

prostep ivip/VDA



ReqIF Benchmark 2021/2022

Short Report

Version 1.0, 23.12.2022

Status: final



Abstract

Requirements Management (RM) has been established to ensure seamless specifications along the product creation process. To manage complex specification processes and requirements dependencies companies introduced requirements management systems. The generic „Requirement Interchange Format (RIF)“ was created to enable the exchange of information across different requirements management systems.

In summer 2008 the prostep ivip association initiated the project group IntRIF to increase the acceptance and application of RIF by transferring the recommendation into an international standard. With the successful standardization in April 2011 OMG ReqIF 1.0.1 has been published as the official successor of RIF.

Two project groups are currently working on the enhancement of the format and its application. In 2011 the ReqIF Implementor Forum was established for realizing a strong technological basis. In 2016 the community of user representatives then consequently made the next step: Specifying relevant use cases for ReqIF application in industry.

To evaluate the feasibility of requirement data exchange with ReqIF, benchmarks are conducted, the very first in 2018. The benchmarks were well received by the users and implementers, as they provided valuable information for the usage and further development of requirement management tools. In this fourth benchmark, the tested scenario is a customer/supplier data exchange with comments on supplier and updates on customer side. With 6 participating software providers, there were a total of 9 participating RM-systems and connectors that were tested in 72 different tool combinations. The benchmark was run at prostep ivip site with support of the participating software providers. The criteria and test data were defined by the ReqIF Workflow Forum.

Disclaimer

This document is a prostep ivip Documentation (PSI Documentation). Those are freely available for all prostep ivip e.V. members and VDA members. Anyone using these recommendations is responsible for ensuring that they are used correctly.

This PSI Documentation gives due consideration to the prevailing state-of-the-art at the time of publication. Anyone using PSI Documentations must assume responsibility for his or her actions and acts at their own risk. The prostep ivip Association and the parties involved in drawing up the PSI Documentation assume no liability whatsoever.

We request that anyone encountering an error or the possibility of an incorrect interpretation when using the PSI Documentations contact the prostep ivip Association (psi-issues@prostep.org) immediately so that any errors can be rectified.

Copyright

- I. All rights on this PSI Documentation, in particular the copyright rights of use and sale such as the right to duplicate, distribute or publish the Documentation remain exclusively with the prostep ivip Association and its members.
- II. The PSI Documentation may be duplicated and distributed unchanged, for instance for use in the context of creating software or services.
- III. It is not permitted to change or edit this PSI Documentation.
- IV. A suitable notice indicating the copyright owner and the restrictions on use must always appear.

Contents

Table of Contents

1 Introduction.....	1
2 Approach	1
2.1 Four Steps	1
2.2 Scenario: Data Exchange with Changes and Linked Artefacts (PING-PONG-PING).....	3
2.3 Participants	4
2.4 Reference File	7
2.5 Test Criteria.....	11
2.6 Testing.....	13
2.7 Documentation	14
3 Results	14
3.1 Overview	14
3.1.1 Setup	14
3.1.2 Overview Ping 1	15
3.1.3 Overview Pong	21
3.1.4 Overview Ping 2	26
4 Summary and Outlook.....	34
5 Acknowledgements	34

Figures

Figure 1: Process and Actors	2
Figure 2: Ping-Pong-Ping Scenario.....	3
Figure 3: Ping-Pong-Ping Scenario step by step	4
Figure 4: Formatted text in reference file	7
Figure 5: Links between modules A and B	7
Figure 6: Links between modules B and C.....	8
Figure 7: Links between modules A and B before Ping 2	8
Figure 8: Document files embedded in ReqIF module	9
Figure 9: Image files embedded in ReqIF module	9
Figure 10: Adding and removing requirements	10
Figure 11: Changes of requirement's text, attributes and type definition	10
Figure 12: Changes in document structure	10
Figure 13: Results for completeness of reference import	14

Figure 14: Results for validity of first export to ReqIF.....	15
Figure 15: Results for completeness of first import to supplier RM-system	16
Figure 16: Results for exchange of font styles	16
Figure 17: Results for exchange of simple indentations	17
Figure 18: Results for exchange of simple numbered lists.....	17
Figure 19: Results for exchange of simple bullet points	18
Figure 20: Results for exchange of simple tables.....	18
Figure 21: Results for exchange of embedded document files (excl. Visio)	19
Figure 22: Results for exchange of Visio files	19
Figure 23: Results for exchange of embedded image files (excl. png).....	20
Figure 24: Results for exchange of png files.....	20
Figure 25: Results for exchange of requirement links.....	21
Figure 26: Results for validity of supplier export to ReqIF	22
Figure 27: Results for changes indicated on import to customer RM-system.....	22
Figure 28: Results for changes in supplier status	23
Figure 29: Results for changes in supplier comment	23
Figure 30: Results for image attached in supplier comment	24
Figure 31: Results for capability to import selected attributes to customer RM-system.....	24
Figure 32: Results for capability to export selected attributes from supplier RM-system	25
Figure 33: Results for no further changes after re-import to customer RM-system	25
Figure 34: Results for validity of second export to ReqIF.....	26
Figure 35: Results for change of requirement text	27
Figure 36: Results for change of attribute values	27
Figure 37: Results for change of attribute values to empty values.....	28
Figure 38: Results for change of enumeration definition of attribute type.....	28
Figure 39: Results for exchange of added requirement.....	29
Figure 40: Results for exchange of missing requirement.....	29
Figure 41: Results for exchange of changed document structure	30
Figure 42: Results for exchange of updated links	30
Figure 43: Results for changes indicated on import to supplier RM-system	31
Figure 44: Results for no further changes after re-import to supplier RM-system	31
Figure 45: Results for changes in customer status.....	32
Figure 46: Results for changes in customer comment.....	32
Figure 47: Results for exchange of simple numbered lists (Ping 2)	33
Figure 48: Results for exchange of simple tables (Ping 2).....	33

Tables

Table 1: Tested software4
Table 2: Test case matrix6
Table 3: Test Criteria11

1 Introduction

Requirements Management has been established to ensure seamless specifications along the product creation process. To manage complex specification processes and requirements dependencies companies introduced requirements management systems (RM-systems).

To support a proper requirements exchange between partners using different tools, the project group "Simulation and Tools" of the HIS (Hersteller Initiative Software) specified the generic „Requirement Interchange Format (RIF)".

In summer 2008 the prostep ivip association initiated the project group IntRIF to increase the acceptance and application of RIF by transferring the recommendation into an international standard. With the successful standardization in April 2011 OMG ReqIF 1.0.1 has been published as the official successor of RIF.

prostep ivip established two project groups to further drive the ReqIF format:

The goal of the ReqIF Implementor Forum (ReqIF-IF) is to ensure interoperability between different ReqIF-based implementations. Therefore, the ReqIF-IF works very tight together with the newly established ReqIF Workflow Forum (ReqIF-WF).

In 2016, the community of relevant user representatives consequently made the next step: Specifying relevant use cases for ReqIF application in industry.

Thus, the major aim of the prostep ivip / VDA ReqIF-WF is to specify use cases as well as reference processes (customer-customer, customer-supplier etc.) and, related to this, deriving process requirements and test cases. The work is performed in close collaboration with the ReqIF Implementor Forum.

To evaluate the feasibility of requirement data exchange with ReqIF, benchmarks are conducted, the very first in 2018. The benchmarks were well received by the users and implementers, as they provided valuable information for the usage and further development of requirement management tools.

In this fourth benchmark, the tested scenario is a customer/supplier data exchange with comments on supplier and updates on customer side. With 6 participating software providers, there were a total of 9 participating RM-systems and connectors that were tested in 72 different tool combinations. The benchmark was run at prostep ivip site with support of the participating software providers. The criteria and test data were defined by the ReqIF Workflow Forum.

Goal of the benchmarks is a neutral evaluation of the current capabilities in requirement data exchange with ReqIF. Additionally, issues that require further development of either the format itself or of the tested software tools will be identified and addressed.

2 Approach

The following sections describe the basic conditions for the benchmark.

2.1 Four Steps

Based on lessons learned from previous benchmarks, the ReqIF Workflow and ReqIF Implementor Forum agreed on the following four-step approach:

1. The ReqIF Workflow Forum clarified the target intent for the benchmark and provided details on the expected outcome.
2. The software providers made proposals for the ReqIF file scope, configuration settings and evaluation approach which in their eyes would best fit the requirements.

3. A proof of concept / test run for the benchmark was conducted, using agreed-on settings and test files, with close involvement of the software providers.
4. After the successful test run, the actual benchmark was conducted.

Figure 1 shows which tasks were performed by the involved actors during the benchmark.

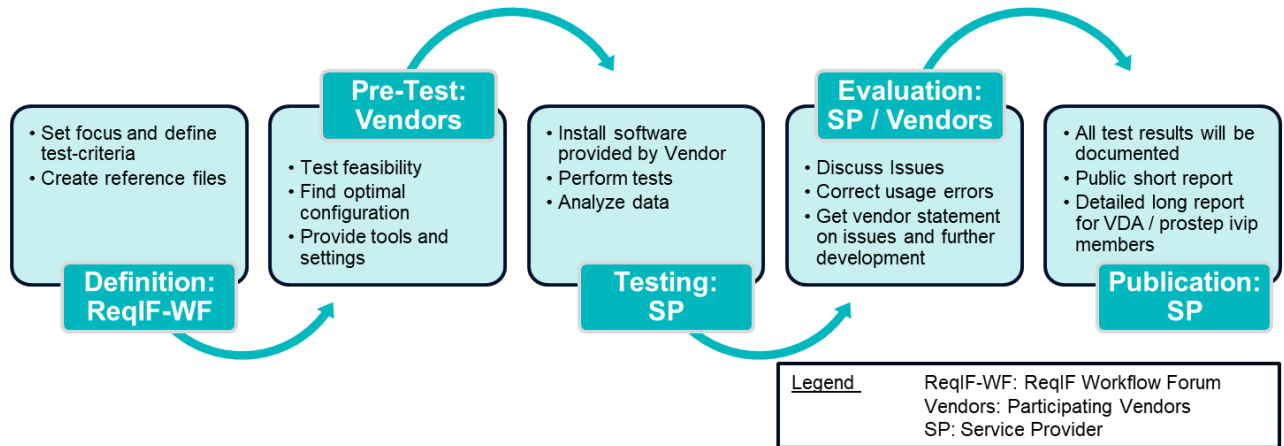


Figure 1: Process and Actors

The involved actors are the following:

- The prostep ivip ReqIF Workflow Forum
- The participating software providers (vendors) from the prostep ivip ReqIF Implementor Forum
- PROSTEP AG (as service provider, SP)

The actions performed during the different steps were:

1. In the first step, the definition phase, ReqIF Workflow Forum members set the focus of each benchmark and defined the test criteria. Also, a set of ReqIF files was chosen as the initial input for the benchmark tests.
2. The second step was the Pre-Test, which was conducted by the participating ReqIF application vendors. In this phase, they tested the feasibility of given test files and test criteria. They hereby had to find and optimize their tool configuration to achieve best results. These resulting configuration settings were finally provided for the benchmark testing.
3. In the third step, the benchmark testing was conducted by PROSTEP. Software made available by the vendors was installed, tests were performed, and results were analyzed.
4. In the fourth step, the preliminary results were discussed with vendors to correct usage errors, to get statements regarding further development of the affected software and to resolve identified issues.

All results will be published in a publicly available short report and in a detailed long report that is available for all prostep ivip and VDA members.

2.2 Scenario: Data Exchange with Changes and Linked Artefacts (PING-PONG-PING)

This scenario reflects a requirements exchange process with updates on the exchanged data on customer side and comments on the customer and supplier side. The scenario consists of three test phases: Ping 1, Pong and Ping 2 (see Figure 2). The exchanged data package consists of three modules which are linked to each other.

The phase "Ping 1", which describes a data exchange between a customer and supplier, deals with the initial export of a ReqIF package with 2 specifications from one RM-system (the customer RM-system) and the subsequent import of this package into another RM-system (the supplier RM-system), in which criteria for the quality of the data exchange, such as the transfer of embedded files/images, of formatted text or of links between requirements, were then evaluated.

