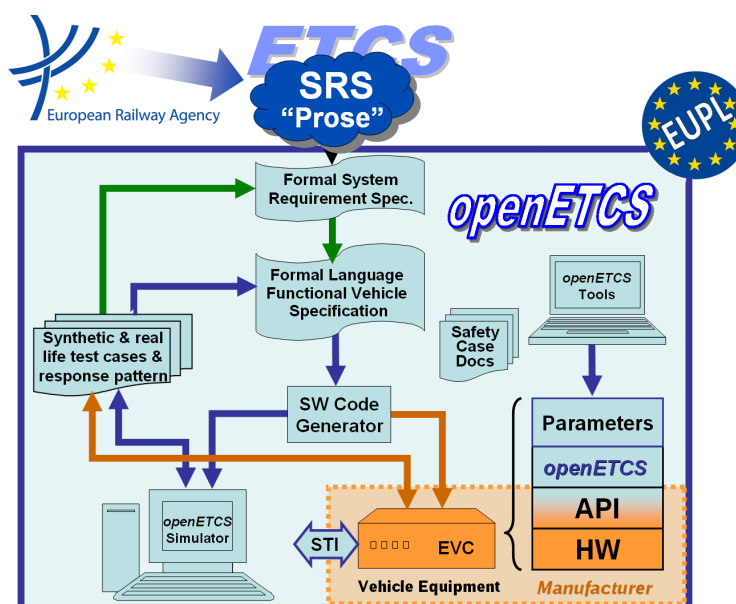openETCS

Work-Package 7: "Secondary tools - Management"

Evaluation of supporting tools and methods against the WP2 requirements and task 1

List of criteria on supporting tools and methods and results on the benchmark

Marielle Petit-Doche, all participants of the benchmark and all participants of VnV and Safety process

October 2013

This page is intentionally left blank

Work-Package 7: "Secondary tools - Management"   **OETCS/WP7/O7.2.1 – 00/05**
October 2013

Evaluation of supporting tools and methods against the WP2 requirements and task 1

List of criteria on supporting tools and methods and results on the benchmark

Marielle Petit-Doche

Systerel

all participants of the benchmark

WP7 partners

all participants of VnV and Safety process

WP4 partners

Evaluation

Prepared for   openETCS@ITEA2 Project

Abstract: This document gives elements to evaluate the tools and methods to complete the primary toolchain and to support verification and validation activities, safety activities, moodel transformation and data management for the whole project. Evaluation on the means and tools of benchmark is also described.

This document focusses on means and tools to support management of data, functions, requirements and all the artifact during the OpenETCS process.

Table of Contents

Figures and Tables

Figures

Tables

Document information	
Work Package	WP7
Deliverable ID or doc. ref.	O7.2.1
Document title	Evaluation of supporting tools and methods against the WP2 requirements and task 1 - Management
Document version	00.05
Document authors (org.)	Marielle Petit-Doche (Systerel)

Review information	
Last version reviewed	00.04
Main reviewers	

Approbation			
	Name	Role	Date
Written by	Marielle Petit-Doche	WP7-T7.1 Sub-Task Leader	
Approved by	Michael Jastram	WP7 leader	

Document evolution			
Version	Date	Author(s)	Justification
00.01	19/07/2013	M. Petit-Doche	Document creation
00.02	09/09/2013	M. Petit-Doche	Major evolutions in all document
00.03	19/09/2013	M. Petit-Doche	Issues: 167, 168, 170
00.04	23/09/2013	M. Petit-Doche	Issues: 164, 169, 174, 175, 177, 178
00.05	01/10/2013	M. Petit-Doche	Split of document O7.2.1. Management part

1   Introduction

The aim of this document is to report the results of the evaluation of means and tools for the secondary means and tools, i.e. the means and tools which complete the primary tool chain dedicated to formal model and software design.

