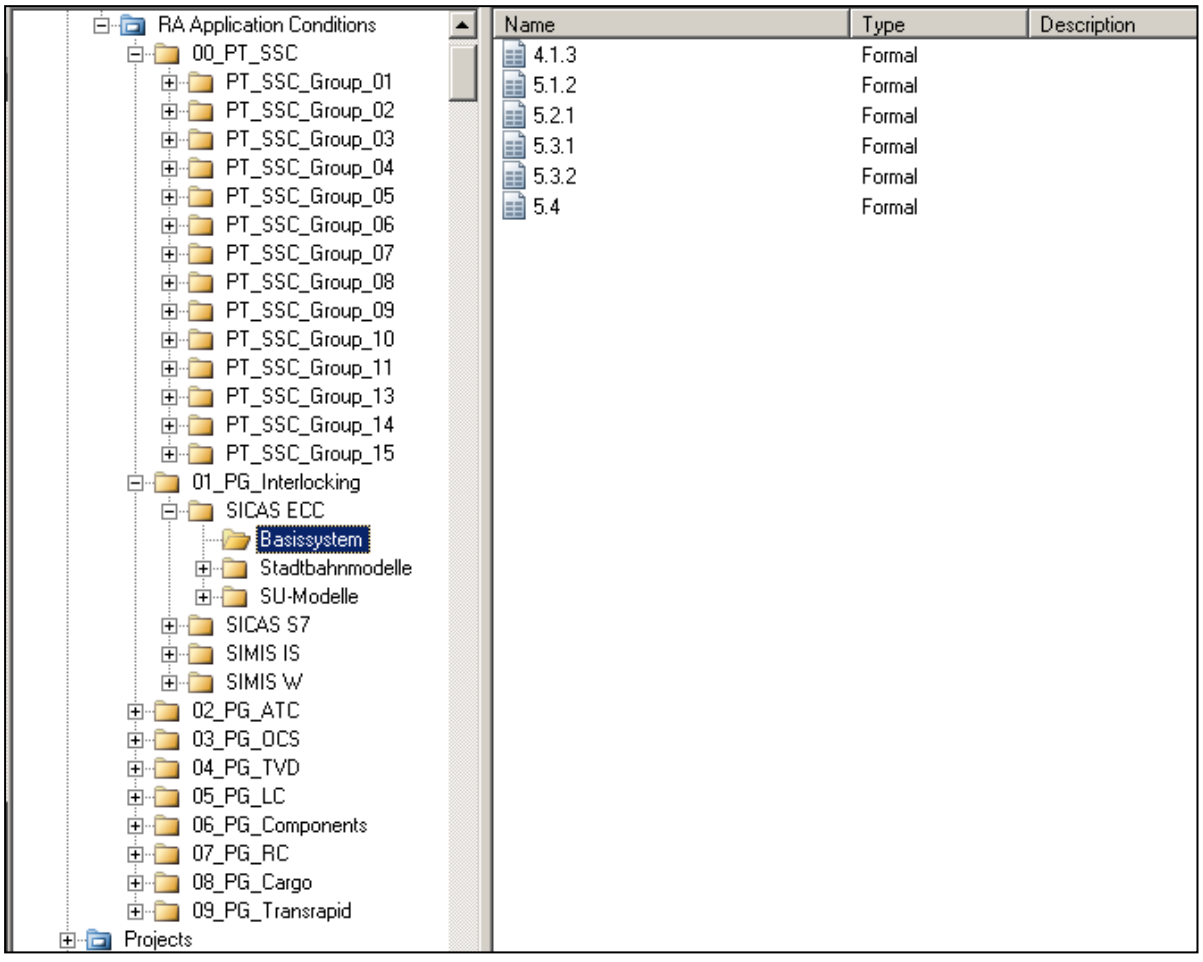

Figure 3 Structure of Central Application Rules

The name and the description in the RM system have to follow the global product specific glossary and systematics.

5.2 Elicitation, Transfer and Usage of Existing Application Rules

To ensure the completeness of all necessary application rules the existing documents containing at least one application rule are imported to the RM system. For each version of a product (or subsystem) a data container is created (Setup of a formal module in the current RM system DOORS). Each module contains all documents with at least one application rule for the usage of the specific product version.

These documents can be:

- Application rules documents (e.g. SRAC, operating instructions, configuring instructions, installation instructions, maintenance instructions, startup instructions)
- Validation reports
- Safety assessment reports
- Addendums to these documents
- Referenced documents by the four document types mentioned above

According to the different product versions the application rules of a product are linked to an existing predecessor rule and from an existing successor rule. In this way the changes, deletions and creation (via non-existing links) of application rules can be detected. The definition of the links between the rules is described in section 9.2.

In the following an example (SIMIS IS) is provided to show the transferring of application rules and the reuse of solutions from former projects.

The following figures show the steps for transferring the application rules and the reuse of solutions of former projects. First the modules for the product or subsystem (using the interlocking SIMIS IS system as an example for this and the next chapters) must be prepared in the RM database from the documents (with identifier from a EDM system, e.g. HTZ (from Switzerland)):

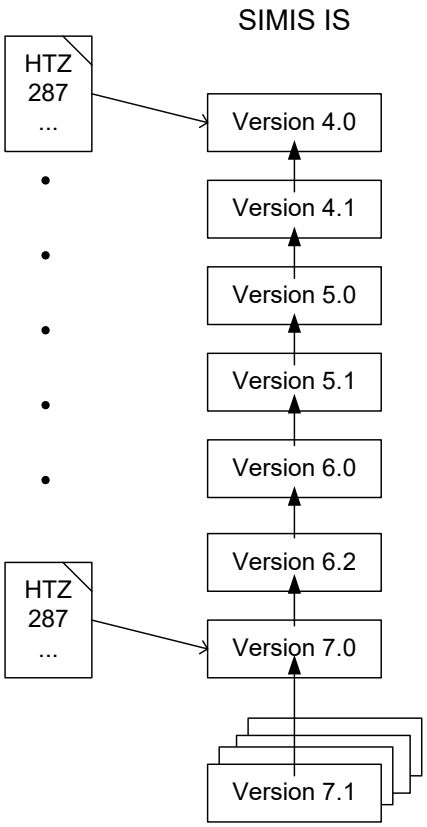

Figure 4 Step 01: Creation of Version Specific Application Rule Modules

In the next step (Figure 5) the application rules are transferred (with tool support, see chapter 8) to a customer project or generic SSC project and processed there. The links between source and target rules are set according to the definition in section 9.2:

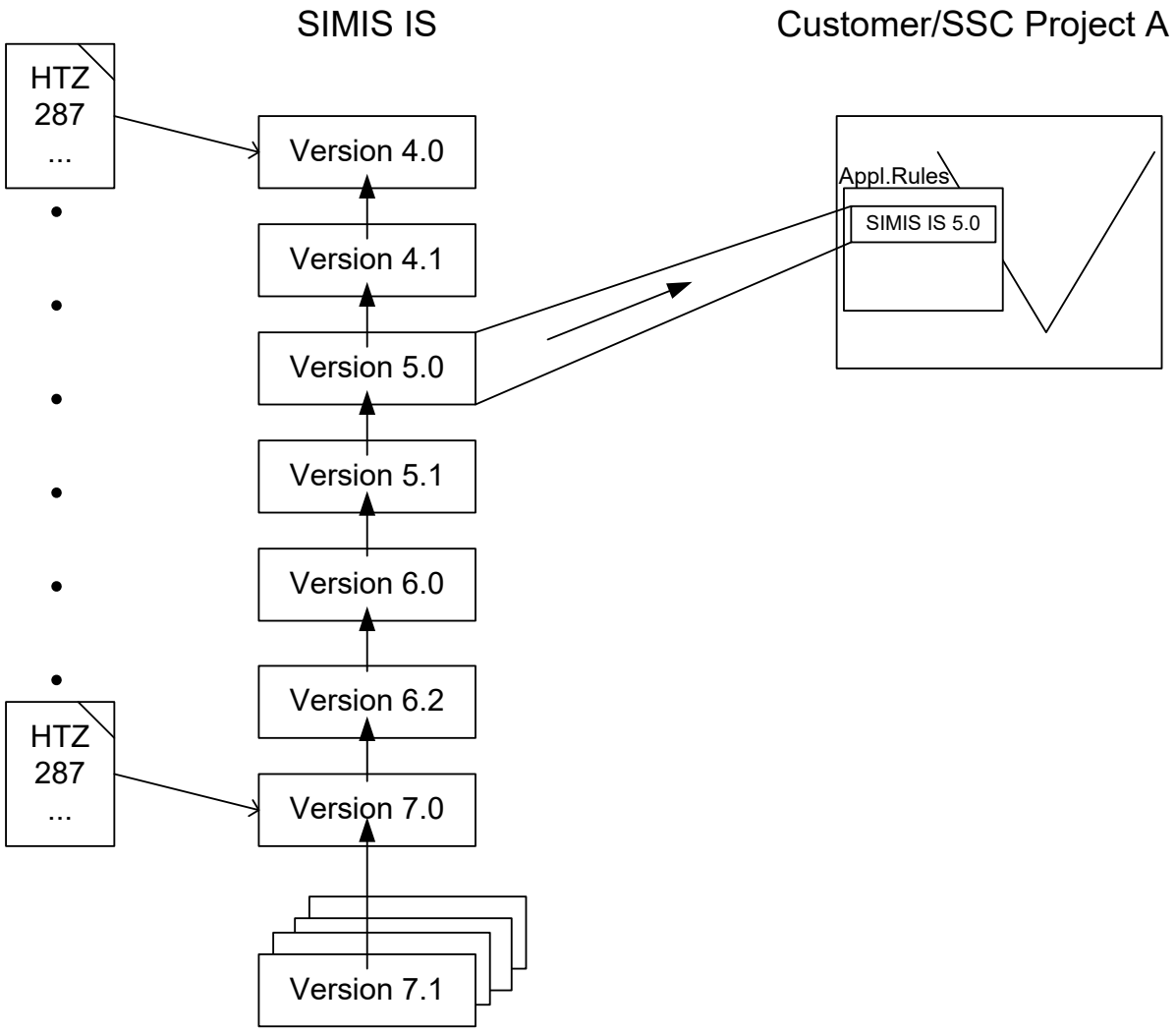

Figure 5 Step 02: Transfer of the Application Rules to a Project

If a project decides to use a newer version of a product, only the new and changed application rules have to be considered after an update from the central storage location. Processed application rules of the previous version which have not been changed must not be evaluated again.

Figure 6 shows that the project specific processing of the application rule will be provided to further projects. The solutions of the project (like design documentation, evidences, etc.) will be stored in the central storage location (with tool support, see chapter 8) and linked to the baseline of the origin application rules according to the link definition in section 9.2:

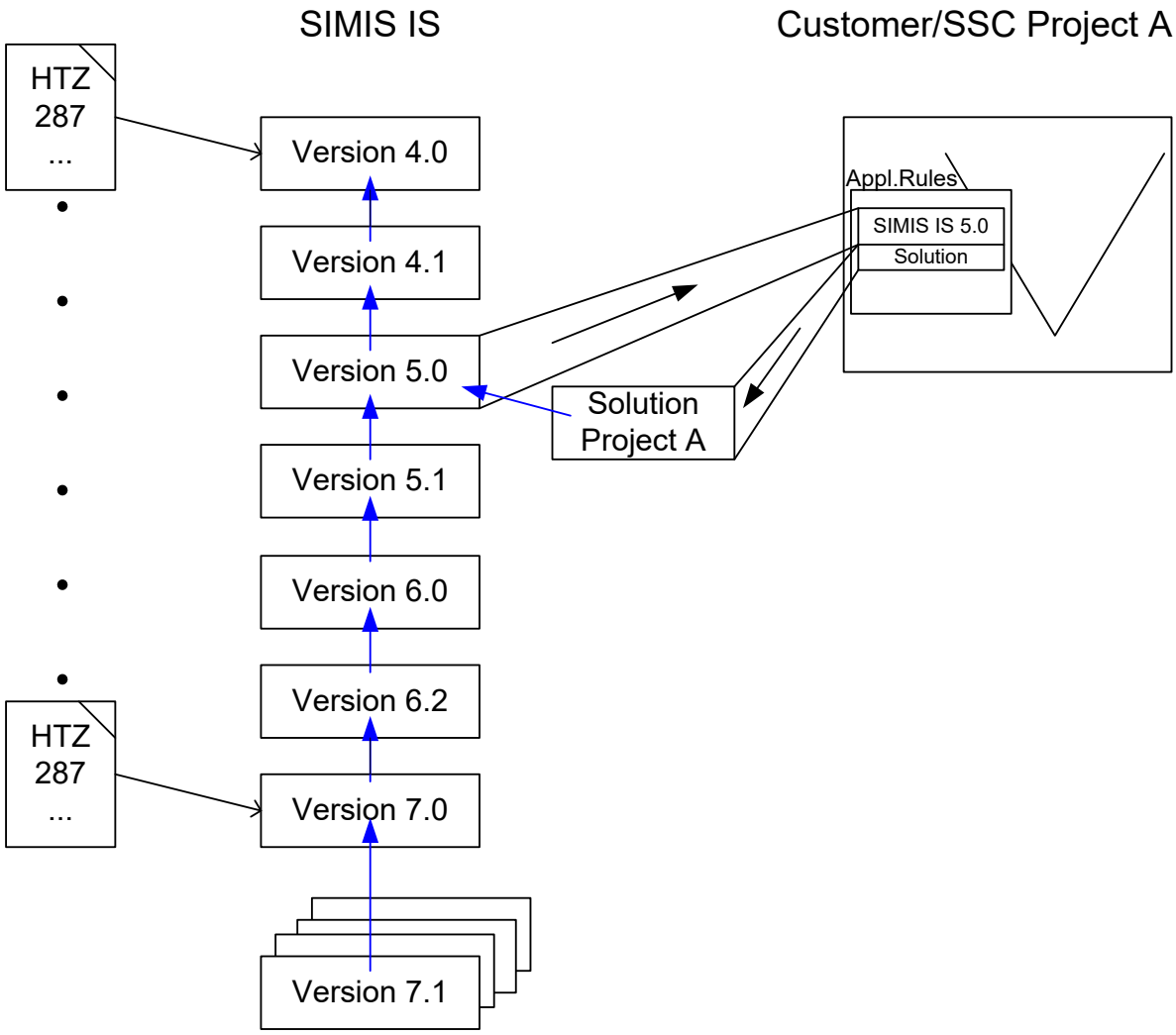

Figure 6 **Step 03: Provision of Solutions for the Application Rules**

In generic SSC projects the evidence documentation shall be defined for the usage in customer projects implementing this SSC.

In the next step (Figure 7) later carried out projects (here customer project B) can read the solutions for the application rules of a former project (here customer project A). Although the new project uses a higher version of the product and therefore updated application rules, it can use the provided information of different versions by other projects for a possible re-use of the solutions.

Via the link between the different versions all changed and non changed application rules can be displayed and their solutions can be taken into account (with tool support, see chapter 8):

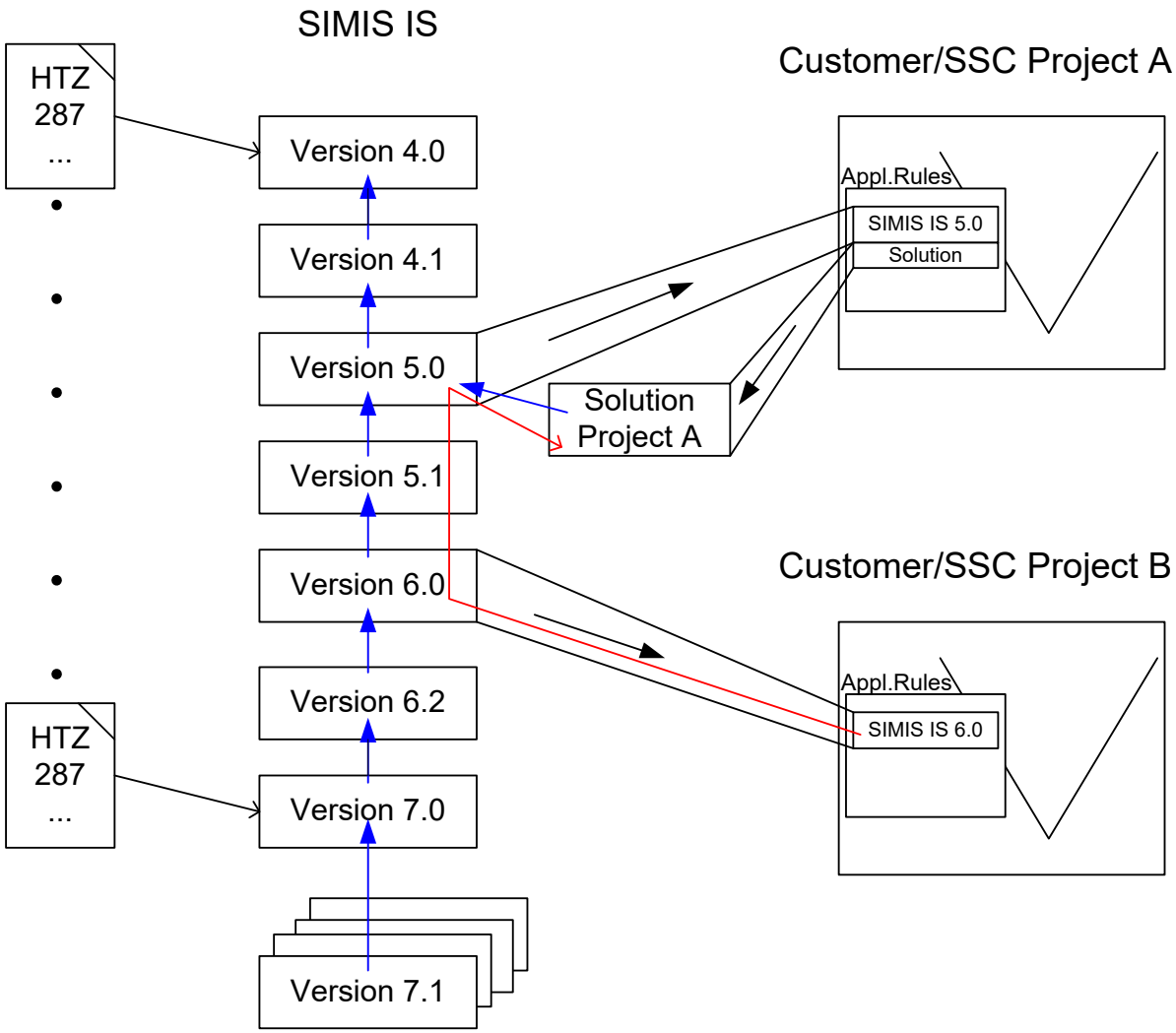

Figure 7 **Step 04: Central Provision of Solutions from Former Projects**

The solutions to the application rules (Figure 8) provided by all projects will be available in the central storage location of the RM system:

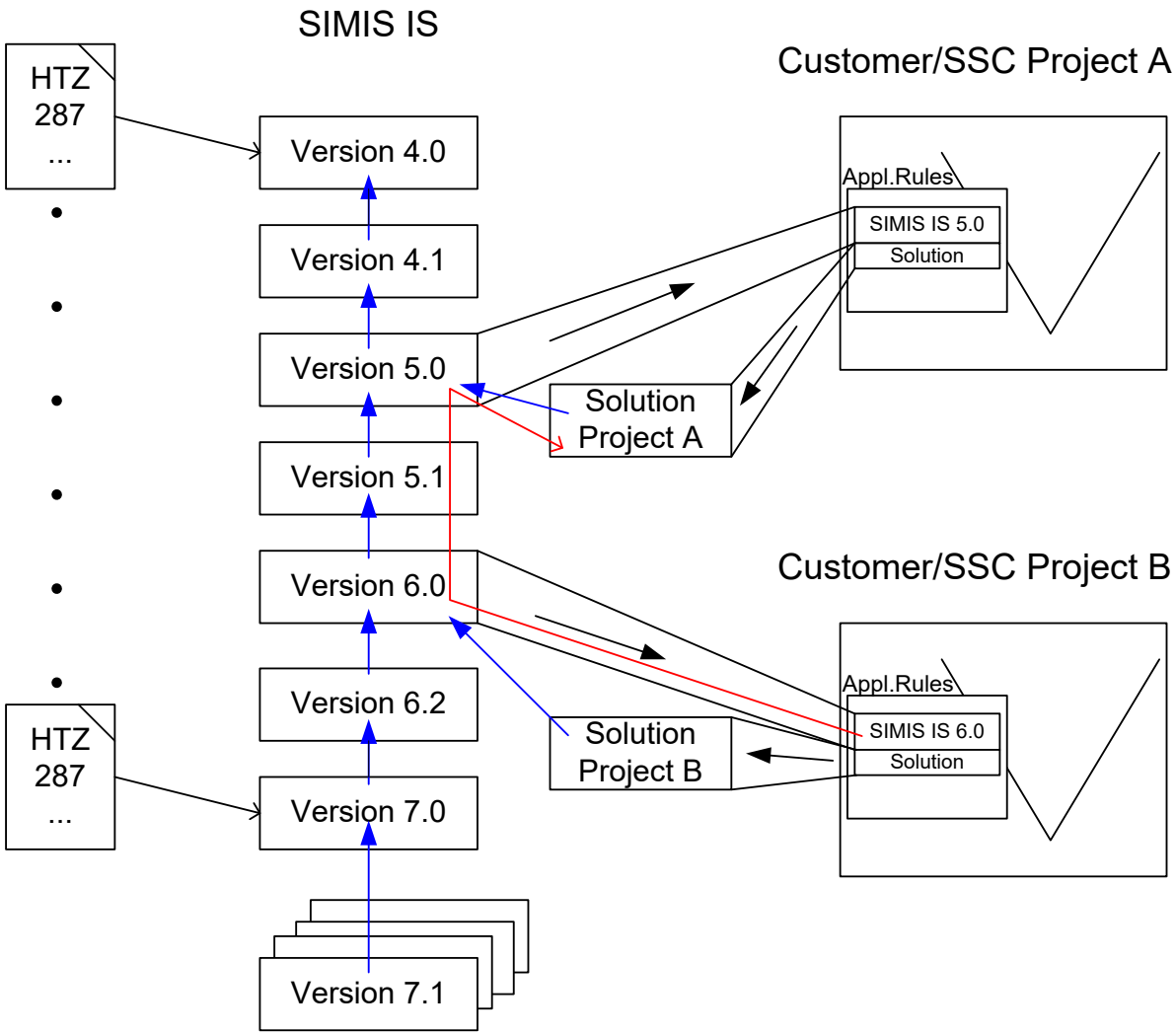

Figure 8 **Step 05: Additional Provision of Solutions for a new Version**

Application rules of higher versions for the same condition shall contain application rules of previous versions in the same definition (same kind of rule text) to ensure downgrade compatibility. New rules shall only be created for new product functions.

5.3 Review of Integration of Existing Application Rules

After transferring all existing documents with application rule for each product into the central storage location the completeness and correctness have to be guaranteed via a review. The responsible product managers have to ensure this task which includes the following steps:

- Check of completeness of all relevant documents
- Check of completeness of objects of each integrated document
- Check of each object setting to DEF (rule) or Prose (prose text)
- Check of integration of all additional information if exists (e.g. Identifier, Note, etc.)
- Check of links to predecessor in a previous version and from successor in a subsequent version

The reviewer has to document all checks for each version in a protocol. The integrity check for the creation of future application rules has to fulfill the requirements for a standard document review (see section 6.2).

5.4 Interface Description between Central Storage and Project

As described in section 5.2 the content of each module contains the application rules of a specific version of a product or subsystem. For composing all application rules in the project specific module/document the responsible requirements manager must take the interface between central storage and project into account during the elicitation of the application rules and later for using the correct module template (with predefined attributes) in the RM project. As the requirements manager set traces between source and target module only some information must be transferred. All other fix attributes like *STD Release Number* or *STD Pass Fail Criteria* can be displayed by following the trace. The interface shall be able to transfer at least the following attributes:

- Headings: *Object Heading* to *Object Heading*
- Short text: *Object Short Text* to *Object Short Text*
- Rule text: *Object Text* to *Object Text*
- Rule definition: *STD Object Type* to *REQ Type*

6 Definition of Application Rules for New Product Versions in the Future

6.1 Proceeding of Defining Application Rules for New Product Versions

After establishing the current status of application rules for each product new version of application rules according to new product releases shall be created directly in the RM system. Based on the predecessor baseline version of the module the responsible author has to modify the module and publish this version as current one.