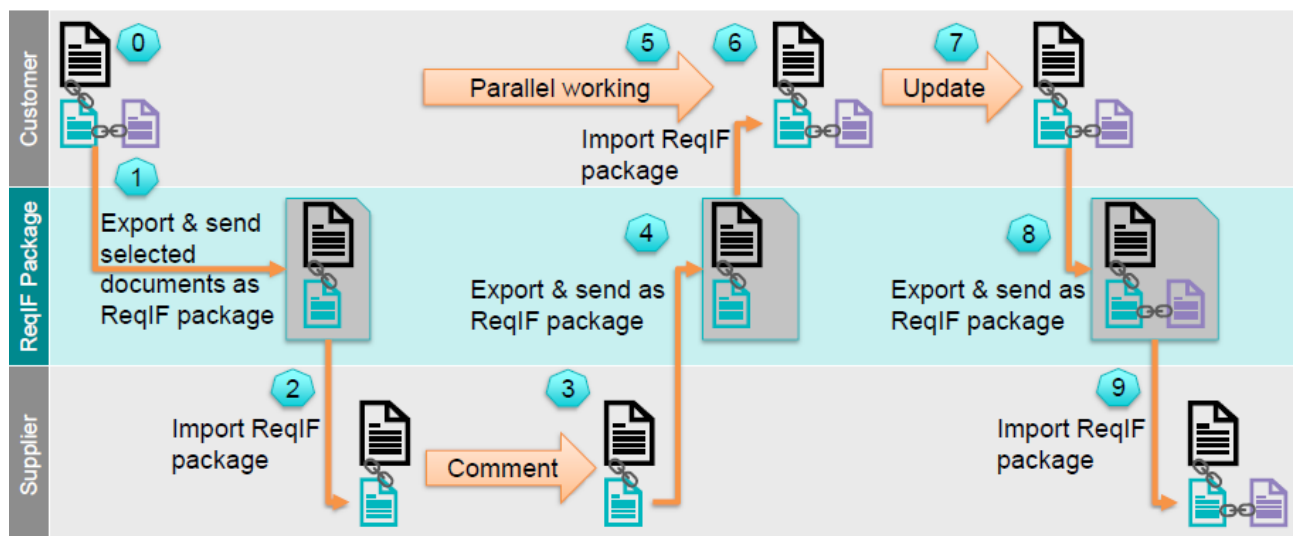


Figure 2: Ping-Pong-Ping Scenario

After Ping 1 the test phase Pong follows where the imported requirements are evaluated in the supplier RM-system by using the supplier attributes ("ReqIF-WF.SupplierStatus" and "ReqIF-WF.SupplierComment"). The supplier creates an export of the supplier attributes of all modules which is then being imported again by the customer after the customer changes content and links in one module in a parallel working scenario. After the import the modules are checked according to the benchmark criteria, such as e.g. the validity of the exported ReqIF packages, if changes are indicated on import or the capability of editing and exchanging supplier attributes.

For Ping 2 the customer analyzes the supplier evaluation and adds "ReqIF-WF.CustomerStatus" and "ReqIF-WF.CustomerComment" attribute values. The customer creates a second set of modules now including all three modules of the specification and performs an export which is then again imported by the supplier. The import is checked according to the benchmark criteria, such as e.g. the validity of the exported ReqIF packages, structural changes of module content or the capability of editing and exchanging customer attributes.

Figure 3 shows a brief step by step description of the three test phases of the benchmark scenario.

Ping-Pong-Ping (with parallel working) Step by Step

Setup:

0. Reference ReqIF packages are imported to the customer systems
 - Modules in customer systems are checked and modified if necessary

1st Ping:

1. Customer exports first set of modules (A, B) of the specification to ReqIF
 - Exported ReqIF-files are checked
2. Supplier imports the first set of modules of the specification
 - Modules are checked in supplier systems according to benchmark criteria

Pong:

3. Supplier evaluates requirements in the Supplier Attributes
 - Supplier Status and Comment are added
4. Supplier creates an ReqIF-export of the Supplier Attributes of all modules
 - Exported ReqIF-files are checked
5. Customer changes content and links in module A of the specification (parallel working scenario)

6. Customer imports the Supplier Attributes
 - Modules are checked in customer systems according to benchmark criteria

2nd Ping:

7. Customer analyzes the Supplier evaluation
 - Customer Status and Comment are added
 - Customer creates a second set of modules (A, B, C) of the specification
8. Customer exports second set of modules of the specification to ReqIF
 - Exported ReqIF-files are checked
9. Supplier imports the second set of modules
 - Modules are checked in supplier systems according to benchmark criteria

Figure 3: Ping-Pong-Ping Scenario step by step

In the benchmark tests, the exported ReqIF packages were checked for its validity against the ReqIF schema before the imports were started.

2.3 Participants

All members of the ReqIF Implementor Forum were asked to participate. Four RM-system vendors and two vendors for data exchange connectors participated in the benchmark:

- Asaro Systems Limited
- IBM
- REQUISIS GmbH
- Intland Software GmbH
- Siemens Industry Software GmbH
- :em engineering methods AG

The tested software is listed in Table 1. The test results are only valid for the versions given in the table. Issues detected during this benchmark may already be solved in versions released since the completion of the benchmark tests as issues are discussed with the vendors during the benchmark.

Table 1: Tested software

Vendor	Software	Version	Description
Asaro Systems	ReqIF for Active Workspace	Ping 1: 22.01.4453 Pong, Ping 2: 22.10.4884	ReqIF connector for Teamcenter Active Workspace
IBM	DOORS	9.7.2.3	RM-system

Vendor	Software	Version	Description
IBM	DOORS NG	7.0.2 + iFix005	RM-system
IBM	IBM Engineering Requirements Management DOORS Family - Data Exchange Add On (eXchange) + IBM DOORS	5.7	ReqIF connector for DOORS
REQUISIS	ReX	V2.16	ReqIF connector for DOORS
REQUISIS	ReqIF Manager	2.6.04.0 – 2.10.0.0	ReqIF connector for DOORS NG
Intland	codeBeamer	21.09-SP2 – 22.10-SP2	RM-system
Siemens	Polarion	21 R1 - 21 R2	RM-system
Siemens	Teamcenter Active Workspace (to be tested with Asaro ReqIF)	5.1.0	PLM System with requirements management functionalities
:em	ReqMan	V2.2021.2-b932ba7265	RM-system (Ping 1 participation only)

With these 9 software systems, 72 combinations for the data exchange are tested in Ping 1, as can be seen in the test case matrix (Table 2). For Pong and Ping 2 ReqMan by :em AG withdrew from the benchmark at the request of the vendor, resulting in a number of 56 system combinations.

As check tool, Asaro Systems ReqIF Q-Checker (version 20.07.4502) and Asaro Systems ReqIF Toolkit (version 22.04.795) were used. The Q-Checker checks the validity of the file, formatting, attribute and datatype definitions and missing or unreferenced files.

The participating software vendors provided software and licenses for the duration of the benchmark. The software was installed on a local machine at PROSTEP or available to PROSTEP as a cloud-based service.

Table 2: Test case matrix

Customer RM-system → Supplier RM-system ↓	ReqMan	ReqIF for Active Workspace + Teamcenter	DOORS (built in)	DOORS Next Generation	Data Exchange Add On + DOORS	codeBeamer	ReqIF Manager + DOORS NG	ReX + DOORS	Polarion
ReqMan	o	✓	✓	✓	✓	✓	✓	✓	✓
ReqIF for Active Workspace + Teamcenter	✓	o	✓	✓	✓	✓	✓	✓	✓
DOORS (built in)	✓	✓	o	✓	✓	✓	✓	✓	✓
DOORS Next Generation	✓	✓	✓	o	✓	✓	✓	✓	✓
Data Exchange Add On + DOORS	✓	✓	✓	✓	o	✓	✓	✓	✓
codeBeamer	✓	✓	✓	✓	✓	o	✓	✓	✓
ReqIF Manager + DOORS Next	✓	✓	✓	✓	✓	✓	o	✓	✓
ReX + DOORS	✓	✓	✓	✓	✓	✓	✓	o	✓
Polarion	✓	✓	✓	✓	✓	✓	✓	✓	o

2.4 Reference File

Before the benchmark tests, reference files were created. The ReqIF Workflow Forum members made sure that all relevant content is in the files, the ReqIF Implementor Forum members checked the created files for their validity. For every RM-system, a package with unique attribute identifiers was created. In the following, the files' relevant content for this benchmark is shown.

To test text formatting, chapters have been added to the reference file that represent aspects of formatted text, such as text formats, text indentations, bullet points, numbered lists and a simple table, as shown in Figure 4.

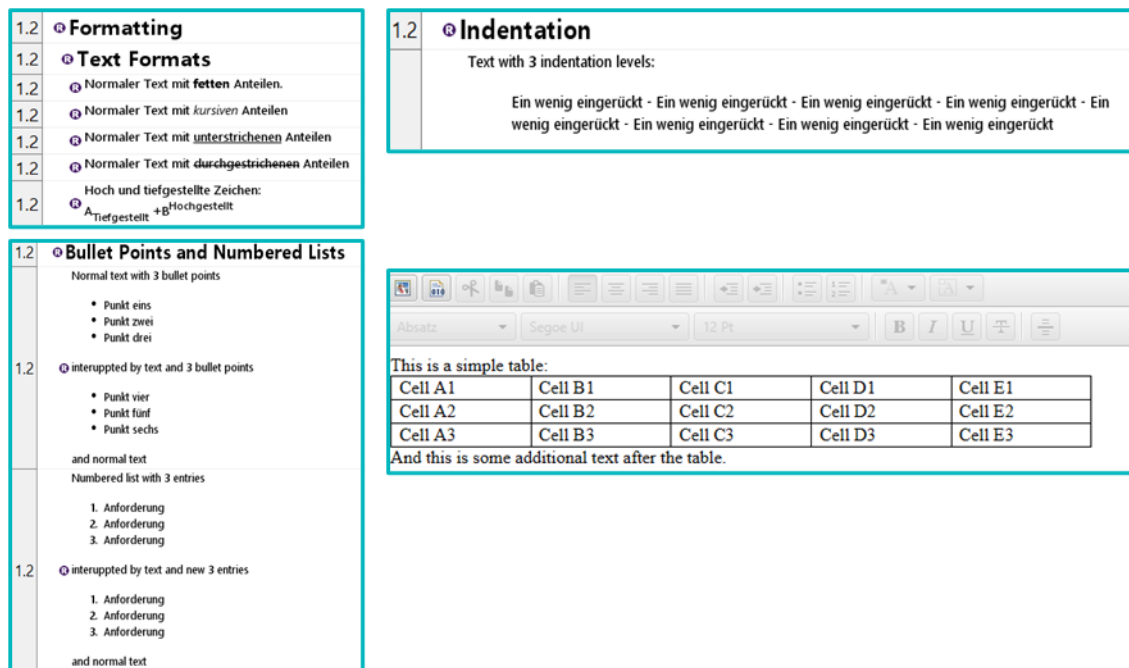


Figure 4: Formatted text in reference file

For testing of links, a chapter and two modules were added to the entire package of the reference file. The original file will be addressed as module A, the other two as module B and C. Figure 5 and Figure 6 show the way the modules are linked to each other.

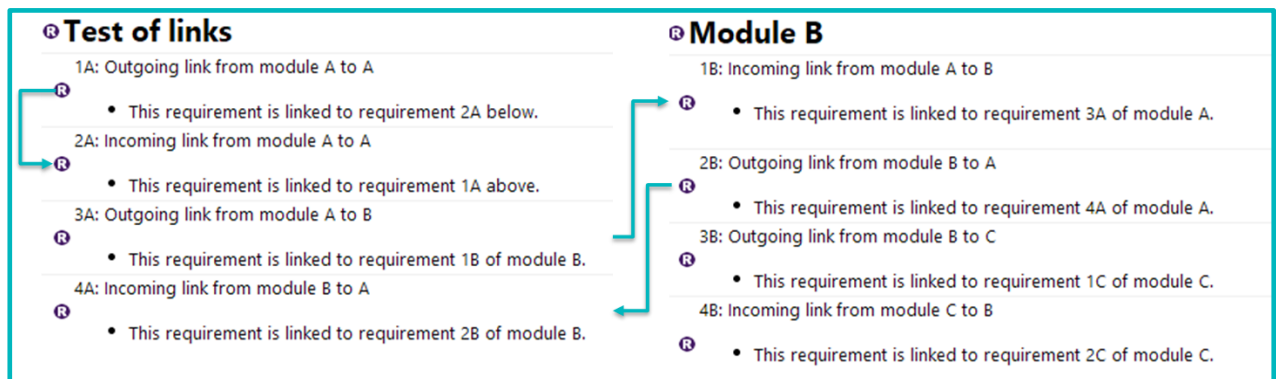


Figure 5: Links between modules A and B

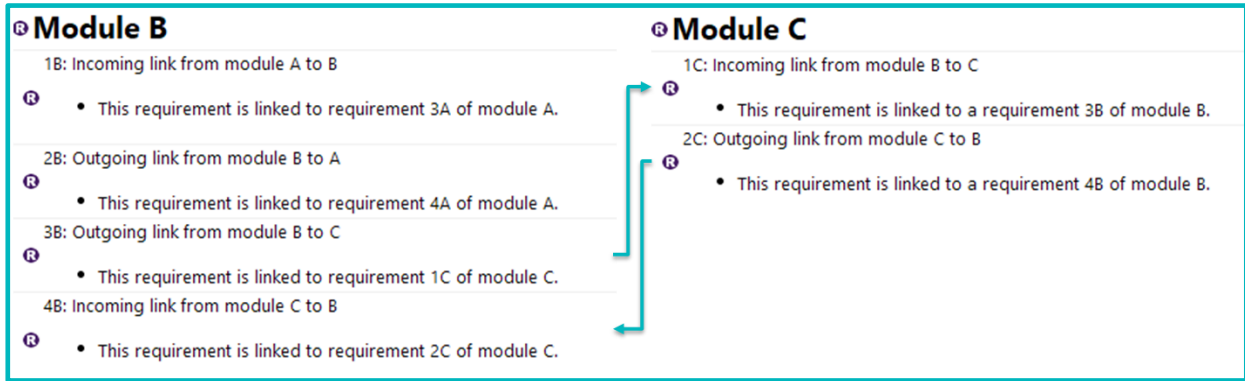


Figure 6: Links between modules B and C

Figure 7 shows the way requirements are linked after the update in the parallel working scenario by the customer before the second ping.