This evaluation task is part of work package WP7, task 2 "Secondary tools analyses and recommendations". According to the results of WP2, especially the OpenETCS process and the requirements on language and tools [2], and the results of T7.1 on the primary toolchain [10], the aim of this task is to determine the best candidates to complete and support the primary toolchain for the following activities:

- verification and validation (WP4)

- safety activities support (WP4)

- data, function and requirement management (SSRS, WP3 and WP4)

- model transformation and code generation (WP3 and WP4)

This document is dedicated to tools and means to support management of data, functions requirements and other artifacts along the openETCS process.

1.1   Organisation of the document

The chapter 2 provides a template to describe the means and tools and a list of criteria according WP2 requirements on language, models and tools, and T7.1 primary tool chain decision. The objectives of this description and criteria are to allow to determine the best means of description and associated tool for a given activities.

The chapter 3 resumes the results of the evaluation at the end of the benchmark activities.

In Appendix, a chapter is dedicated to each models produced during the benchmark activities :

- Scade Suite

- Rodin and Pluggins

- Matelo

- Goal Structuring Notation (GSN)

- Eclipse ProR

- Eclipse EMF Store

- Eclipse EMF Client Platform

2   Template

2.1   Instructions

Author Author of the approaches description %%Name - Company%%

Assessor 1 First assessor of the approaches %%Name - Company%%

Assessor 2 Second assessor of the approaches %%Name - Company%%

In the sequel, main text is under the responsibilities of the author.

Author: *Author can add comments using this format at any place.*

Assessor 1: *First assessor can add comments using this format at any place.*

Assessor 2: *Second assessor can add comments using this format at any place.*

When a note is required, please follow this list :

0 not recommended, not adapted, rejected

1 weakly recommended, adapted after major improvements, weakly rejected

2 recommended, adapted (with light improvements if necessary) weakly accepted

3 highly recommended, well adapted,strongly accepted

* difficult to evaluate with a note (please add a comment under the table)

All the notes can be commented under each table.

2.2   Presentation

This section gives a quick presentation of the approach and the tool.

Name   %%Name of the approach and the tool%%

Web site   %%if available, how to find information%%

Licence   %%Kind of licence%%

Abstract

Short abstract on the approach and tool (10 lines max)

Publications

Short list of publications on the approach (5 max)

For which activities are dedicated the means or tools (give a note from 0 to 3) :

	Author	Assessor 1	Assessor 2	Total
Data Management				
Function Management				
Requirement Management				
Version Management				
Other (give details below)				

According the results of this table, some of the following sections can be skipped.

2.3 Common criteria on secondary means and tools

This section discusses the common criteria of the means and tools according to the project requirements on tools and the results of T7.1.

2.3.1 Project and WP2 requirements

The objectives of this list of criteria is to check if the proposed means and tools meet the main criteria of the project: open-source approaches, usability, modularity, coverage of the objectives,...

According WP2 requirements, give a note for characteristics of the use of the tool (from 0 to 3) :

	Author	Assessor 1	Assessor 2	Total
Open Source (D2.6-02-074)				
Portability to operating systems (D2.6-02-075)				
Cooperation of tools (D2.6-02-076)				
Robustness (D2.6-02-078)				
Modularity (D2.6-02-078.1)				
Documentation management (D2.6-02-078.02)				
Distributed software development (D2.6-02-078.03)				
Simultaneous multi-users (D2.6-02-078.04)				
Issue tracking (D2.6-02-078.05)				
Differences between models (D2.6-02-078.06)				
Version management (D2.6-02-078.07)				
Concurrent version development (D2.6-02-078.08)				
Model-based version control (D2.6-02-078.09)				
Role traceability (D2.6-02-078.10)				
Safety version traceability (D2.6-02-078.11)				
Model traceability (D2.6-02-079)				
Tool chain integration				
Scalability				
User Friendliness				

2.3.2 Qualification

This section discusses how the tool can be classified according EN50128 requirements (D2.6-02-085). Some qualification shall be mandatory if the tool is involved to design a SIL4 software.

	Author	Assessor 1	Assessor 2	Total
Tool manual (D.2.6-01-42.02)				
Proof of correctness (D.2.6-01-42.03)				
Existing industrial usage				
Model verification				
Test generation				
Simulation, execution, debugging				
Formal proof				

Which scope of qualification is expected according EN50128 (section 6.7) ?