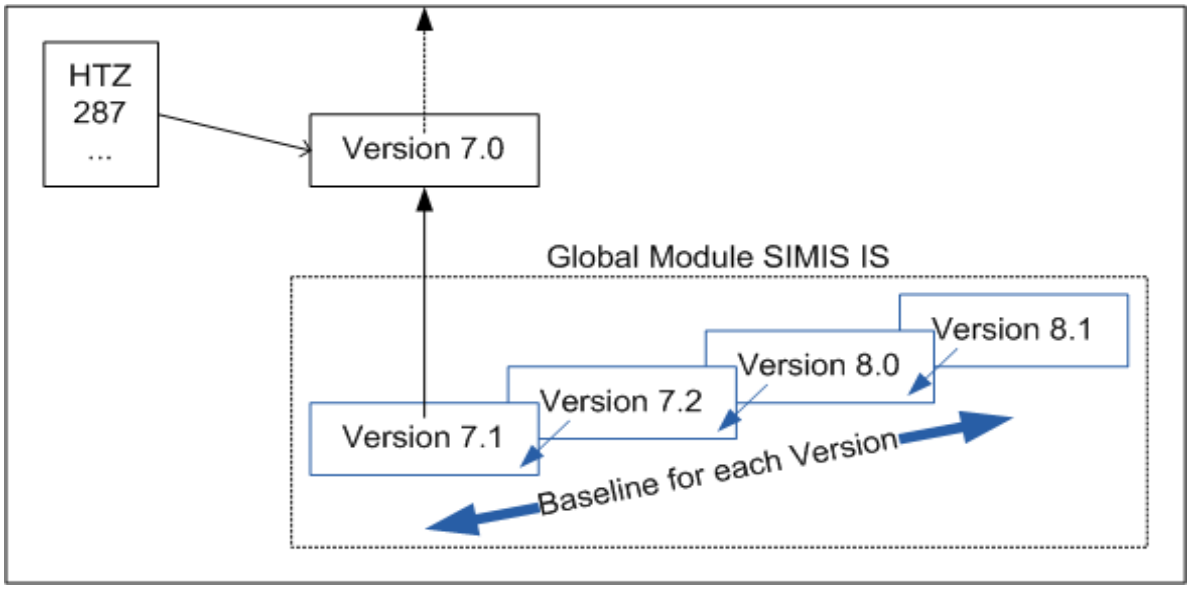

Figure 9 Using one Global Module for Update of Application Rules

The customer projects or generic SSC projects must select the correct baseline of the module for the version of the integrated product or system. This baseline will be transferred to the consuming project and can be handled in the same way as separate modules described in Figure 5. Figure 10 displays the transfer of application rules to a customer project (e.g. for interlocking SIMIS IS).

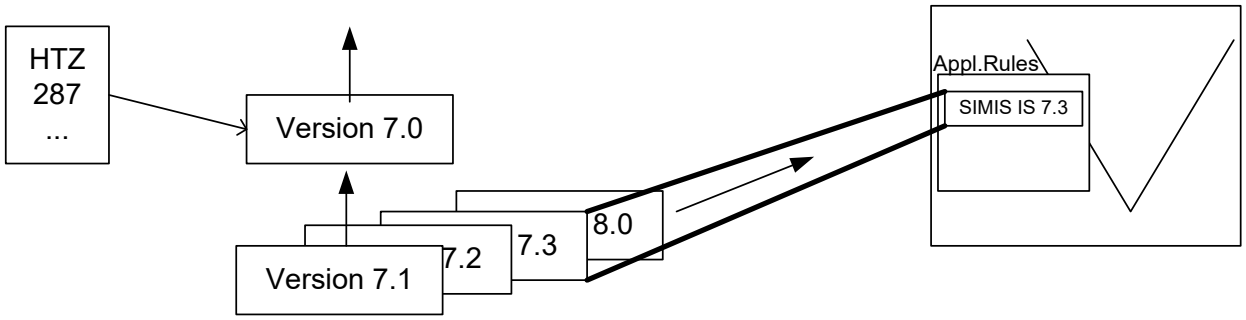

Figure 10 Transferring a Specific Baseline Version to a Customer Project or Generic SSC Project

The collecting of project specific solutions will be executed in the same way but at the concrete baseline version of the global module.

6.2 Review of New Defined Application Rules

Newly defined application rules will be defined and collected in a module based on a previous version. The rules for writing the application rule have been defined in the separate document [ApplicationRule_Definition] which will be mandatory for all development projects in the future.

A specific review must be executed before releasing these rules for usage in customer projects or generic SSC projects. The product manager is responsible for the completeness and correctness of all application rules including all conditions transferred from possibly additional documents, e.g. external assessment reports (see section 5.3).

It is recommended to execute the review of the application rule directly in the RM system. By this way it can be ensured that changes to the previous version can be displayed and checked with less effort.

7 Processing of Application Rules in Projects

7.1 Import and Following the V Model in Projects

After transferring all applicable sets of application rules to the dedicated project folder / module on system level they shall be processed as every system requirement. In the first step the system manager must decide which rules are applicable and which are not applicable. Then the project can close applicable rules with a design solution or assign them to one or more subsystems.

Following the project specific v-shaped process according to RM@RI the rules are assigned, derived and linked and at the end the check of their fulfillment must be carried out via verification and validation. The specific rules for tracing and handling of requirements during the project execution phase are explained in the specific RM manual [RM_Process_Manual_PE]. The training documentation [Processing of Project Execution with DOORS] shows the different tasks for all included roles.

Note: All projects following the GUIDE] process have to process the application rules in a similar way (e.g. SRS is called SyReqSpec, SDS SyArchSpec, etc).

The following Figure 11 displays possible proceedings to fulfill the (safety) application rules (SARs) in the project execution:

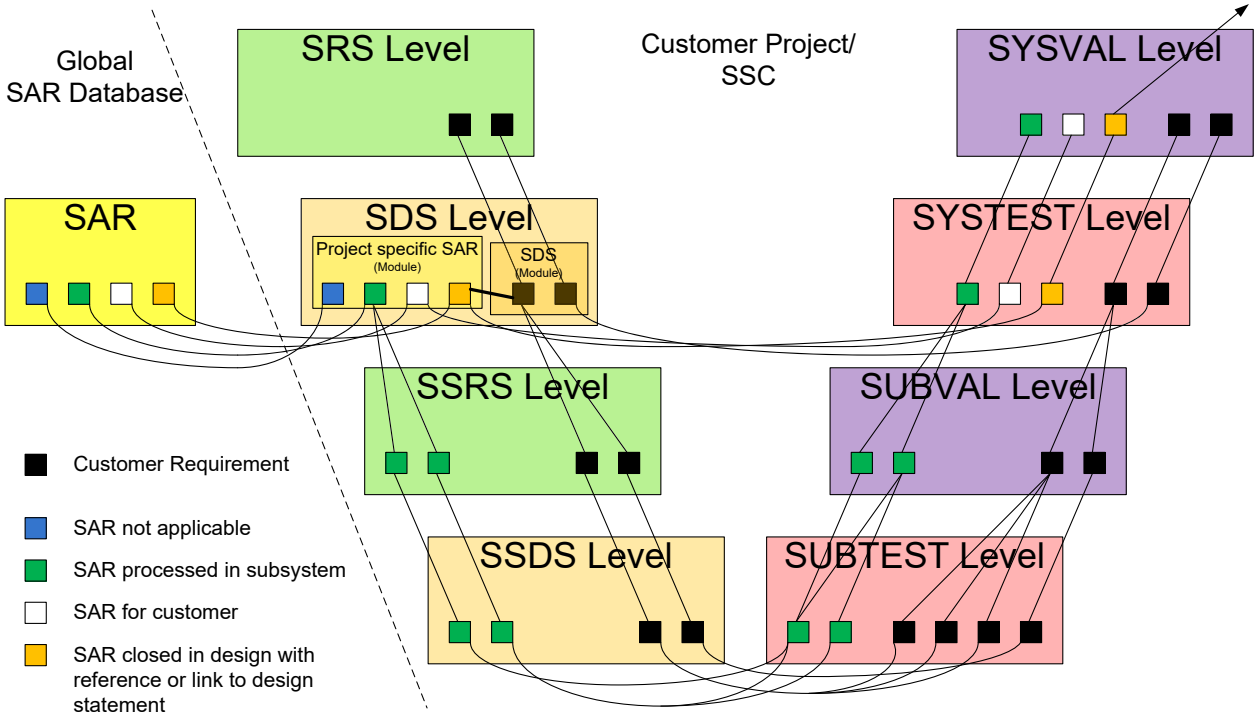

Figure 11 Flow of Application Rules in the V Model of the Project Execution

7.2 Processing in Customer and Generic SSC Projects

In the same way as requirements from the System Requirements Specification all application rules can be

- set progress to not applicable, if the rule is not applicable with a reason/reference,
- forwarded to one or more subsystem to be processed there,
- set progress to terminated with a reference to design document or forward to a design statement on SDS level with incoming link or
- set progress to terminated with a reference to a customer document for submission

according to the defined rules of [RM_Process_Manual_PE].