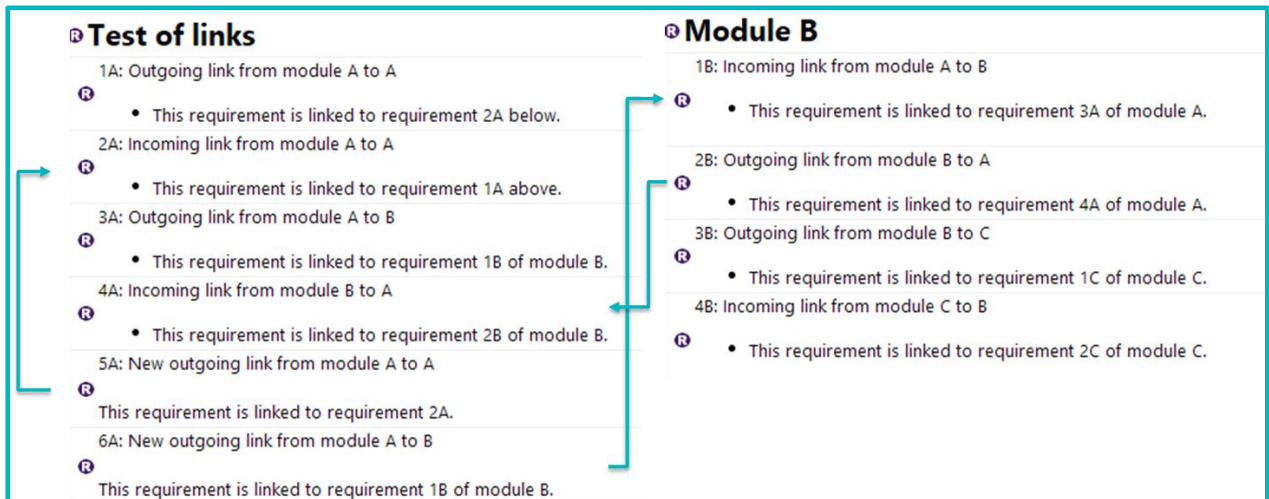


Figure 7: Links between modules A and B before Ping 2

The package contains docx, pptx, xlsx, pdf and vsdx files which will be referred to as document files further on. Figure 8 shows the way the files are depicted inside the module and a screenshot of the opened file.

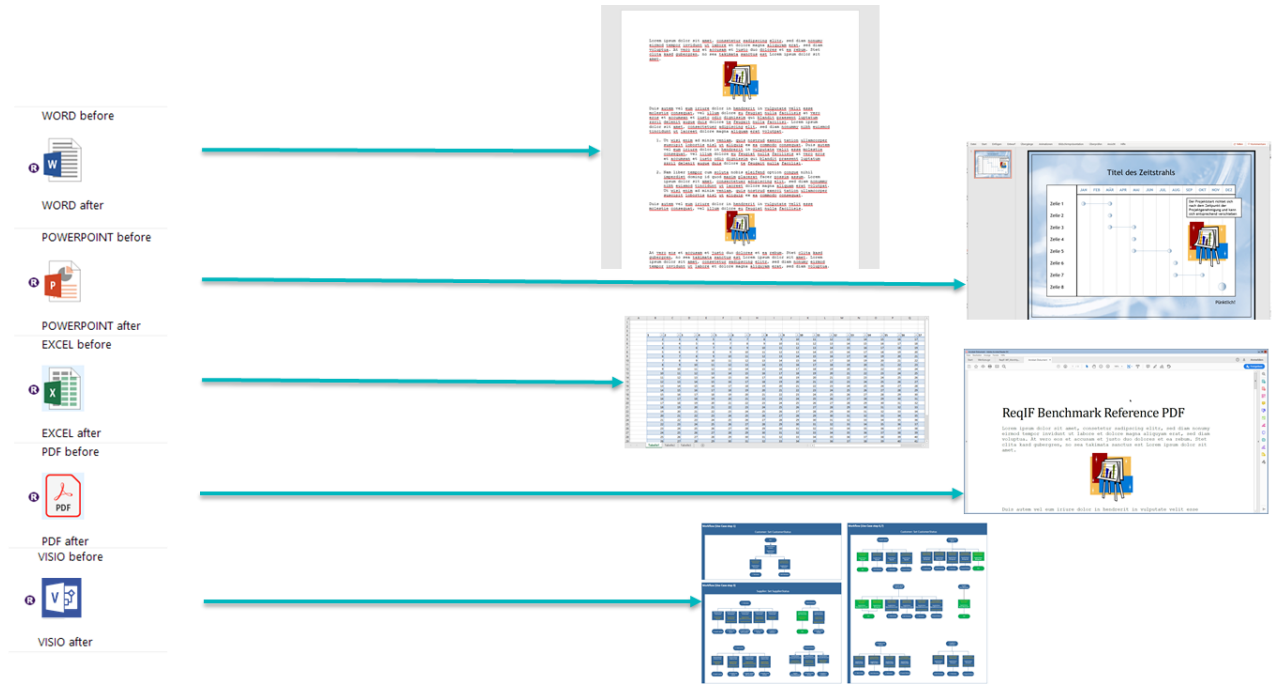


Figure 8: Document files embedded in ReqIF module

Different types of image files (JPG, GIF, EMF, BMP and PNG) are embedded in the module in the same way as the document files, as can be seen in Figure 9.

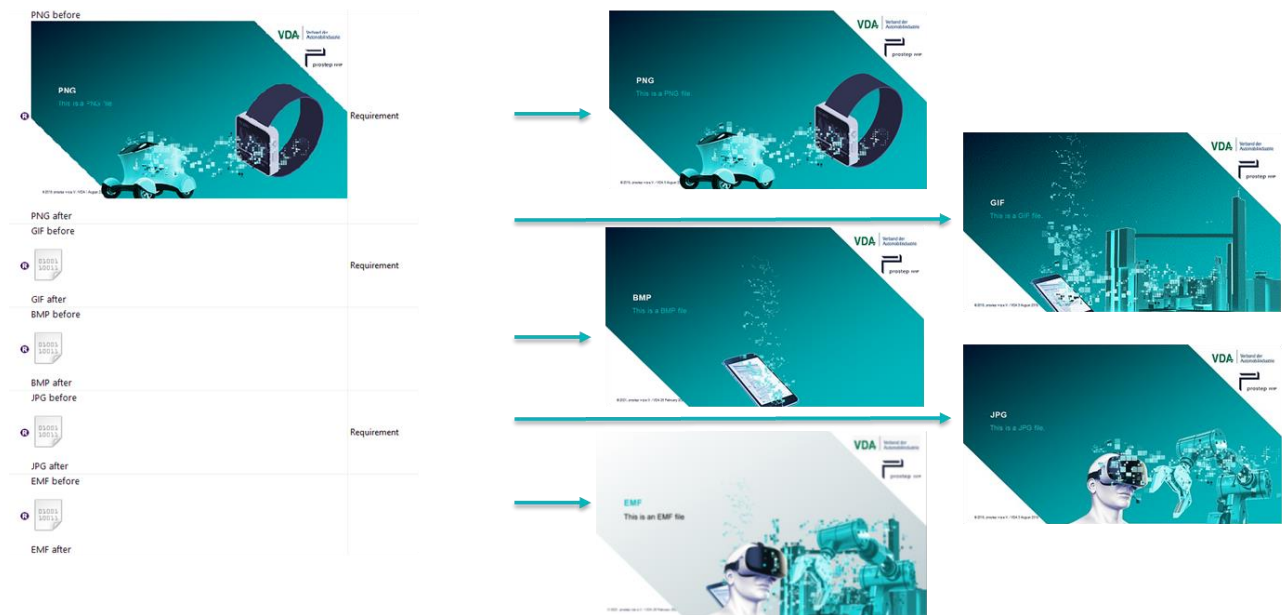


Figure 9: Image files embedded in ReqIF module

Requirements have been added to the original reference file used in the previous benchmark to make added and removed requirements easily recognizable to the user. Figure 10 shows the relevant part of the reference file before and after the changes of the parallel working scenario.

Structure before 1st ping (Step 1)

Main	ReqIF-WF.Custorm
Identification of new or removed requirements	
<ul style="list-style-type: none"> This is the first requirement of this section. This is the second requirement of this section <ul style="list-style-type: none"> Step 3: Remove or delete this requirement from export 	this requirement will be deleted
Hint: Important is, that this element is not part of the ReqIF file. Check ReqIF file for text <DELETION>.	
<ul style="list-style-type: none"> Step 4: The requirement must be identifiable as "not in the ReqIF file" 	
This is the third requirement of this section	
<ul style="list-style-type: none"> Step 3: Add a new requirement "This is a new requirement." after this requirement Step 4: The new requirement must be added behind this requirement 	a new requirement will be added

Structure before 2nd ping (Step 7)

Main	ReqIF-WF.Custorm	
Identification of new or removed requirements		
<ul style="list-style-type: none"> This is the first requirement of this section. This is the third requirement of this section <ul style="list-style-type: none"> Step 3: Add a new requirement "This is a new requirement." after this requirement Step 4: The new requirement must be added behind this requirement 	a new requirement will be added	
<ul style="list-style-type: none"> This is a new requirement. 		

Figure 10: Adding and removing requirements

For the changes of requirement text, attributes and attribute definition, dedicated requirements have been added as shown in Figure 11. These also contain information about the changes to be made to make them easily recognizable and traceable.

Main	ReqIF-WF.Custorm
Change of requirement's text and attributes	
This is the first element of this section, it has a sub-element:	
<ul style="list-style-type: none"> Step 1: no change of sub-element Step 3: change text of sub-element to "This is the second export (Step 3)." 	
This is the first export (Step 1).	
The attributes are set to defined values:	
<ul style="list-style-type: none"> Step 1: aBoolean="true"; aDate="27.02.2020..."; aEnum="Variant A"; integer="1111"; aReal=22.22; aString="Plaint text in a string type" Step 3: aBoolean="false"; aDate=<today>; aEnum="Variant C"; integer="3333"; aReal=44.44; aString="String type with plaint text" 	
The attributes are set to an <empty> value (if possible):	
<ul style="list-style-type: none"> Step 1: aBoolean="true"; aDate="27.02.2020..."; aEnum="Variant A"; integer="1111"; aReal=22.22; aString="Plaint text in a string type" Step 3: set all attributes "aXXXX" to an "empty" value 	
If a change is not possible give a comment in "CustomerComment".	

Main	ReqIF-WF.Custorm
Change of attribute type definition	
Change of enumeration type definition:	
<ul style="list-style-type: none"> Step 1: aEnum="Variant B" Step 3: Change "VariantType": delete "Variant B"; add "Variant D"; set aEnum="Variant D" 	
If a change is not possible give a comment in "CustomerComment".	
Hint: Perhaps the enum value must be changed before deletion of enum definition "Variant B"	

Figure 11: Changes of requirement's text, attributes and type definition

Additional chapters have been added to test changes in the document structure. These chapters will be reordered in the document before the second export (Ping 2) as Figure 12 shows.

Main	ReqIF-WF.Custorm
Change of document structure	
This chapter will be restructured in step 3: headings are numbered, the text elements are indicated with abc:	
<ul style="list-style-type: none"> Step 1: nothing Step 2: check structure and completeness Step 3: Bring all elements in the right order, e.g. 1, 1a, 1b, 1.1, 1.1a ... Step 4: Check correct structure and sequence 	
Help: See also sequence number 1..15 in "CustomerComment"	
Third Chapter 3	14
3a: First requirement of chapter 3	15
First Chapter 1	1
1a: First requirement of chapter 1	2
Sub-Chapter 1.1	4
1.1b: Second requirement of chapter 1.1	6
1.2b: Second requirement of chapter 1.2	9
Sub-Chapter 1.2	7
1.1a: First requirement of chapter 1.1	5
1.2a: First requirement of chapter 1.2	8
Second Chapter 2	10
2c: Third requirement of chapter 2	13
2b: Second requirement of chapter 2	12
2a: First requirement of chapter 2	11
1b: Second requirement of chapter 1 (behind 1.a)	3

Main	ReqIF-WF.Custorm
Change of document structure	
This chapter will be restructured in step 3: headings are numbered, the text elements are indicated with abc:	
<ul style="list-style-type: none"> Step 1: nothing Step 2: check structure and completeness Step 3: Bring all elements in the right order, e.g. 1, 1a, 1b, 1.1, 1.1a ... Step 4: Check correct structure and sequence 	
Help: See also sequence number 1..15 in "CustomerComment"	
First Chapter 1	1
1a: First requirement of chapter 1	2
1b: Second requirement of chapter 1 (behind 1.a)	3
Sub-Chapter 1.1	4
1.1a: First requirement of chapter 1.1	5
1.1b: Second requirement of chapter 1.1	6
Sub-Chapter 1.2	7
1.2a: First requirement of chapter 1.2	8
1.2b: Second requirement of chapter 1.2	9
Second Chapter 2	10
2a: First requirement of chapter 2	11
2b: Second requirement of chapter 2	12
2c: Third requirement of chapter 2	13
Third Chapter 3	14
3a: First requirement of chapter 3	15

Figure 12: Changes in document structure

2.5 Test Criteria

The test criteria and tolerances were defined by the ReqIF Workflow Forum. They are listed in the following Table 3.