Score :

3 already qualified for this level

2 qualification possible to this level, but some elements shall be provided

---

0 qualification not recommended for this level

	Author	Assessor 1	Assessor 2	Total
class T1				
class T2				
class T3				

Other elements for tool certification

2.3.3   Complementarity with primary toolchain

The objectives of this list of criteria is to check if the proposed means and tools can be easily integrated to the primary toolchain.

2.3.3.1   Language

According to the decisions and the propositions of T7.1, how the mean and approach can be adapted to or can complete the chosen language and methods:

	Author	Assessor 1	Assessor 2	Total
SysML				
Scade method				
EFS language				
B Method				
C language				

SysML

How the means or tools can complete SysML ?

Scade, EFS, Classical B

How the means or tools can complete the current proposals for formal modeling language ?

C language

How the means or tools can complete or be adapted to SIL4 software in C language ?

2.3.3.2   Tools and platforms

According to the decisions and the propositions of T7.1, how the mean and approach can be integrated to or can complete the chosen tools and platforms:

	Author	Assessor 1	Assessor 2	Total
Eclipse				
Papyrus				
Scade				
EFS tools				
B tools				

Eclipse

How the means or tools can be integrated to the Eclipse platform ?

Papyrus

How the means or tools can complete Papyrus ?

Scade, EFS, Classical B

How the means or tools can complete the current proposals for formal modeling tools ?

2.4 Means and tools for data, function and requirement management

This section defines the criteria for the means and tools dedicated to data, function and requirement management. These activities are shared by the work packages WP3, WP4 and the activities dedicated to SSRS. These means and tools shall integrate the primary toolchain to complete its gap and facilitate the integration of different activities. First of all, they allow the management of a common repository of data, functions and requirements, shared between the models (from SSRS informal specification to code) and the verification and validation activities. Then, they shall support traceability of requirements between models and activities, and facilitate the verification of the traceability. Besides they shall support the design of SIL4 software with model comparison or document production facilities, and version management.

2.4.1 Management activities

Which activites, linked to help the management of SSRS definition and whole process are covered by the mean or tool ?

	Author	Assessor 1	Assessor 2	Total
Requirement capturing				
Requirement management				
Data management				
Function management				
Requirement traceability				
Model traceability				
Function architecture				
Version management				
Model comparison				
Documentation production				
Others (give details)				

2.4.2 Input Artifacts

Which artifacts are used as input of the mean or tool ?

	Author	Assessor 1	Assessor 2	Total
Informal description				
Structured description				
Spread sheet				
EFS model				
DSL				
Others (give details)				

2.4.3 Output Artifacts

Which artifacts are used as output of the mean or tool ?

	Author	Assessor 1	Assessor 2	Total
Informal description				
Structured description				
Spread sheet				
EFS model				
DSL				
Others (give details)				

2.4.4 Other Criterias

Comment. MPD : Todo Ideas welcomed !

2.5   Other comments

Comment.   This section is available for the author or the assessors to complete the description and criteria.

3 Conclusion

Comment. MPD : Todo

The sequel is let as an example is this early version.

Criteria to discuss here are those which concerns all the secondary tools as open-source issues, compatibility with primary tool-chain, compatibility with eclipse,...

This conclusion give a sum up of the evaluation results for each approach. The detailed results of each approach are given in the appendix.

Minus mark "-" means this criteria as not been evaluated for this approach.

Star mark "*" means this criteria has been difficult to evaluate for this approach.

The highest score is **9** and means that the criteria is fully respected, the lowest score is 0.

3.1 Main usage of the approach

Comment. MPD : Todo

The sequel is let as an example in this early version.

Score and results shall be corrected latter.

This section discusses the main usage of the approach.