7.2.1 Not Applicable Application Rules

The following setting shall be set for an application rule if it is not applicable

- REQ Type: DEF
- REQ Progress: not applicable
- REQ Forwarding: -
- REQ Statement: <reason for status not applicable>
- REQ Pass Fail Criteria: -
- REQ Origin: <title of source AR document>
- REQ Safety_related: <yes, no> according to AR source module setting
- REQ RAM_related: <yes, no> according to AR source module setting

7.2.2 Forwarded Application Rules

The following setting shall be set for an application rule if it shall be forwarded to subsystem(s)

- REQ Type: DEF
- REQ Progress: forwarded
- REQ Forwarding: <Subsystem_1>, …
- REQ Statement: optional: <reason for assignment to subsystem(s)>
- REQ Pass Fail Criteria: <definition of acceptance criteria e.g. integration test>
- REQ Origin: <title of source AR document>
- REQ Safety_related: <yes, no> according to AR source module setting
- REQ RAM_related: <yes, no> according to AR source module setting

7.2.3 Closed Application Rules

The following setting shall be set for an application rule if it shall be closed on system design level with a design statement

- REQ Type: DEF
- REQ Progress: terminated / forwarded (with link reakizes from SDS)
- REQ Forwarding: -
- REQ Statement: <reference to design or other project document> (if not linked link realizes from SDS)
- REQ Pass Fail Criteria: <definition of review method for reference document>
- REQ Origin: <title of source AR document>
- REQ Safety_related: <yes, no> according to AR source module setting
- REQ RAM_related: <yes, no> according to AR source module setting

7.2.4 Application Rules to be forwarded to Customer

The following setting shall be set for an application rule if it shall be forwarded to the customer with a reference to an instruction/rule document for the customer.

- REQ Type: DEF
- REQ Progress: terminated
- REQ Forwarding: -
- REQ Statement: <ambiguous reference to document for the customer>
- REQ Pass Fail Criteria: <definition of review method for reference document>
- REQ Origin: <title of source AR document>
- REQ Safety_related: <yes, no> according to AR source module setting
- REQ RAM_related: <yes, no> according to AR source module setting

7.3 Processing in Development Projects

As requirements from the SyReqSpec all SARs can be

- set to not applicable, if the rule is not applicable with a reason/reference,
- forwarded to one or more forwarding target(s) to be processed there,
- terminated with a realize reference from a realization object
- terminated with a reference to a customer document for submission

The processing of the requirements follows the definition in [GL_RM_RD].

7.4 Verification and Validation of Application Rules

According to the rules of project requirements processing [RM_Process_Manual_PE] respectively according to the target in development projects [GL_RM_RD] all applicable application rules shall be tracked in the RM system during project execution phase. On each level a verification task like integration or tests shall be executed as well as all requirements and their evidences shall be validated.

System integration tests have to cover all application rules which are split into more than one subsystem requirement in the design phase.

For safety related application rules the same proceeding is adopted as for safety related requirements derived from the contract. The preparation of safety case documents and the task of safety assessment have to be executed according to the project needs (e.g. category, SIL, etc.).

7.5 Update of Application Rules to New Version

Each single application rule is linked to its predecessor in the previous version of the application rules. New application rules are of course not linked to any previous version. Depending on the creation of the central application rule module the changes between versions can be displayed via comparing two linked modules or two baselines of the same module.

The update procedure works for development projects as well as customer or generic SSC projects. In the following the figures display the update steps for a customer/generic SSC project for the product SIMIS IS.

7.5.1 Update of Application Rules from two Different Modules

In modules transferred from documents (each version of a document has been transferred to a separate module) the application rules are linked to their previous version via the link **reference_of** (former: **42_reference_of**). Therefore, the update procedure for the change to a new product version uses the existing links to identify new (no link to the module of the previous version), changed and omitted (no link from the module from the new version) rules. These changes must be integrated in the project module via the link relation **41_derived** and the links must be redirected to the application rules of the new product version.

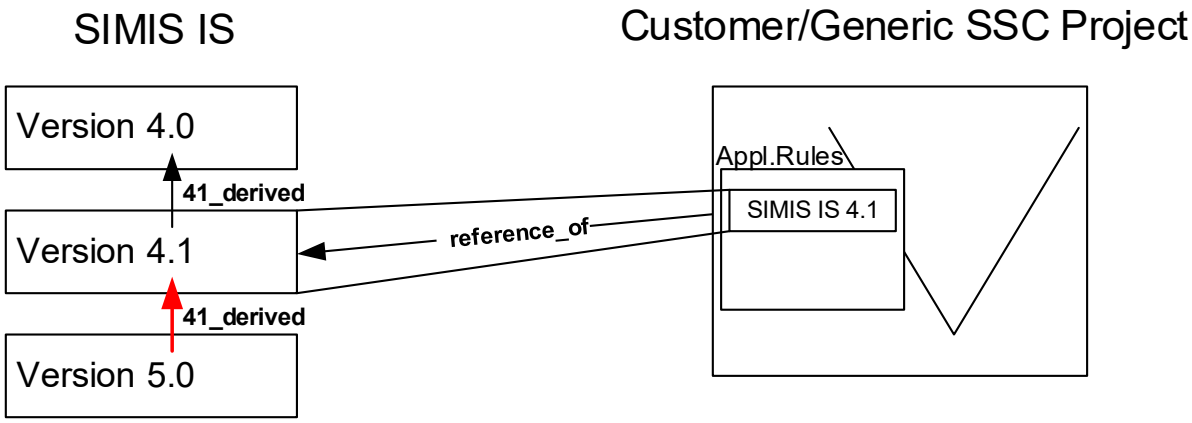

Figure 12 Examination of Traces of Type 41_derived between Versions

Figure 12 shows the link relation (41_derived, see section 9.2) between the version which will be used in the project (Version 5.0) and the current version (Version 4.0).