Table 3: Test Criteria

Phase	Criterion	Description
Ping 1	Validity	All exported ReqIF files must be valid. Tests with invalid files will be continued with the results marked accordingly. For this criterion only “successful” or “failed” exists.
	Completeness	The entire requirement module exported from the customer RM-system must be imported to the supplier RM-system.
	Formatted text	A standard formatted text must be exchangeable. This includes font styles (bold , <i>italic</i> , <u>underlined</u> , strike-out , subscript, ^{superscript}), simple indentations, simple numbered lists and bullet points. Simple tables must be exchangeable, rows and columns must be recognizable. Minor deviations are acceptable (requirement: no comprehension problems).
	Links	Three modules are linked to each other. The links must be imported.
	Embedded files	Files must be accessible from the attribute where they were originally embedded. Also, they must occur at the same position in the attribute value's content. Pdf, docx, pptx, xlsx and vsdx files were used for the tests. Example: If in the customer's RM-system, the content is "Text1 <document> Text2", then in the supplier's RM-system, the order must be the same: First, Text1, then the document, then Text2.
	Embedded images	In the supplier's RM-system, images must occur at the same position in the attribute value's content as in the customer's RM-system. Example: If in the customer's RM-system, the content is "Text1 <image> Text2", then in the supplier's RM-system, the order must be the same: First, Text1, then the image, then Text2. Differences in image size and scaling are acceptable if there are no human visible errors in representation. JPG, GIF, EMF, BMP and PNG image files were used for the tests.
Pong	Validity	All exported ReqIF files must be valid. Tests with invalid files will be continued with the results marked accordingly. For this criterion only “successful” or “failed” exists.
	Changes indicated on import	The user must be able to easily identify changes when a requirement module is updated. The customer importing RM-system must be able to visualize changes before, during or after the import.
	Capability to edit and exchange supplier attributes with supplier RM-system	The value of the attribute “ReqIF-WF.SupplierStatus” is changed. The value of the attribute “ReqIF-WF.SupplierComment” is changed (incl. double line break) and an image is attached to this attribute. Changes in Supplier Status and Comment must be imported to the customer RM-system and the image must be accessible from the attribute.

Phase	Criterion	Description
	Customer attributes unchanged	Capability test to import selected attributes to customer RM-system (tool feature).
	Capability to export selected attributes from supplier RM-system	Capability test to export selected attributes from supplier RM-system (tool feature).
	No further changes after re-import to customer RM-system	No further changes to the modules should occur after re-import to customer RM-system. The customer RM-system should only change Supplier Status and Supplier Comment (including added files). Module changes of parallel working scenario should not be overwritten.
Ping 2	Validity	All exported ReqIF files must be valid. Tests with invalid files will be continued with the results marked accordingly. For this criterion only "successful" or "failed" exists.
	Change of requirement text	The change of a requirements text must be adopted on import to the supplier RM-system.
	Change of attribute values	The changes of attribute values (also to empty values) must be adopted on import to the supplier RM-system.
	Change of attribute type definition	The enumeration definition of an attribute type is changed (option added). The change must be adopted on import to the supplier RM-system.
	Structural changes (added requirement)	Added requirement must be imported and recognized.
	Structural changes (removed requirement)	Missing requirement must be identified on or after import to the supplier RM-system.
	Structural changes (order)	The position of objects inside the module is changed. The reordered objects must be adopted on import to the supplier RM-system.
	Update of links	New links must be imported (A-A, A-B). Links to a third module imported initially must be imported (B-C, C-B) to the supplier RM-system.
	Changes indicated on import	The user must be able to easily identify changes when a requirement module is updated. The supplier importing RM-system must be able to visualize changes before, during or after the import.
	No further changes after re-import to supplier RM-system	No further changes to the modules should occur after re-import to supplier RM-system. The supplier RM-system should maintain Supplier Status and Supplier Comment.
	Capability to edit and exchange customer attributes with	The value of the attribute "ReqIF-WF.CustomerStatus" and "ReqIF-WF.CustomerComment" is changed. Changes in Customer Status and Comment must be imported to the supplier RM-system.

Phase	Criterion	Description
	customer RM-system	
	Formatted text (Recheck open points Ping 1)	Simple numbered lists and simple tables must be exchangeable. Rows and columns must be recognizable. Minor deviations are acceptable (requirement: no comprehension problems). This criterion applies on all test cases with DOORS/DOORS Add-ons as customer)

The validity of the exported files was checked with the Asaro Systems ReqIF Q-Checker, all other criteria were checked within the requirements management system to which the ReqIF package was imported. The Asaro ReqIF Toolkit was used to check the content of the ReqIF files.

2.6 Testing

As a starting point the reference ReqIF packages were initially imported to the tested RM-systems and it was checked whether the content was imported as expected. If necessary, adaptations to the imported modules were made to match the needs of the testing procedure. The requirements of this packages are the set with which the following export and import tests were performed. In this case, the systems acted as the customer RM-systems.

The imported and revised requirement set was then exported to ReqIF. In Ping 1, only two of three modules were exported. This ReqIF export was checked with the Asaro Systems ReqIF Q-Checker for validity.

Afterwards, the exports were imported to the other tested RM-systems acting now as supplier RM-systems.

In the supplier RM-systems, the tester checked the modules according to the benchmark criteria e.g., the completeness of the modules, the correctness of the formatted text, the accessibility of documents and images and the links and attributes relevant for the benchmark tests (see Chapter 2.5). After this check Ping 1 is completed.

For the Pong phase the tester evaluated predefined requirements in the supplier attributes by adding supplier status and supplier comment attribute values in the supplier RM-systems. Then the tester created a ReqIF export of the imported modules and the export was checked for validity. In the customer RM-systems the tester performed changes of content and links in one module of the specification in a parallel working scenario. The supplier exports were imported in the customer RM-systems and the modules were checked according to the benchmark criteria e.g., if changes are indicated on import and the capability of editing and exchanging supplier attributes with the supplier RM-system (see Chapter 2.5). After this check Pong is completed.

For Ping 2 the tester added customer status and customer comment attribute values in the customer RM-systems and created a ReqIF export containing all three modules. The export was checked for validity. The ReqIF exported was then imported again to the supplier RM-systems. The imported modules were checked according to the benchmark criteria e.g., successful change of requirement text, change of attribute values, change of attribute type definition and structural changes (see Chapter 2.5). After this check Ping 2 is completed.

2.7 Documentation

This short report is made publicly available. A long report with more detailed information is provided to the members of prostep ivip and VDA.

3 Results

In the following an overview of the results is given. Then, the detailed results for every RM-system are described.

3.1 Overview

The overview is separated in results for setup, the first ping, for the pong and for the second ping. The evaluation of the tested criteria is classified in four different categories: Correct, partially correct, failed and not tested. The results overview does not distinguish between valid and invalid exports.

3.1.1 Setup

Before the first ping, the reference ReqIF packages were imported to all RM-systems. In order to establish a comparable starting point in the different RM-systems, manual corrections were made as needed after the initial import. Criteria not supported by the RM-systems were not tested.

The initial import of the reference ReqIF packages shows 24 “partially correct” test cases (see Figure 13). Three RM-systems have issues importing the reference ReqIF packages and did not perform a complete import. Two systems did not import the Visio file and one system could only import the reference ReqIF packages after changing the encoding of the file from UTF-8-BOM to UTF-8. However, the same system did not import the indentation levels and simple table correctly. Furthermore the PNG file was imported in the order “text1, text2, png” instead of “text1, png, text2” and no Visio file and links were imported.

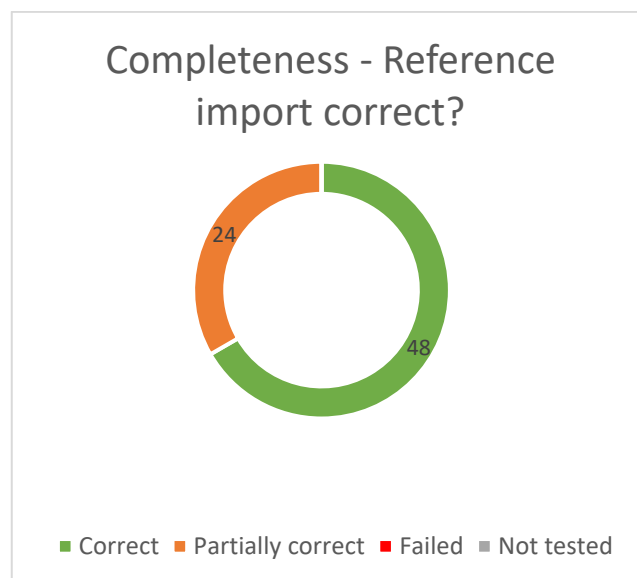


Figure 13: Results for completeness of reference import

3.1.2 Overview Ping 1

The focus of Ping 1 is on the completeness of the exchanged specification. All requirements must be exchanged in the formatted style with the corresponding attributes and values. Embedded document and image files must be exchanged. Links between requirements in different modules must be exchanged when the linked modules are in the exchanged data packages. As described in Chapter 2.3, there are 72 system combinations to be tested in this phase of the benchmark.

Validity/Completeness

Figure 14 shows the results for validity of the first export from the customer system to ReqIF. In 56 test cases the exported files were valid ReqIF files according to the ReqIF XML schema. 16 test cases are rated as "failed" because the Asaro Q-Checker reported issues for two systems concerning the first ReqIF-exports. The Q-Checker reported unreferenced files in the archive. For one system the Boolean-Attribute could not be exported.

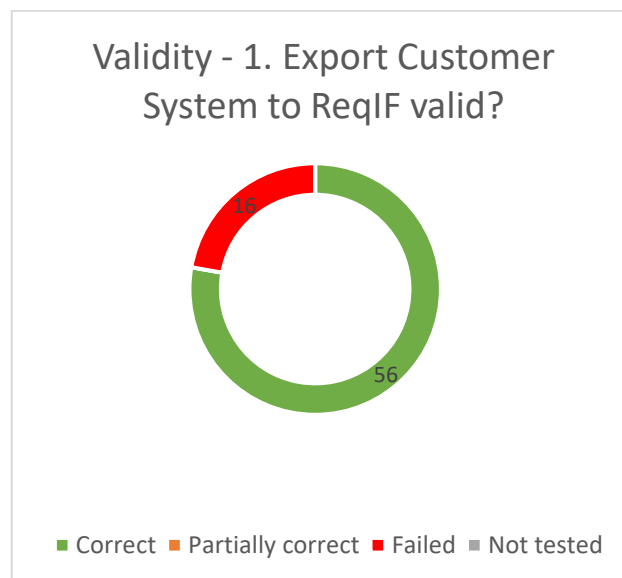


Figure 14: Results for validity of first export to ReqIF

Figure 15 shows that all systems were able to import the specifications completely with all requirements and relevant attributes that are contained in the first customer exports.

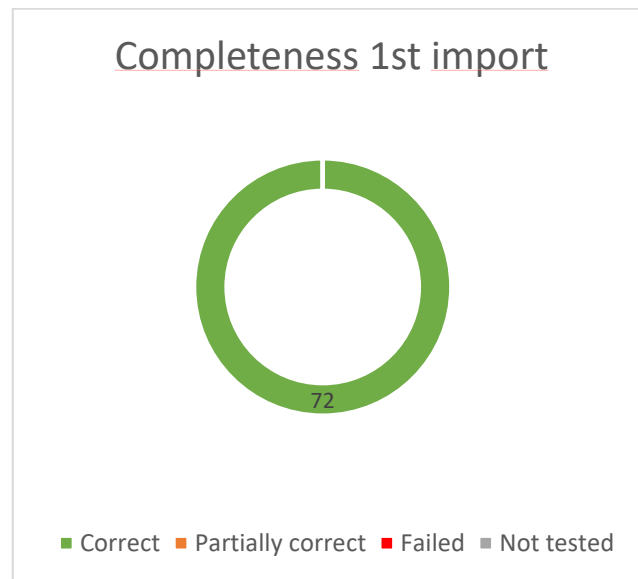


Figure 15: Results for completeness of first import to supplier RM-system

Exchange of formatted text

Figure 16 shows the results for the exchange of the font styles bold, italic, underlined, strike out, subscript and superscript. In five test cases the font styles were not imported completely and were therefore evaluated as partially correct. Three test cases only exchanged the styles underlined and strike out successfully and in two test cases the styles underlined and strike out are missing.

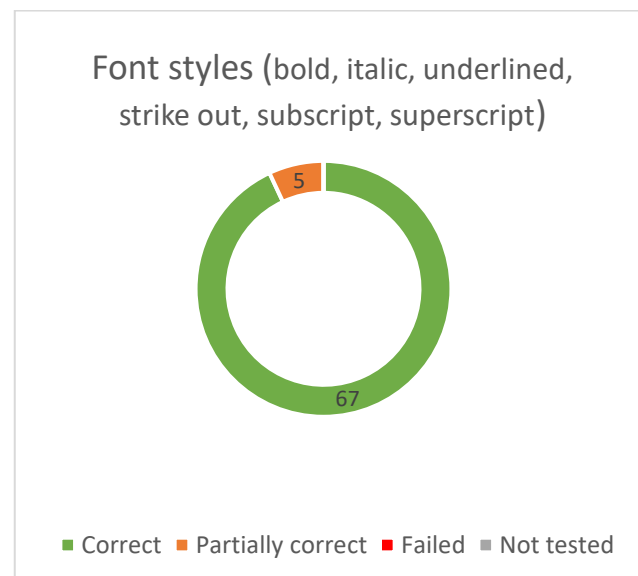


Figure 16: Results for exchange of font styles

Figure 17 shows the results for the exchange of simple indentations. In six test cases no simple indentations could be imported. In four test cases only one level instead of several levels of indentations could be imported. This behavior is evaluated as partially correct.