According to the figure **??**, for which phases do you recommend the approach (give a note from 0 to 3) :

	GOPRR	ERTMSFormalSpecs	SysML with Papyrus	SysML with EA	SCADE	EventB	Classical B	System C	Petri Nets	GNATprove
Verification	5	1	7	9	3	9	3	2	6(9)	2 (3)
Validation	9	9	6	7	9	9	5	5	6(9)	3 (4)
Safety analysis	9	0	6	7	9	6	9	9	6(9)	6(9)
Data, function or requirement management	9	0	3	3	9	3	9	6	2 (3)	6(9)
Model or code transformation	9	0	3	3	9	3	9	6	2 (3)	6(9)

Appendix A: ProR

A.1　Author and Assessors

Author　Michael Jastram, Formal Mind GmbH

Assessor 1　First assessor of the approaches　%%Name - Company%%

Assessor 2　Second assessor of the approaches　%%Name - Company%%

A.2　Presentation

Name　Eclipse ProR

Web site　`http://eclipse.org/rmf/pror`

Licence　Eclipse Public License

Abstract

ProR is an open source tool, which is part of the Eclipse Requirements Modeling Framework (RMF) [1]. As the underlying data format, ProR uses ReqIF, a standard for exchange of requirements with other tools. It therefore provides interoperability with industry-strenth tools like Rational DOORS or MKS Integrity. It supports traceability within requiremenets, and traceability solutions for artefacts outside ProR exist.

Publications

- [5] contains a broad overview of ProR, as well as an approach to requirements structuring and model traceability.

- [4] looks at ProR as part of an Eclipse-based systems engineering environment.

- [6] suggests the integration of Topcased with ProR, an idea that has been picked up by the Topcased community.

- [8] is the openETCS benchmark, which contains a longer list of literature.

For which activities are dedicated the means or tools (give a note from 0 to 3) :

	Author	Assessor 1	Assessor 2	Total
Data Management	2			
Function Management	2			
Requirement Management	3			
Version Management	2			
Other (give details below)	3			

Author: *ProR clearly is a specialized tool for requirements management. While data and functions could be managed as plain text or numeric data in ProR, this would not be very productive.*

Data and Function Management. *However, ProR is capable of "pulling information together" from various sources. It would be feasible to show data and functions from Papyrus in-line in a ProR specification. Such an integration plug-in does not currently exist, but could be creeated with not too much effort.*

Version Management. *ProR does not contain version management, but works well with any underlying version management tool like Subversion of git, which already works today. However, the granularity of the versioning would be file-based, which is less than perfect. Versioning of individual requirements would be better. Further, higher-level versioning, like baselining, currently does not exist.*

Other. *Missing from the list is Traceability Management and Model Integration. ProR works very well in these two areas. Specialized traceability to Event-B exists and has been evaluated in [7]. At least two third-party traceability solutions exist: ReqCycle[1] is an open source solution for Toopcased (hosted on gitHub), and Yakindu Crema[2], which is commercial and closed-source.*

According the results of this table, some of the following sections can be skipped.

A.3   Common criteria on secondary means and tools

This section discusses the common criteria of the means and tools according to the project requirements on tools and the results of T7.1.

A.3.1   Project and WP2 requirements

The objectives of this list of criteria is to check if the proposed means and tools meet the main criteria of the project: open-source approaches, usability, modularity, coverage of the objectives,...

According WP2 requirements, give a note for characteristics of the use of the tool (from 0 to 3) :

---

[1] http://www.eclipsecon.org/france2013/sessions/reqcycle-coming-some-details
[2] http://www.guersoy.net/knowledge/crema

	Author	Assessor 1	Assessor 2	Total
Open Source (D2.6-02-074)	3			
Portability to operating systems (D2.6-02-075)	3			
Cooperation of tools (D2.6-02-076)	3			
Robustness (D2.6-02-078)	3			
Modularity (D2.6-02-078.1)	3			
Documentation management (D2.6-02-078.02)	2			
Distributed software development (D2.6-02-078.03)	2			
Simultaneous multi-users (D2.6-02-078.04)	2			
Issue tracking (D2.6-02-078.05)	1			
Differences between models (D2.6-02-078.06)	2			
Version management (D2.6-02-078.07)	2			
Concurrent version development (D2.6-02-078.08)	2			
Model-based version control (D2.6-02-078.09)	3			
Role traceability (D2.6-02-078.10)	1			
Safety version traceability (D2.6-02-078.11)	2			
Model traceability (D2.6-02-079)	2			
Tool chain integration	3			
Scalability	3			
User Friendliness	2			