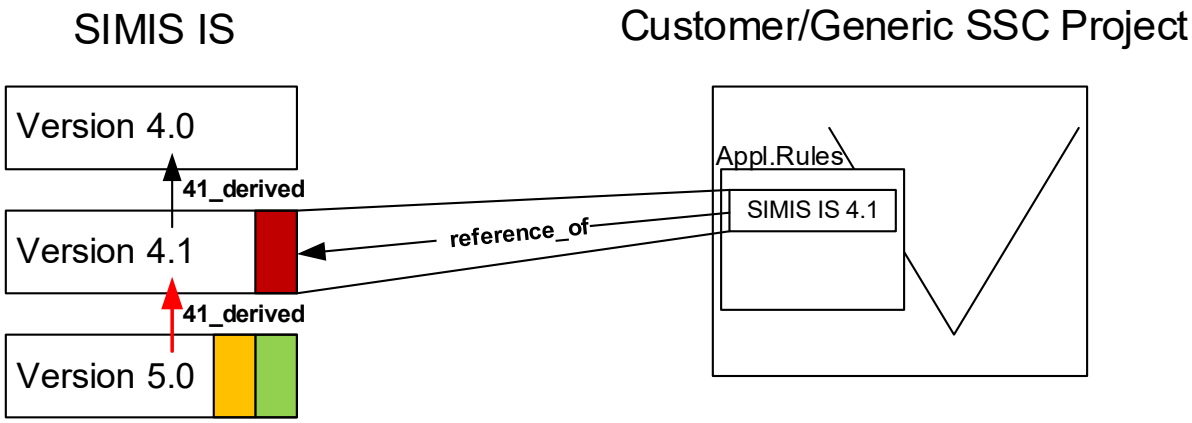

Figure 13 Identification of New, Changed and Omitted Application Rules

After identification of all changes the following steps must be done by the RMiP or via usage of the administrative tool (see chapter 8) in the future:

1. Define a new baseline of the SAR module in the project.
2. Redirect links of non-changed objects from SAR module of the project to the object in the module of the new version via the links **reference_of** and **41_derived** (see also section 9.2 for link definition)
3. Redirect links of all changed objects from SAR module of the project to the object in the module of the new version via the links **reference_of** and **41_derived** and update of all objects in the SAR module in the project (yellow box in Figure 13).
4. Delete all objects in the SAR module which are not traced via the links **reference_of** and **41_derived** to the module of the new version (red box in Figure 13).
5. Transfer all new rules (green box in Figure 13) from the module of the new version to the SAR module.

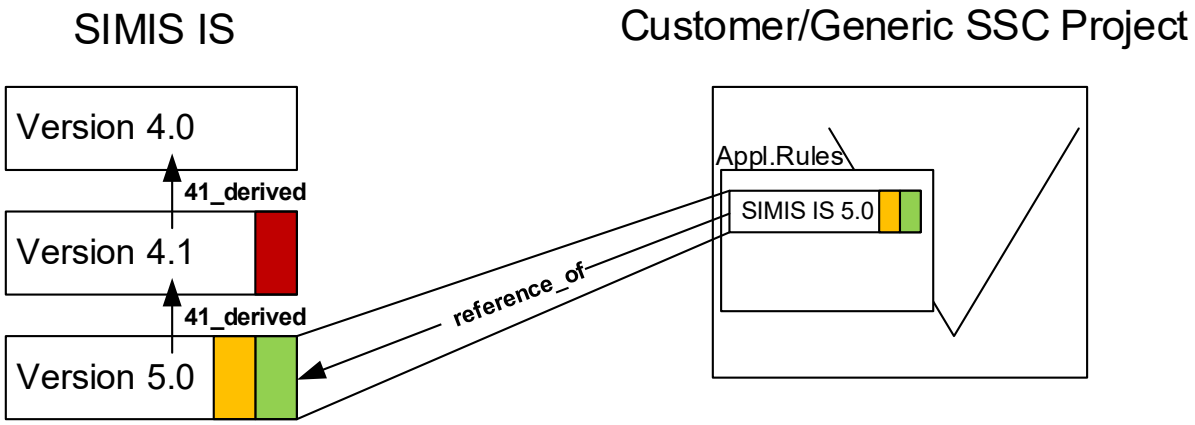

Figure 14 Complete Application Rules of the New Product Version

After execution of all steps the SAR module of the project contains the same objects of the module in the central location storage, and these are linked via the link **reference_of**. So the user can access all information to the application rules in the central module.

7.5.2 Update of Application Rules between two Baselines

The steps for updating a SAR module of a project from existing baselines are the same as with two different modules (see previous section 7.5.1). The information about the changes between the former and the new version need not be determined via the link **41_derived** but via a baseline comparison. Supported by an administrative tool (see chapter 8) the update steps will be executed without regard to different modules in the central storage location. Figure 15 shows the result after the update from two baselines of a module (e.g. update from Version 7.2 to 8.0).

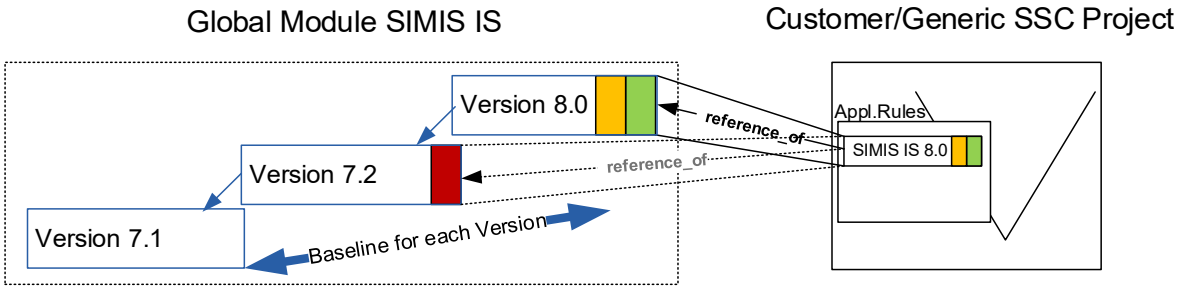

Figure 15 Update of Application Rules after Baseline Comparison

8 Tool Support for administrative Tasks

The transfer and the provision of solutions must be managed by the requirements manager. An administration tool for the current RM system DOORS will support the RMiP to execute the miscellaneous tasks. These tool functions are specified in the [Administration Tool for Handling Application Rules in DOORS]. They comprehend the following actions:

- Transfer of application rule versions of all products to the customer project or generic SSC project
- Provision of internal solution (e.g. hints of product development) to the project
- Collection and storing of all project-specific solutions
- Provision of project specific solutions of all projects which have used a version of the product (selectable)
- Preparation of baselines and new versions for definition of new version of the application rules
- Upgrade or downgrade of a application rule version according the tracing between the versions in the central database
- Displaying all changes (new, deleted, text changed, identifier changed) between versions before executing the up-/downgrade

9 Preference for Handling of Application Rules

9.1 Templates and Checklists

The template for the definition of application rules are defined in the document [ApplicationRule_Definition]. It lists the mandatory attributes and an export of the RM system shall fill all values in this format. Additional attributes are allowed for the RM system.