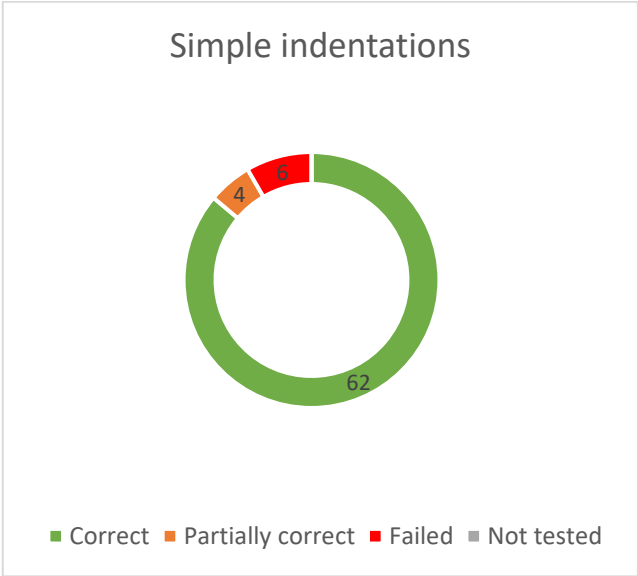


Figure 17: Results for exchange of simple indentations

Figure 18 shows the results for the exchange of simple numbered lists. In 29 test cases simple numbered lists could be exchanged successfully. 24 test cases are marked as not tested because these test cases are related to one system which does not support numbered lists in general as stated by the vendor. Therefore, numbered lists could not be exported from that system and consequently not be imported in the supplier system. In 19 test cases the import of simple numbered lists was evaluated as failed because the correctly performed export could not be imported by the system which does not support numbered lists.

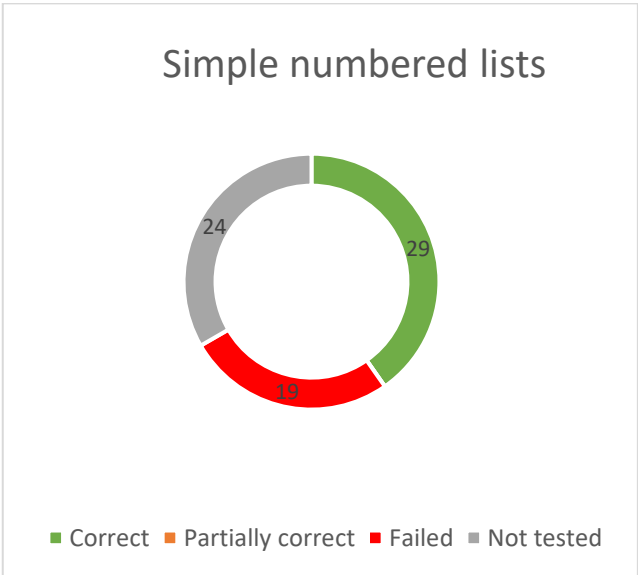


Figure 18: Results for exchange of simple numbered lists

Figure 19 shows the results for the exchange of simple bullet points. In all test cases simple bullet points could be exchanged.

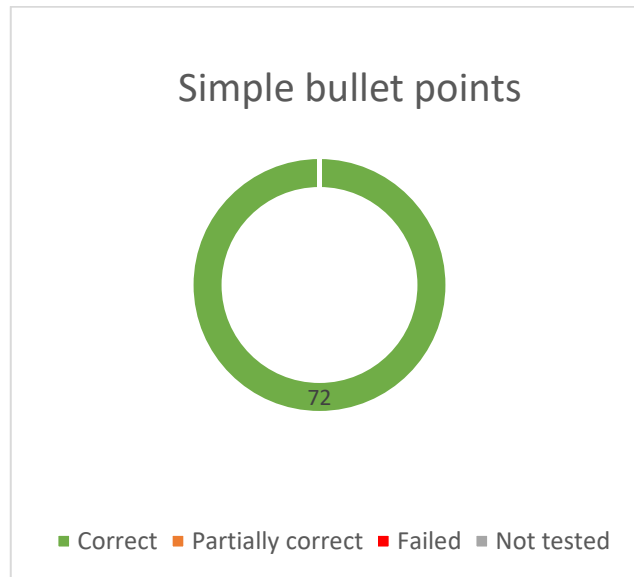


Figure 19: Results for exchange of simple bullet points

Figure 20 shows the results for the exchange of simple tables. In 40 test cases simple tables could be exchanged successfully. 24 test cases are marked as not tested because these test cases are related to one system which does not support simple tables in general as stated by the vendor. Therefore, simple tables could not be exported from that system and consequently not be imported in the supplier system. In eight test cases the import of simple tables was evaluated as failed.

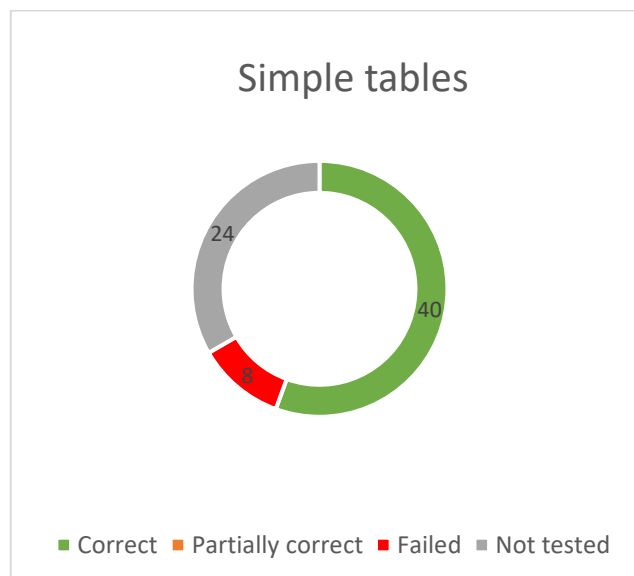


Figure 20: Results for exchange of simple tables

Exchange of embedded document files

Figure 21 and Figure 22 show the results for the exchange of document files (Word, Excel, PowerPoint, PDF and Visio). In all test cases Excel files were exchanged successfully. Successful in that context means, that the

embedded document file must be accessible (no preview necessary) from the attribute where they were originally embedded in and the content must be in the same order <text1, embedded object, text2> as before the exchange. For the exchange of Word, PowerPoint and PDF files one test case was evaluated as partially correct. This case applies to the same tool combination. The embedded document files were accessible, but the order of text and embedded object was rearranged as <text1, text2, embedded object>.

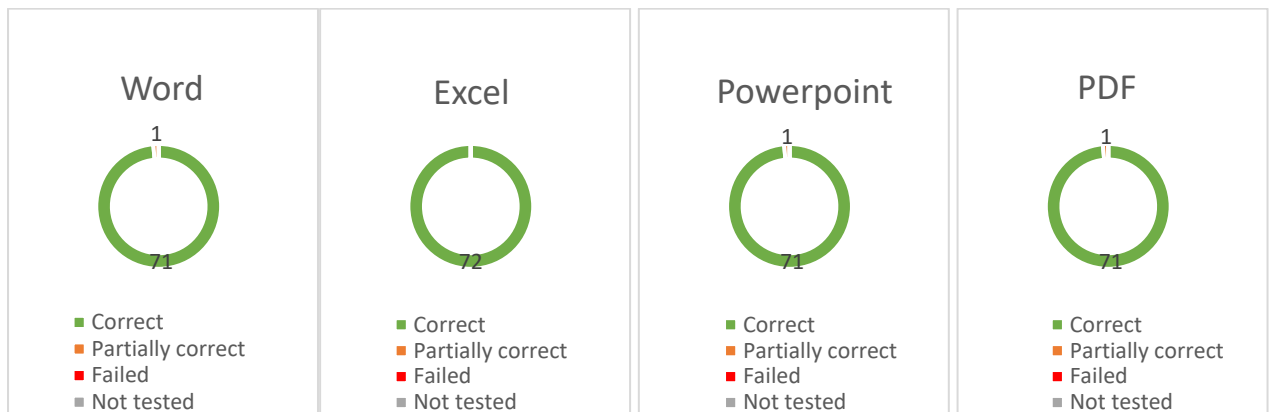


Figure 21: Results for exchange of embedded document files (excl. Visio)

The exchange of Visio files (Figure 22) was successful in 50 test cases. In 22 test cases the exchange of the Visio file failed because either no file was imported or the file could not be opened.

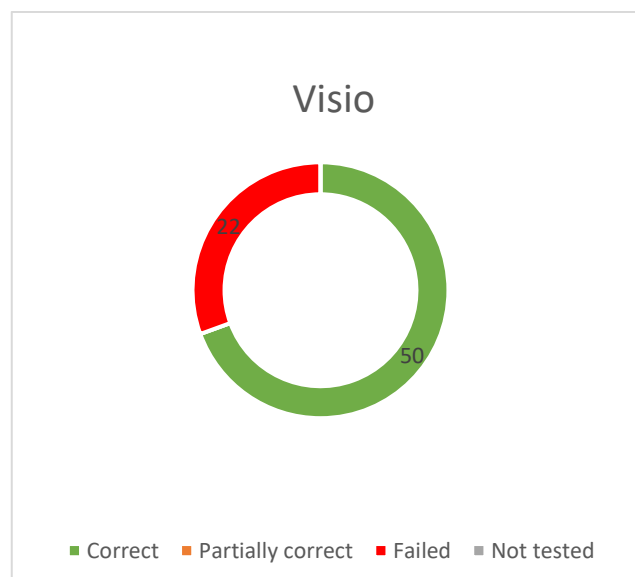


Figure 22: Results for exchange of Visio files

Exchange of embedded image files

Figure 23 and Figure 24 show the results for the exchange of image files (emf, bmp, gif, jpg and png). In four test cases the exchange of emf files failed because the file could not be accessed. For the exchange of bmp

and gif files one test case regarding the same tool combination was evaluated as partially correct. The embedded images were rearranged in the order <text1, text2, bmp/gif>. In three test cases the bmp and gif files could not be accessed which was evaluated as failed. The exchange of jpg files failed in three test cases where the file could not be accessed.

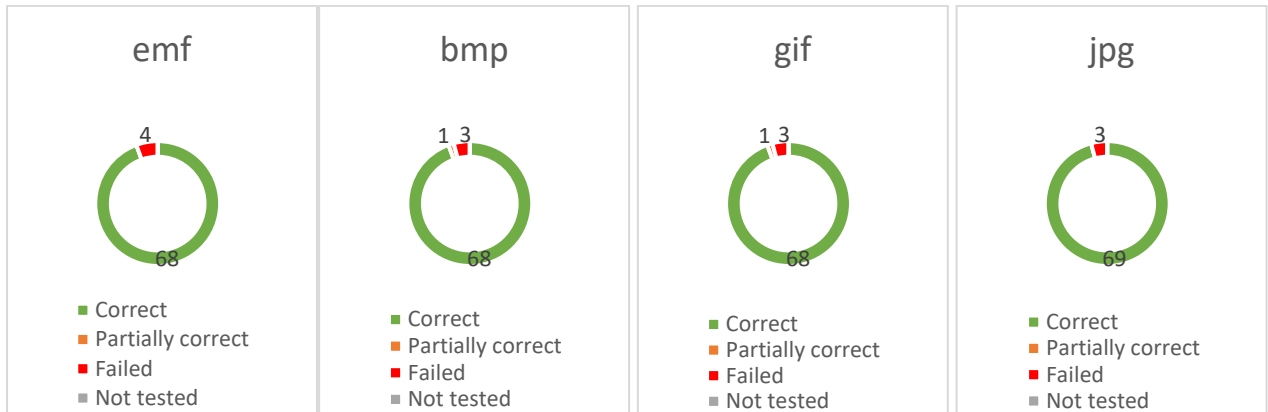


Figure 23: Results for exchange of embedded image files (excl. png)

The exchange of png files (Figure 24) was successful in 64 test cases. In five test cases the exchange of the png file was evaluated as partially correct because in one tool combination was only a preview of the png file available and the embedded object could not be accessed. Furthermore, the order of the embedded object was rearranged to <text1, text2, png> which occurred in the remaining four partially correct test cases. Three test cases failed because the png file could not be accessed.

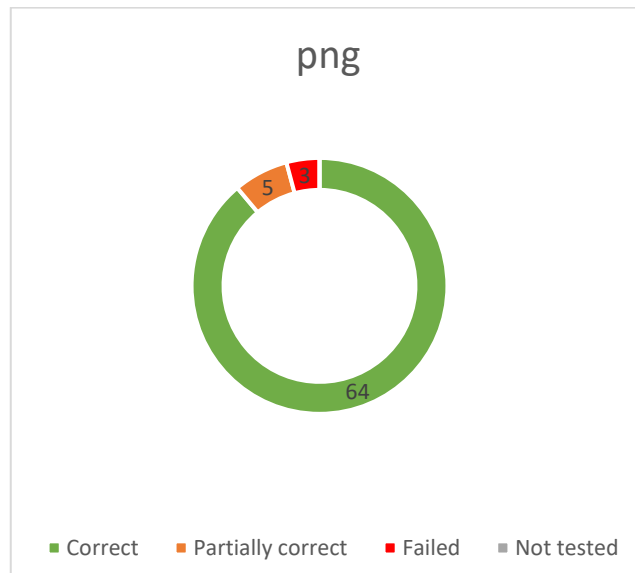


Figure 24: Results for exchange of png files

Exchange of linked requirements

Figure 25 shows the results for the exchange of linked requirements. The links between requirements within the same module (Links A-A) were exchanged successfully in 65 test cases. In seven test cases the links between requirements within the same module were not exchanged.

Links between requirements in different modules (Links A-B) were exchanged successfully in 59 test cases and were missing in nine test cases. In four test cases links between module A and B could not be imported due to an issue relating to one RM-system during the first setup of the benchmark. As stated by the vendor this RM-system is in principle able to import the links. For this reason, four test cases were evaluated as partially correct.

The other way round (Links B-A) links were exchanged successfully in 64 test cases and were missing in eight test cases. Links between requirements of an exported module and a module that was not exported (Links B-C, Links C-B) have not been exchanged which is the expected behavior and therefore evaluated as correct.