Author: *Generally, the data structures of the underlying requirements model will not change, as they are based on an international standard, ReqIF 1.0.1 [9]. These data structures are powerful enough to model pretty much everything described here. However, just the existence of the right data structures does not mean that they are used properly. In many cases, it would be preferrable to integrate a separate tool (e.g. Mylyn for issue tracking), or to constrain the behavior of the tool programmatically. The bottom line is that for many questions the answer is: Yes, it is possible, but not very comfortable. For a comfortable solution, development resources are required.*

A.3.2 Qualification

This section discusses how the tool can be classified according EN50128 requirements (D2.6-02-085). Some qualification shall be mandatory if the tool is involved to design a SIL4 software.

	Author	Assessor 1	Assessor 2	Total
Tool manual (D.2.6-01-42.02)	2			
Proof of correctness (D.2.6-01-42.03)	*			
Existing industrial usage	2			
Model verification	1			
Test generation	*			
Simulation, execution, debugging	*			
Formal proof	*			

Author: The sections marked with asterisk are not applicable.

Model verification. *There is ongoing research in supporting V&V activities by establishing a traceability to the corresponding model [3].*

Which scope of qualification is expected according EN50128 (section 6.7)?

Score:

3 already qualified for this level

2 qualification possible to this level, but some elements shall be provided

0 qualification not recommended for this level

	Author	Assessor 1	Assessor 2	Total
class T1	2			
class T2	*			
class T3	*			

Author: I don't quite understand this section. ProR should be classified as T1 and has never been qualified.

Other elements for tool certification

A.3.3   Complementarity with primary toolchain

The objectives of this list of criteria is to check if the proposed means and tools can be easily integrated to the primary toolchain.

A.3.3.1   Language

According to the decisions and the propositions of T7.1, how the mean and approach can be adapted to or can complete the chosen language and methods:

	Author	Assessor 1	Assessor 2	Total
SysML	3			
Scade method	3			
EFS language	3			
B Method	3			
C language	*			

SysML

How the means or tools can complete SysML ?

Author: SysML itself provides a requirements element. However, this is little more than a box with text in it. We already realized a prototypical implementation, where the text in the SysML box is kept in sync with an attribute of a ProR specification.

A good SysML integration would go much further, allowing diagrams (e.g. state diagrams) to be inserted into the requirements text, or to color highlight symbols in the requirements text.

Scade, EFS, Classical B

How the means or tools can complete the current proposals for formal modeling language ?

Author: By providing a traceability between textual requirements and model elements.

C language

How the means or tools can complete or be adapted to SIL4 software in C language ?

Author: This is not clear. Conceivable are traceabillity to code, or incorporation of test results. How useful this would be and how much effort would be required depends on the underlying process.

A.3.3.2   Tools and platforms

According to the decisions and the propositions of T7.1, how the mean and approach can be integrated to or can complete the chosen tools and platforms:

	Author	Assessor 1	Assessor 2	Total
Eclipse	3			
Papyrus	2			
Scade	2			
EFS tools	2			
B tools	*			

Author: **B tools.** *For Event-B, an integration already exists. Classical B is much trickier, as has been described below.*

Eclipse

How the means or tools can be integrated to the Eclipse platform ?

Author: *ProR is Eclipse-based.*

Papyrus

How the means or tools can complete Papyrus ?

Author: *As Papyrus is based on EMF, it is possible to implement a traceability based on the EMF model. This has been done in the past (Event-B). It's fairly straight forward with textual elements, and a little more involved with graphical elements, like state diagrams.*

Scade, EFS, Classical B

How the means or tools can complete the current proposals for formal modeling tools ?

Author: *To create a traceability, ProR would need access to the other tool's model. If the tool writes XML, this is relatively easy. For a textual language like Classical B thinks are trickier, as we would need a parser. Further, in Classical B, not all elements have unique identifiers, making things even more tricky.*

A.4   Means and tools for data, function and requirement management

This section defines the criteria for the means and tools dedicated to data, function and requirement management. These activities are shared by the work packages WP3, WP4 and the activities dedicated to SSRS. These means and tools shall integrate the primary toolchain to complete its gap and facilitate the integration of different activities. First of all, they allow the management of a common repository of data, functions and requirements, shared between the models (from SSRS informal specification to code) and the verification and validation activities. Then, they shall support traceability of requirements between models and activities, and facilitate the verification of the traceability. Besides they shall support the design of SIL4 software with model comparison or document production facilities, and version management.