9.2 Link Definitions for Using Application Rules

Links define the relationship between objects in the same or in different modules. For the handling of application rules links are defined between:

1. different versions of application rules (part of and defined in central application rules database): **41_derived**
2. an application rule in the consuming project to its source in the central application rules database (link is part of and defined in customer project): **42_reference_of**
3. an application rule and its documented solution imported from a customer project or generic SSC project (link is part of and defined in central application rules database): **43_solved_with**

Figure 12 shows the different link types which have to be defined in the RM system. The RMiP must configure restrictions for the link setting or, if not available, to survey the correct link setting with tool support (consistency check).

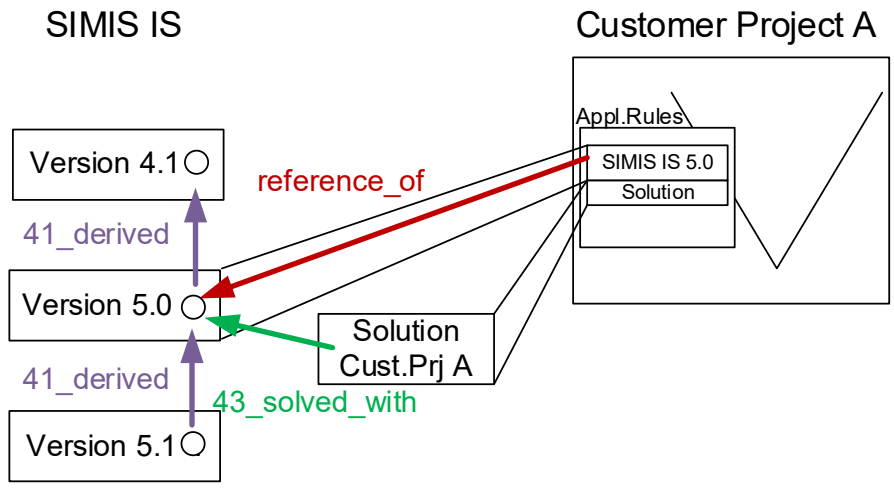

Figure 16 Link Definition for Handling Application Rules

The links shall be stored – tool dependent – next to their source objects:

1. **41_derived**: in the central application rules database in the same folder of the product
2. **reference_of (former: 42_reference_of)**: in the customer project or generic SSC project in the project specific link folder

3. **43_solved_with**: in the central application rules database in the same folder of the product

In the current RM system of SMO RI the folder structure is defined as follows:

Figure 17 Link Modules Folder for Application Rules in DOORS

9.3 Glossary

Abbreviation	Name	Explanation
AR	Application Rule (or in German: Anwendungsregel)	
BE	Business Excellence	BU of RI
BS	Business Segment	
BU	Business Unit	
BPM	Bid (Project) Manager	
DOORS	Dynamic Object Oriented Requirements System	RM system of IBM Rational
EDM	Electronic Document Management	
FC	Feature Catalog	
GUIDE	Globally unified development process	RI product development process.
HTZ	HerstellerTeileKennzeichen = Identifikationsziffer der technischen Zeichnung	Identifier of document management system of SMO RI Switzerland
IEEE	Institute of Electrical and Electronics Engineers	
ILC	Interlocking	
PLM	Product Lifecycle Management	
PM	Project Manager	
RE	Requirements Engineering/Requirements Engineer	
REQ	Requirement	Prefix of DOORS attribute
RI	Rail Infrastructure	Business unit of SMO
RM	Requirements Management/Requirements Manager	
RMiP	Requirements Manager in a project	
SAR	Safety Application Rule (or in German: Sicherheitsbezogene Anwendungsregel)	

SIMIS IS	Sicheres Mikroprozessor System – Interlocking Small	Interlocking system of Siemens
SM	System Management	
SMiP	System Manager in a project	
SMO	Siemens Mobility	Company of Siemens
SSC	Standard System Configuration	
STD	Standard	Prefix of DOORS attribute

Table 5 Abbreviations

SIEMENS

10 Appendix A IT Tool specific Manuals

IT Tool specific Manuals and Presentations

Text mark	Reference
[Processing of Project Execution with DOORS]	Introduction for users to project processing (realization phase) with DOORS in accordance with SMO RI's RM@RI process, A6Z00026507655
[Administration Tool for Handling Application Rules in DOORS]	Requirements Specification of Administration Tool for Handling Application Rules in DOORS, A6Z00034474887

Table 6 IT tool specific manuals

11 Appendix B Reference documents

Reference Documents

Text mark	Reference
[ApplicationRule_Definition]	Requirements: Application Rules/ Operating Conditions, A6Z00034445689
[Guideline_Requirements]	Leitfaden Requirements Definition; Notations- und Methodenratgeber, A6Z08110430427
[RM_Process_Manual_PE]	RM@RI Process Manual - Project Execution, A6Z00020793861
[RI_AR_Standard_Attributes]	List of Attributes for Application Rules; Requirements Management, A6Z00034488280
[RI_RM_Process_in_BIC]	RM process definition within SMO RI in BIC Cloud, https://siemens-mobility.biccloud.com/client/index.html#/view/tenant/15614856-a90c-4032-8d6d-acb3efba4b8a/repository/bd218ffc-ec4b-4b06-9d8e-ae83ae3ae0ad/stage/published/diagrams/d223b050-029f-11e8-0b92-00505682811f?forceReload=false
[CMMI]	Capability Maturity Model Integration Software Engineering Institute Carnegie Mellon University Pittsburgh, Version 1.3
[Requirements_Properties]	M. Mannion, B. Keepence: SMART Requirements Software Engineering Research Group, Napier University Edinburgh, 1995

Table 7 Reference documents

Standards and guidelines

Text mark	Reference
[EN_50129]	European Norm EN 50129: Railway applications - Communications, signaling and processing systems - Safety related electronic systems for signaling, February 2003
[GL_RM_RD]	Guideline for Requirements Management of the global SMO RI R&D, A6Z00002541903
[GUIDE]	Development process of the global SMO RI R&D, A6Z00035038061
[SMO_PM_Guideline]	Guideline Project Management at Mobility Worldwide, Reg.-Nr. 127_C_15, Version 3.1

Table 8 Standards and guidelines

12 Lists

12.1 List of figures

12.2 List of tables