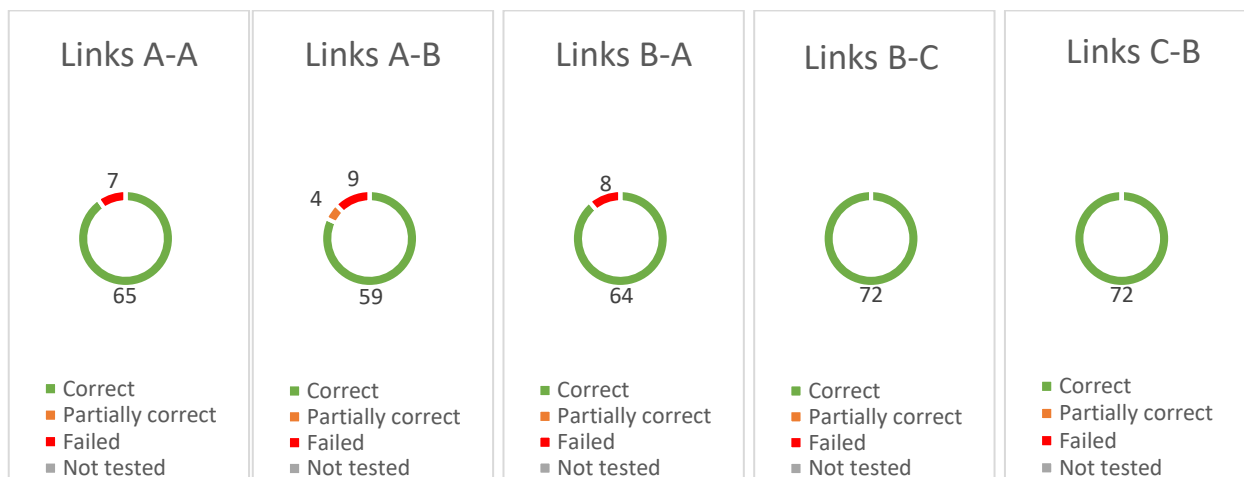


Figure 25: Results for exchange of requirement links

3.1.3 Overview Pong

The focus of the Pong is on the supplier feedback in the supplier attributes to be imported to the existing specification in the customer RM-system. The additional focus is on ensuring that the changes made during parallel working are not overwritten by the import. As described in Chapter 2.3, there are 56 system combinations to be tested in this phase of the benchmark due to the absence of one RM-system. For one test case Pong and Ping 2 could not be completed because of an unexpected change of an attribute type definition. Therefore, this test case was evaluated as not tested in the following criteria.

Validity

Figure 26 shows the results for validity of the export from the supplier RM-system to ReqIF. In 41 test cases the exported files were valid ReqIF files according to the ReqIF XML schema. 15 test cases are rated as "failed" because the Asaro Q-Checker reported issues for six RM-systems concerning the ReqIF exports. These issues were related to validation errors, unreferenced files or missing files. The tests could be continued with the exports evaluated as invalid.

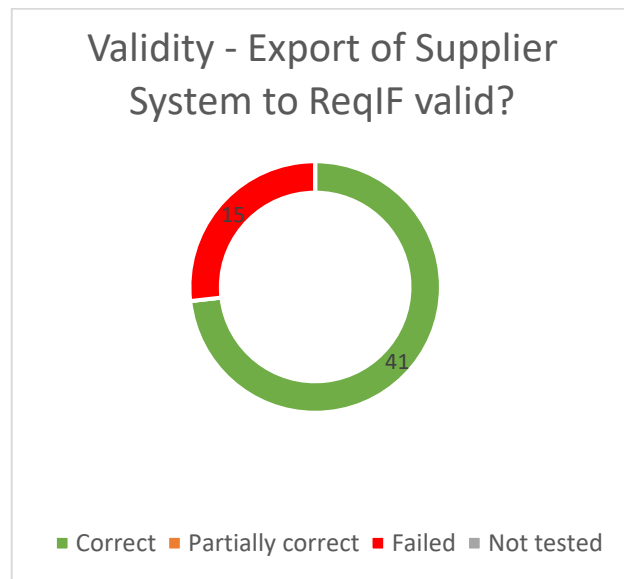


Figure 26: Results for validity of supplier export to ReqIF

Changes indicated on import to customer system

Figure 27 shows the results for changes indicated on import to the customer RM-system. In 55 test cases it was possible to identify changes when a requirement module is updated. The customer importing RM-system was able to visualize changes before, during or after the import.

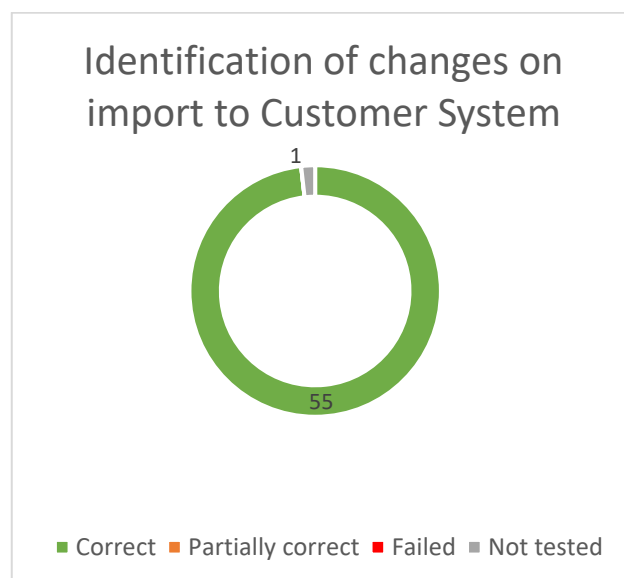


Figure 27: Results for changes indicated on import to customer RM-system

Capability to edit supplier attributes in supplier RM-system

Figure 28 shows the results for changes in supplier status. In 55 test cases the supplier status could be transferred successfully. As described above one test case could not be completed.

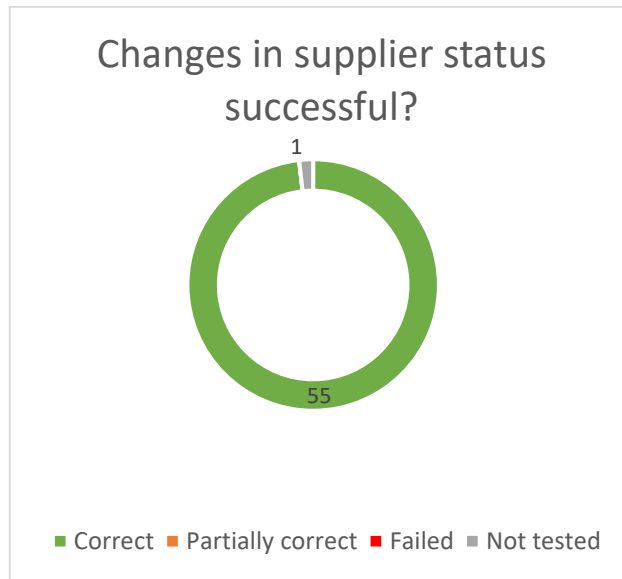


Figure 28: Results for changes in supplier status

Figure 29 shows the results for changes in supplier comment. In 41 test cases the supplier comment including double line break could be transferred successfully. 12 test cases were evaluated as partially correct because the comment text was transferred successfully but the double line caused problems which are expressed by either only one line break or no line break visible in the customer RM-system. In some partially correct test cases the line break already could not be created in the appropriate attribute field of the supplier RM-system. In two test cases concerning the same customer RM-system the supplier comments were not adopted.

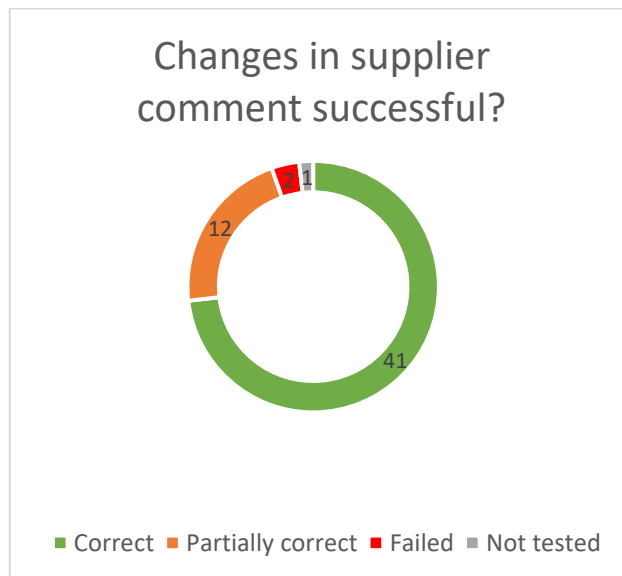


Figure 29: Results for changes in supplier comment

Figure 30 shows the results for the exchanged image (JPG) attached in the supplier comment. In 28 test cases the attached image could be transferred successfully. In one partially correct test case there was only a small,

truncated preview of the image visible and the image could not be accessed. In seven test cases the customer system did not show an image in the supplier comment field. 20 test cases are marked as not tested because it was not possible to attach the image to the supplier comment.

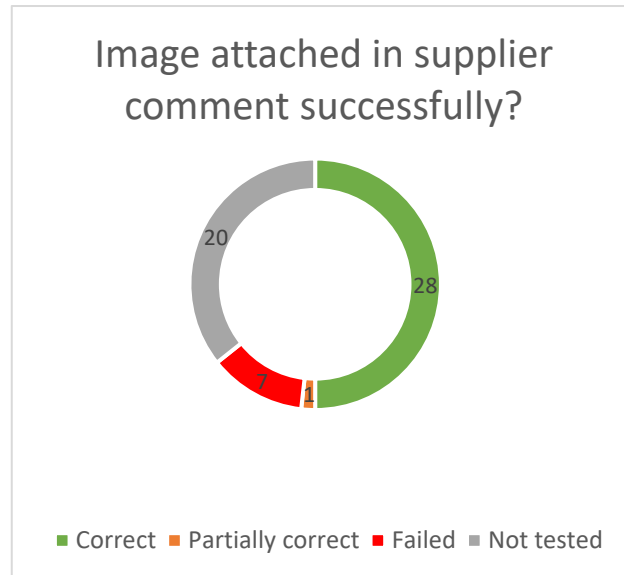


Figure 30: Results for image attached in supplier comment

Customer attributes unchanged

Figure 31 shows the results for the capability of the different RM-systems acting as customer to import selected attributes. In 49 test cases the attributes to be imported could be selected. For seven test cases concerning the same RM-system it was not possible to select the attributes to be imported.

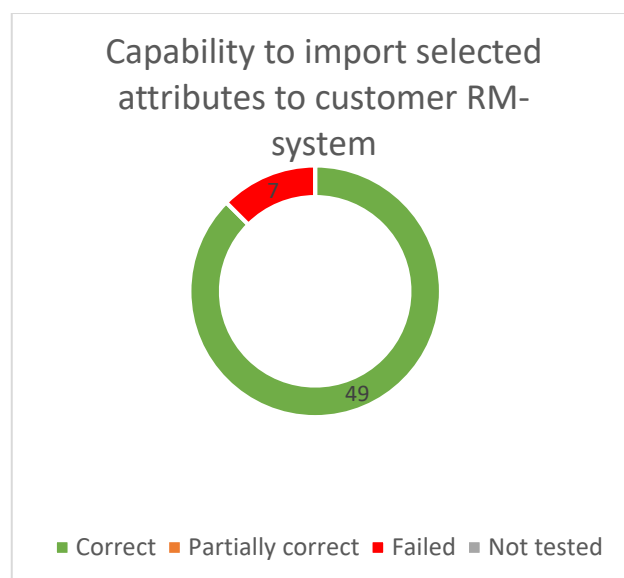


Figure 31: Results for capability to import selected attributes to customer RM-system

Capability to export selected attributes from supplier RM-system

Figure 32 shows the results for the capability of the different RM-systems acting as supplier to export selected attributes. In all test cases the attributes to be exported could be selected.

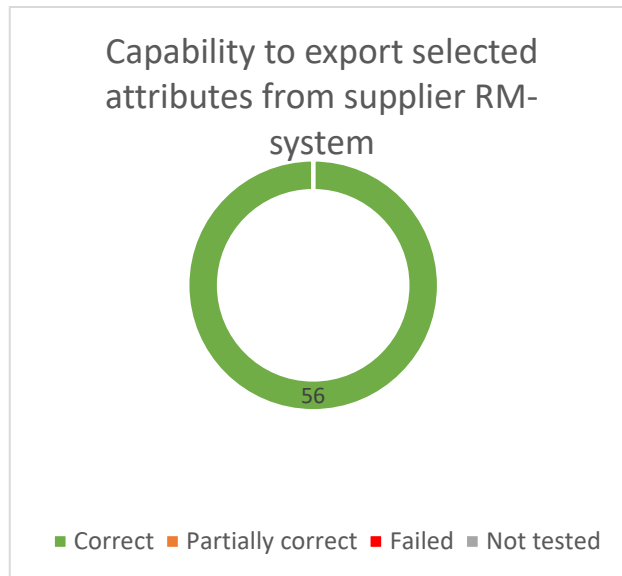


Figure 32: Results for capability to export selected attributes from supplier RM-system

No further changes after re-import to customer system

Figure 33 shows the results for no further changes after re-import to customer RM-system. In 32 test cases no further changes to the already imported and modified specification occurred. In 23 test cases, in addition to the supplier attributes to be imported, other changes to the already existing and modified specification were detected such as recovery of deleted links, changes in document structure and changes in newly created/deleted requirements.