A.4.1   Management activities

Which activites, linked to help the management of SSRS definition and whole process are covered by the mean or tool ?

	Author	Assessor 1	Assessor 2	Total
Requirement capturing	3			
Requirement management	3			
Data management	1			
Function management	1			
Requirement traceability	3			
Model traceability	2			
Function architecture	1			
Version management	2			
Model comparison	2			
Documentation production	2			
Others (give details)	n/a			

A.4.2 Input Artifacts

Which artifacts are used as input of the mean or tool ?

	Author	Assessor 1	Assessor 2	Total
Informal description	3			
Structured description	3			
Spread sheet	3			
EFS model	1			
DSL	2			
Others (give details)	n/a			

Author:

> **Informal/Structured description.** *The only input format ProR currently accepts is ReqIF. However, ReqIF can be created with many tools, including Rational DOORS. At least for the time being, this could be used as a universal converter, allowing Word, Spreadsheets, plain text, etc. to be converted to ReqIF.*

> **EFS model.** *It would make little sense to "convert" EFS to ReqIF. Instead, it would make sense to create an adapter that would allow traceability to/from EFS.*

> **DSL.** *There have been prototypical implementations of XText (DSL framework) and ProR. Such an implementation allows to edit in the ProR-cells with a DSL editor.*

A.4.3 Output Artifacts

Which artifacts are used as output of the mean or tool ?

	Author	Assessor 1	Assessor 2	Total
Informal description	2			
Structured description	3			
Spread sheet	2			
EFS model	0			
DSL	2			
Others (give details)				

Comment. MPD : Todo Ideas welcomed !

A.5 Other comments

Comment. This section is available for the author or the assessors to complete the description and criteria.

Appendix: References

[1] Eclipse Requirements Modeling Framework. `http://eclipse.org/rmf`.

[2] Sylvain Baro and Jan Welte. Requirements for openETCS. Technical Report D2.6, OpenETCS, 2013.

[3] Stefan Hallerstede, Michael Jastram, and Lukas Ladenberger. A method and tool for tracing requirements into specifications. page 36 p, 2013. Science of Computer Programming.

[4] Michael Jastram. A Systems Engineering Tool Chain Based on Eclipse and Rodin. In *Forms/Format*, 2012. `http://www.stups.uni-duesseldorf.de/w/Special:Publication/jastram_forms_2012`.

[5] Michael Jastram. *Managing Requirements Knowledge*, chapter 16: The Eclipse Requirements Modeling Framework, pages 353–372. Springer, March 2013. `http://www.stups.uni-duesseldorf.de/w/Special:Publication/RMF_Mark_Book_Jastram_2013`.

[6] Michael Jastram and Andreas Graf. Requirement traceability in Topcased with the requirements interchange format (RIF/ReqIF). *First Topcased Days Toulouse*, 2011. `http://www.stups.uni-duesseldorf.de/w/Special:Publication/topcase-JaGr2011`.

[7] Matthias Güdemann. openETCS Event-B Benchmark. `https://github.com/openETCS/model-evaluation/blob/master/model/Event_B_Systerel/rodin-projects-github.pdf?raw=true`, 2013.

[8] Michael Jastram. openETCS Benchmark: Using Eclipse ProR for openETCS. `https://github.com/openETCS/model-evaluation/blob/master/management/ProR_FormalMind/pror_evaluation.pdf`, 2013.

[9] OMG. Requirements Interchange Format (ReqIF) 1.0.1. `http://www.omg.org/spec/ReqIF/`, 2011.

[10] Marielle Petit-Doche and WP7 Participants. D7.1: Report on the final choice of the primary toolchain. Primary Toolchain OETCS/WP7/D7.1, openETCS, July 2013.