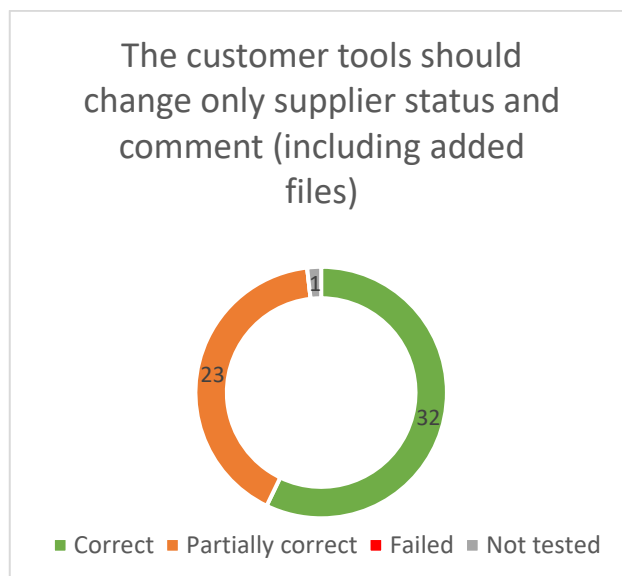


Figure 33: Results for no further changes after re-import to customer RM-system

3.1.4 Overview Ping 2

The focus of Ping 2 is on the one hand on the customer feedback in the customer attributes to be imported and on the other hand on the changes performed in the customer RM-systems in a parallel working scenario to be imported in the supplier RM-systems. As described in Chapter 2.3, there are 56 system combinations to be tested in this phase of the benchmark due to the absence of one RM-system. In this benchmark phase one further test case could not be completed because of an internal tool error (reported to vendor).

Validity

Figure 34 shows the results for validity of the export from the customer RM-system to ReqIF. In 41 test cases the exported files were valid ReqIF files according to the ReqIF XML schema. 14 test cases are rated as "failed" because the Asaro Q-Checker reported issues for four RM-systems concerning the ReqIF exports. The Q-Checker reported mostly validation errors and missing files. The tests could be continued with the exports evaluated as invalid. Identical errors, which already occurred during the check of Ping 1 are evaluated as subsequent errors and therefore as correct.

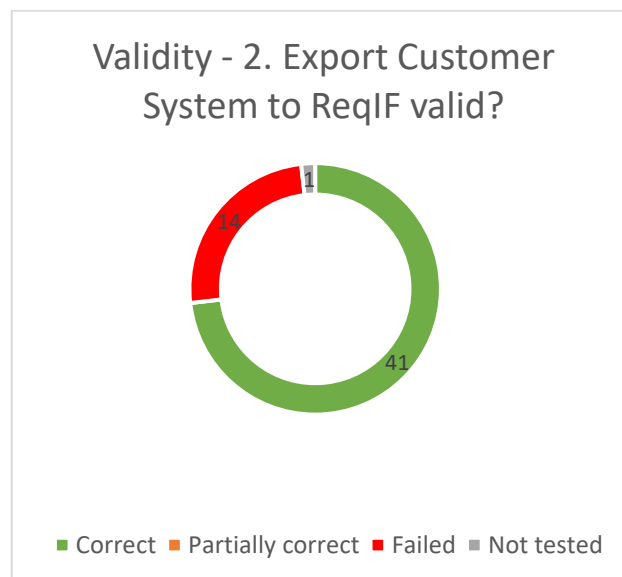


Figure 34: Results for validity of second export to ReqIF

Change of requirement text

Figure 35 shows the results for the change of requirement text. In 52 test cases the changed requirement text in the customer RM-system was adopted on import to the supplier RM-system. In two test cases the changed requirement text was not imported.

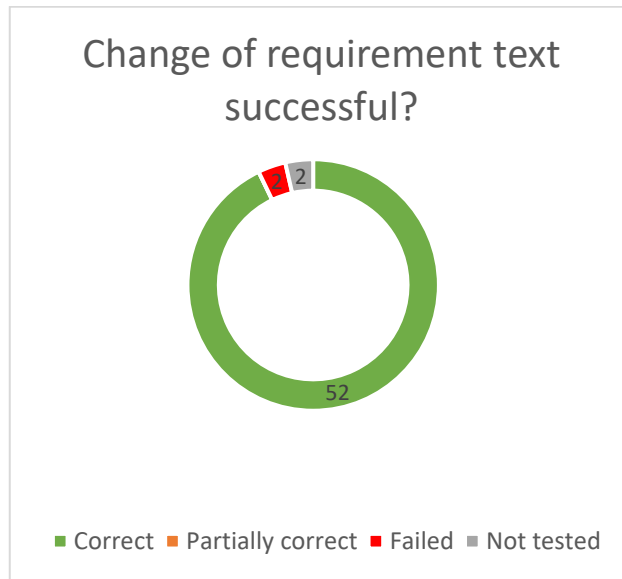


Figure 35: Results for change of requirement text

Change of attribute values

Figure 36 shows the results for the change of attribute values. In 47 test cases the changed attribute values for different type definitions (Boolean, Date, Enumeration, Integer, Real and String) were adopted successfully on import to the supplier RM-system. In six partially correct test cases the changed attribute values were not imported completely because the concerning RM-system could not export the Boolean type. In one test case the attribute values were not adopted.

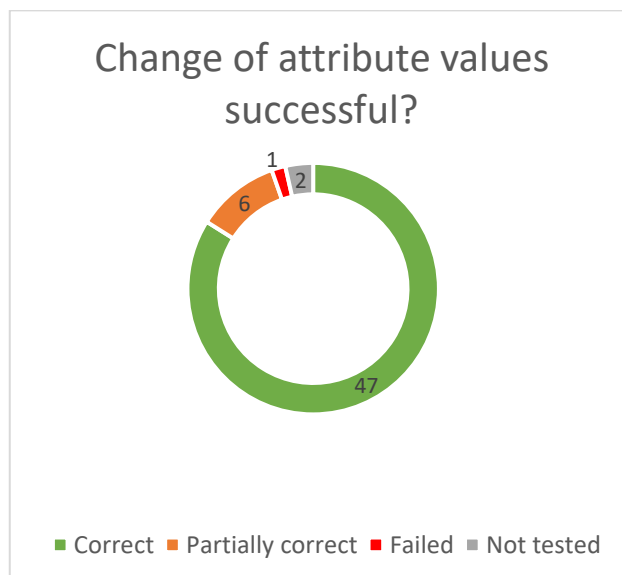


Figure 36: Results for change of attribute values

Figure 37 shows the results for the change of attribute values for the type definitions described above to empty values. In 38 test cases the empty values were adopted successfully on import to the supplier RM-system. 14 test cases were evaluated as partially correct because in two RM-systems either date or Date,

Real, Integer and Boolean values cannot be set to empty. One test case failed due to an internal error in the RM-system which was responsible for only partially completing the merge process with the already existing specification. This situation was reported to the vendor. The other test case failed because attribute values were not adopted.

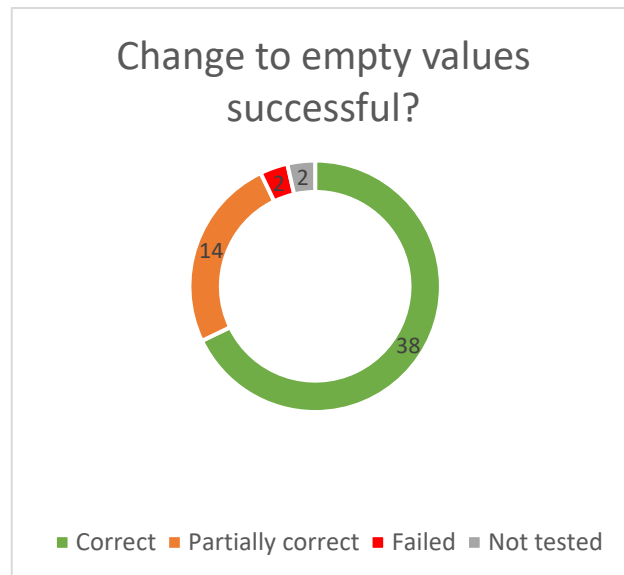


Figure 37: Results for change of attribute values to empty values

Change of attribute type definition

Figure 38 shows the results for the change of the enumeration definition of an attribute type where an additional option "Variant D" was added. In 36 test cases the changed attribute type definition was adopted successfully on import to the supplier RM-system. In 11 test cases the changed attribute type definition was not imported as intended and therefore evaluated as failed. In addition to the already mentioned two not tested test cases, seven test cases were evaluated as not tested, since it was not possible to define the additional option for Variant D in one RM-system.

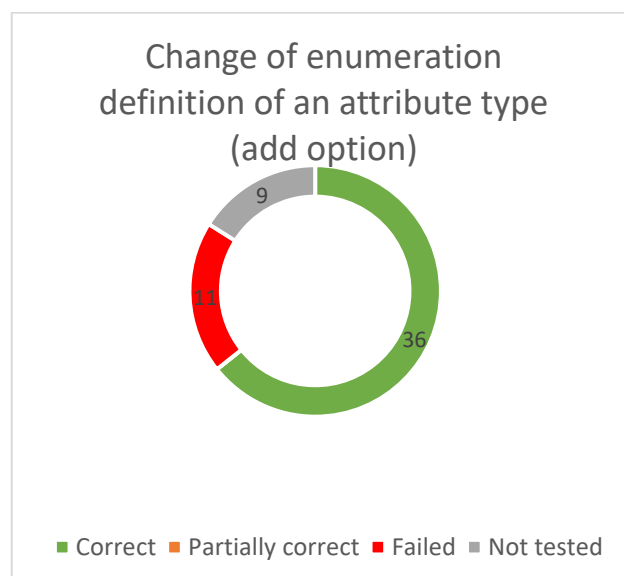


Figure 38: Results for change of enumeration definition of attribute type

Exchange of structural changes

Figure 39 shows the results for the exchange of a requirement added within the parallel working scenario to be imported to the supplier RM-system. In 52 test cases the added requirement could be identified and imported to the supplier RM-system. In the partially correct test case, the new requirement was not imported as intended. In one test case the added requirement could not be identified in the supplier RM-system due to the already described internal error in one RM-system which was responsible for only partially completing the merge process with the already existing specification.

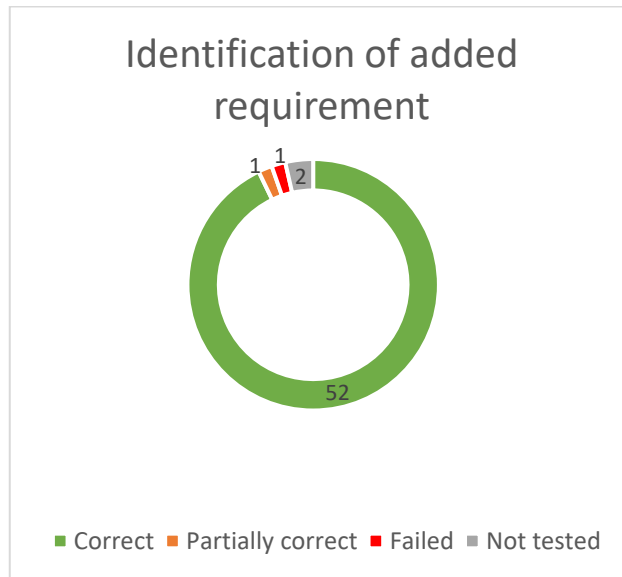


Figure 39: Results for exchange of added requirement

Figure 40 shows the results for the exchange of a missing requirement deleted within the parallel working scenario to be imported to the supplier RM-system. In 47 test cases the missing requirement could be identified in the supplier RM-system. In seven test cases the missing requirement could not be identified in the supplier RM-system and was still available.

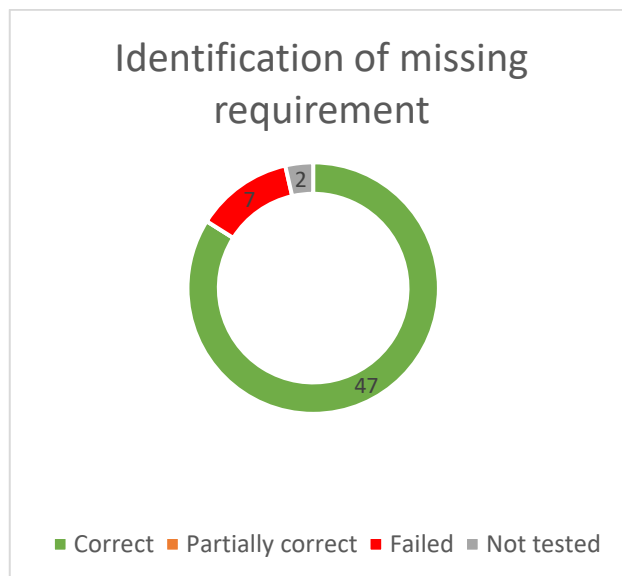


Figure 40: Results for exchange of missing requirement

Figure 41 shows the results for the exchange of a changed document structure reordered within the parallel working scenario to be imported to the supplier RM-system. In 52 test cases the changed document structure was adopted on import to the supplier RM-system. In two test cases the changed document structure was not adopted on import to the supplier RM-system of which one failed test case is assigned to the described internal error during the merge process.

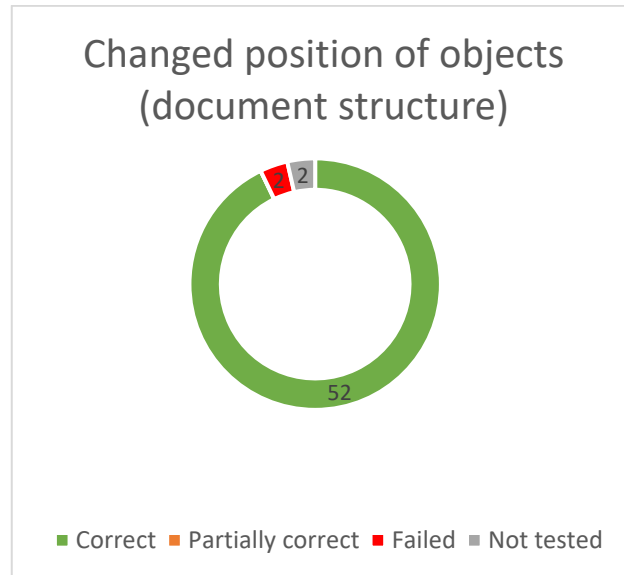


Figure 41: Results for exchange of changed document structure

Update of links

Figure 42 shows the results for the exchange of updated links between the modules A, B and C. As part of the parallel working scenario new links from module A-A and A-B were created. In 41 test cases the new links from A-A and A-B were imported successfully by the supplier RM-system. In 13 test cases the new links from A-A and A-B were not imported by the supplier RM-system.

Furthermore, module C which contains links to module B, was imported initially to the supplier RM-system. In 43 test cases links from module B to module C were imported successfully by the supplier RM-system and in 11 test cases the links were not imported.

Links from module C to B were successfully imported by the supplier RM-system in 45 test cases and failed in nine test cases.

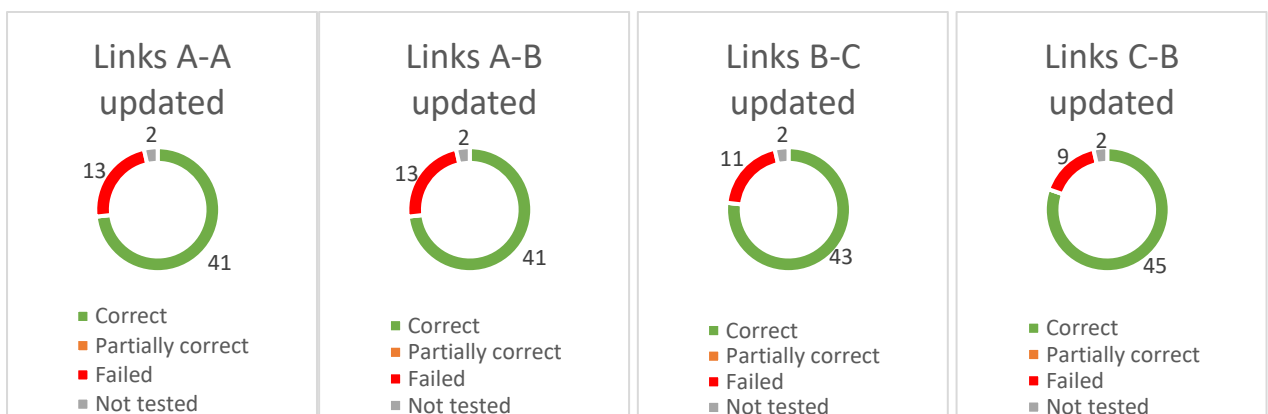


Figure 42: Results for exchange of updated links

Changes indicated on import to supplier RM-system

Figure 43 shows the results for changes indicated on import to the supplier RM-system. In 54 test cases it was possible to identify changes when a requirement module is updated. The supplier importing RM-system was able to visualize changes before, during or after the import.

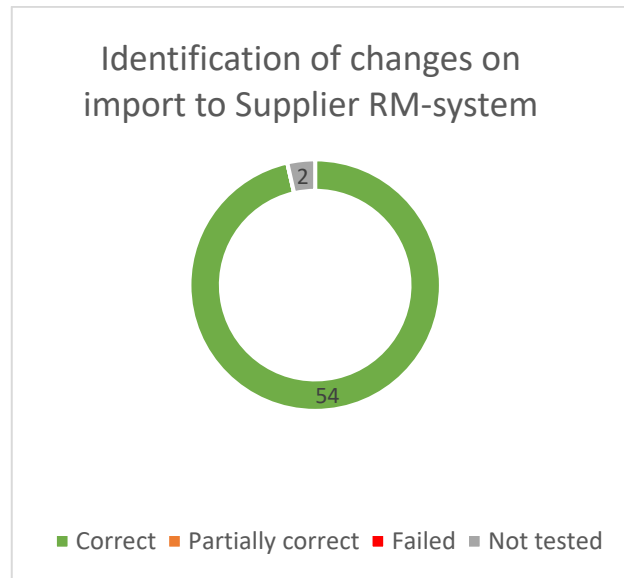


Figure 43: Results for changes indicated on import to supplier RM-system

No further changes after re-import to supplier RM-system

Figure 44 shows the results for no further changes after re-import to the supplier RM-system. The supplier RM-system should maintain the supplier attributes. In 54 test cases the supplier attributes were successfully maintained.

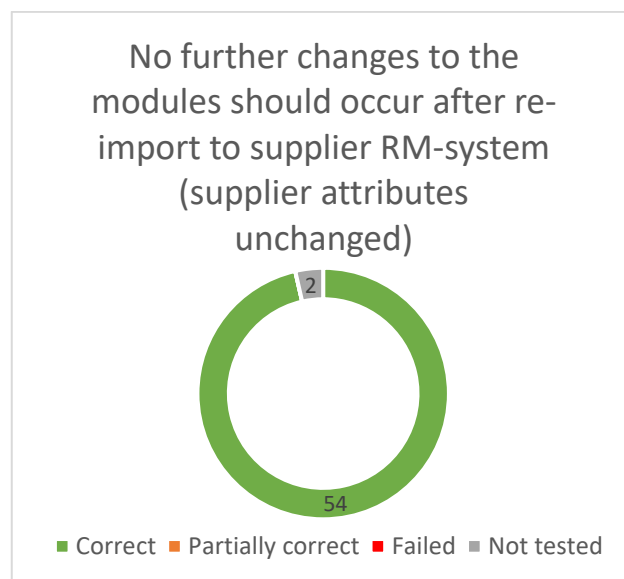


Figure 44: Results for no further changes after re-import to supplier RM-system

Capability to edit and exchange customer attributes with customer RM-system

Figure 45 shows the results for changes in customer status. In 53 test cases the customer status could be transferred successfully. In one test case the customer status attribute was not imported.

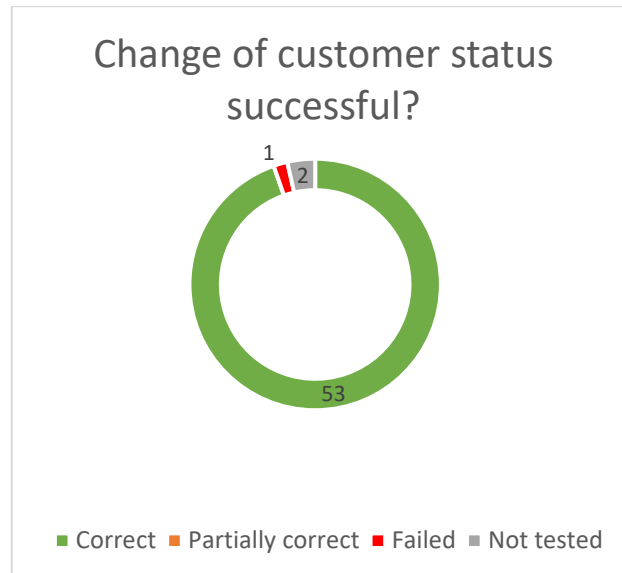


Figure 45: Results for changes in customer status

Figure 46 shows the results for changes in customer comment. In 53 test cases the customer comment could be transferred successfully. In one test case the customer comment attribute was not imported.

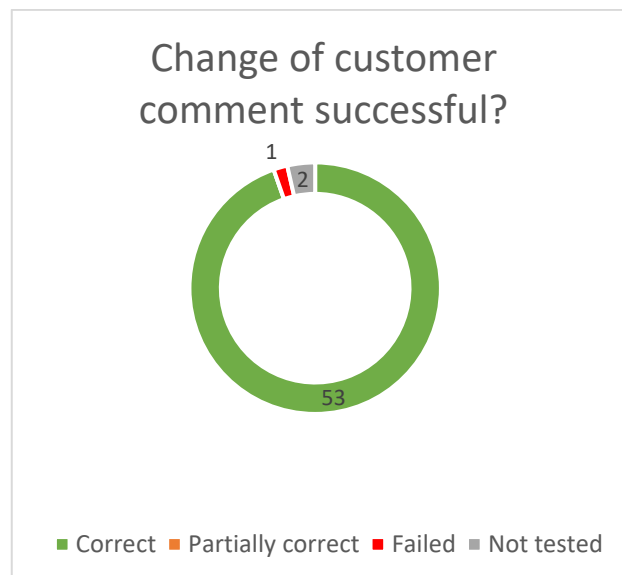


Figure 46: Results for changes in customer comment

Formatted text (recheck open points Ping 1)

The recheck concerning formatted text is performed for all test cases with one specific RM-system including Add-ons acting as customer. As a starting point in these RM-systems simple numbered lists and simple tables have been inserted in RTF format and consequently been exported.

Figure 47 shows the results for the exchange of simple numbered lists. 35 test cases were not tested because of concerning RM-systems for which a recheck was not necessary and one was not tested as already described. In 20 test cases simple numbered lists were not exchanged successfully because the numbered list was converted into bullet points during export. In consequence the importing RM-systems displayed only bullet points.

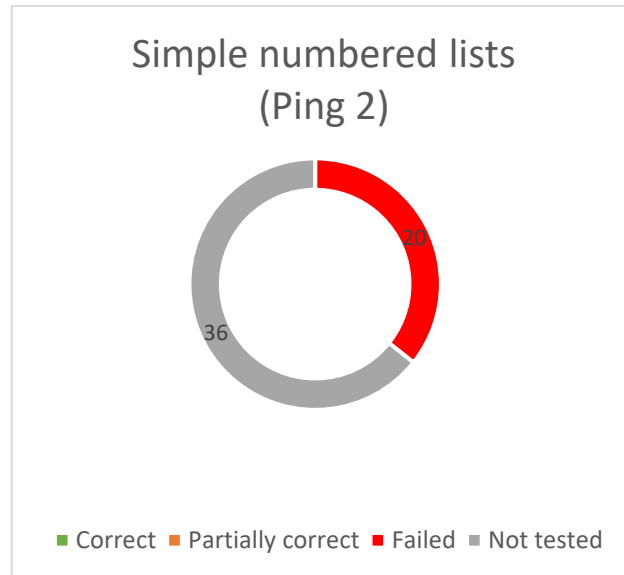


Figure 47: Results for exchange of simple numbered lists (Ping 2)

Figure 48 shows the results for the exchange of simple tables. Also, for this criterion 35 test cases were not tested because a recheck was not necessary and one was not tested as already described. In five test cases simple tables have been transferred successfully. 15 test cases were marked as partially correct because not all rows and columns of the simple table were displayed and table boundaries were missing.

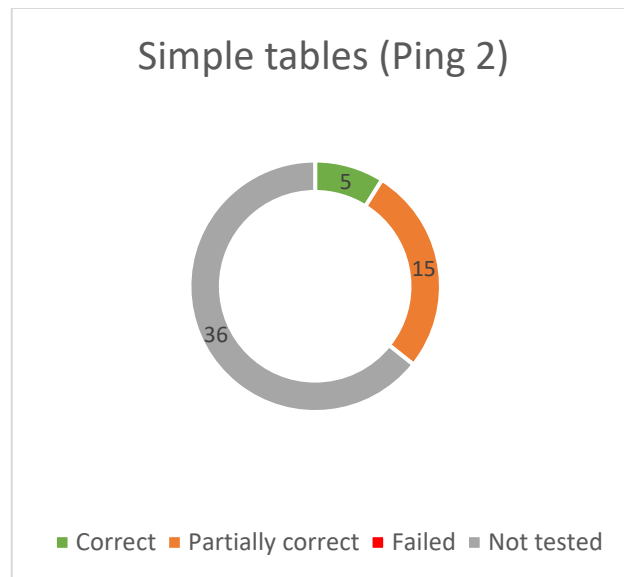


Figure 48: Results for exchange of simple tables (Ping 2)

4 Summary and Outlook

The benchmark shows the possibilities and limits of ReqIF for the defined requirement exchange process. Regarding the transfer of requirement data, further improvements can be seen since the last benchmark.

This benchmark also helped to identify issues in the requirement exchange process and raised awareness with the implementers and users of Requirements Management Systems. The issues raised during this benchmark may be solved in further releases of the used software and the implementers will continue working together in the ReqIF Implementor Forum to improve the exchange across different RM-systems. The users organized in the ReqIF Workflow Forum can use the information created in this benchmark to define use cases and requirements more explicitly.

For the future there is the possibility to conduct further benchmarks according to the need of the users to improve the ReqIF standard and interfaces. Future benchmarks will be based on user-defined processes and requirements. Software vendors may also propose topics that might be tested in a future benchmark.

5 Acknowledgements

prostep ivip and VDA like to thank all participating companies, Asaro Systems Limited, IBM, Intland Software GmbH, REQUISIS GmbH, :em engineering methods AG and Siemens Industry Software GmbH for providing software and licenses as well as technical support and advice during this benchmark.