ERTMSFormalSpec Workbench
Release Notes

Version 0.6.1

1 Table of contents

2 Revision history

Version	Date	Name	Description	Paragraphs
0.2.1	15/12/2010	Laurent Ferier	First version	All
0.3.1	31/01/2011	Laurent Ferier	Release of the WP2 snapshot 1	All
0.4.1	04/03/2011	Laurent Ferier	Release of the WP2 snapshot 2	All
0.5.1	06/04/2011	Laurent Ferier	Release of the WP2 snapshot 3	All
0.6.1	13/05/2011	Laurent Ferier	Release of the WP2	All

3 Introduction

This document holds is the release note for the ERTMSFormalSpecs Workbench (EFSW).

3.1 References

Document	Title	Date
[1]	EFSW Technical design	04/03/2011
[2]	EFSW User's Manual	04/03/2011

3.2 Terms and definitions

Term	Definition
ESFW	ERTMSFormalSpec Workbench

4 Package description

The ESFW version 0.5 is distributed as auto-installable executable. Installation is performed, by default in

C:\Program Files\ERTMSSolutions\ERTMSFormalSpecs

Installed files follow the structure

Location	File	Comment
.	unins000.dat and unins000.exe	Uninstallers
	EFSW_Release_Notes.pdf	These release notes
./bin	DataDictionary.dll DataDictionary.pdb	DLL used to handle the data dictionary
	Escslib.dll	ERTMS Solutions common tools
	GUI.exe GUI.pdb	The Workbench
	Importer.dll Importer.pdb	The utility used to import element in EFS
	Utils.dll Utils.pdb	EFSW Tools
	XmlBooster.dll XmlBooster.pdb	DLL used to read / write XML files
	Reports.dll Reports.pdb Docbook, fop-1.0, libxml2-2.7.7.win32, res	Files related to the report creation
./data		Files related to the report creation
./doc	EFSW_Technical_Guide.pdf	The technical guide of EFS
	EFSW_User_Guide.pdf	The user's guide for the workbench
./doc/specs	Subset-026.efs	The model file which contains the subset 26 requirements, the modelized elements of the subset 26 and the corresponding functional tests
	disable shunting.efs	The national rule which states that the Shunting mode should not be available
./src	DataDictionary	The source files used to generate DataDictionary.dll
	etc	Common configuration files
	GUI	The source files used to generate GUI.exe
	Importer	The source files used to import data in EFS
	Reports	The source fiels used to create reports
	Utils	The source files used to generate Utils.dll
	XmlBooster	The source files used to generate

		XmlBooster.dll

5 Installation & Execution

Installation of the EFSW is performed by executing the self-installer provided. This creates icons in the start menu to access the documentation, to launch the workbench and uninstall the EFSW.

When launching the workbench, no EFS file is opened by default. You have to select one or more, as explained in the user's guide.

The Microsoft Visual Studio solution is provided along with the source code which allows compiling and executing the Workbench in debug mode (for instance, to trace the execution of a specific routine, in order to understand it).

6 Known issues

This paragraph shows the issues known about the EFSW. These issues can be either bugs, change requests or planned evolution of the tool, according to the project management.

1145179828	Baseline updates
1145179836	Explain the SIL4 aspects of the exported model
1145179850	SysML integration
1145179878	White box, real-time analysis tool
1145179884	ShadowEVC
1145182928	6.4 Application of several actions which change the same variable
1145185572	select some S026 requirements related to STM for WP2
1145185580	Question: source code of EFSW delivery included or not?
1145185756	Application messages browser
1145189142	user mode vs admin mode
1145189154	What are optional features in S026
1145189160	simplified braking curves algorithm
1145189176	Plan EFS third-party tests with ERSA + get LOP 2.3.0d
1145189182	Shall draft a list of SIL4 requirements applicable to EFS model
1145189218	Identify all external functions needed by the EFS model
1145189222	Draft WP3+4+5 offer
1145189226	Track needs for Euroradio and/or SLL/STL software
1145189228	S076 traceability in EFSW
1145189230	S076 tests reuse from EFS model
1145192696	Implement 1 slide with EFS interfaces
1145205235	Request acknowledgement
1145212670	Impact analysis related to a rule change
1145215418	Cross check SPEC BUGs with ERA work
1145215434	Inconsistency in subset26 v3.0.0 in procedure change of train orientation
1145215486	Requirement allocation between Hardware and Software
1145215524	Provide feedback about EFSW SIL4 paper
1145215572	Historical information
1145215580	Unique identifiers for tables
1145215608	Reviewed by + review date
1145215654	Model review
1145215660	Report file format
1145215708	Track changes between deliveries
1145222100	EFSW static verif rules non-contradicting
1145230455	Add the field Defaul value for range
1145259868	Product Quality Plan / SIL0 traceability
1145272124	Should we include Subset-040 (engineering rules)
1145275416	There should be a mean to easily search through the model
1145275448	Rule checker
1145275458	Allow to use functions as first-class citizens

1145279728	Functional planning for DP3+4
1145310342	Review message definition for v3.2.0
1145310396	Extend usage display of a variable
1145318431	Sub-elements of some variables of type "Structure" are inaccessible
1145322953	Browse through variables
1145350424	XSD file is not up to date

7 Change Log

7.1 Version 0.2

Initial release of the EFS Workbench.

7.2 Version 0.3.1

First snapshot of the EFS Workbench for WP2. Main differences are the followings

- Review of the data model:
  - merge the content of the spec.xml and datadictionary.xml into a single file.
  - Replaced subsystems declaration by namespace, which can hold
    - Types
    - Variables
    - Functions
    - Procedures
    - Rules
  - Removed ProcedureSet and replaced them by state machines
  - Added functions to factorize common computations
  - Added procedures
  - Added meta data on types, variables, procedures, functions and rules to store
    - The requirement on which the model element is related
    - The status : implemented and verified
    - A comment
  - Added meta data on paragraphs
    - Reviewed
    - Implementation status
  - Replaced Trace by ReqRef
- Interface update
  - Review the test window for easier test processing
  - Allow to step back in a test
  - Fixed issues on pseudo-code colour coding
  - Fixed issues on tree node colour coding
  - Added navigations between types and instances of that type, and between variables and rules using that variable
- Interpreter update
  - Reviewed the concept of initial state of a state machine and removed the ‘is’ operator

7.3 Version 0.4.1

Second snapshot of the EFS Workbench for WP2. Main differences are the followings

- Review of the data model:
  - Allow namespace to enclose other namespaces
  - Added the definition of step translation
  - Added post-actions in a step
  - Allow to declare procedures in a structure to allow several instances of the same sub-system (e.g. several connections to EURORADIO, several DMIs, …)
- Import tests from subset 76
- Added a state diagram view
- Integrated Subset 26 v.3.2.0
- Generate reports for requirement coverage and test coverage
- Encode messages from Subset26 as structures
- Allow to modify / add new requirement to support the requirement review process
- The location of the .EFS file is no more hardcoded
- Created the installer to install the tool and verify system compatibility

7.3.1 Scope

This release implements 5% of the requirements specified in the Subset26, namely most of the start of mission and train trip procedures. A complete description is available in a separate report.

7.4 Version 0.5.1

Third snapshot of the EFS Workbench for WP2. Main differences are the followings

- Collection handling: Expression can now reference elements in a collection using the operators THERE_IS_IN, FORALL_IN, FIRST_IN, LAST_IN, MAP and REDUCE. Moreover, the function Available(x) allows to determine if there is an element ready to be used in the collection.
- Procedures can take several parameters and rules have been added. These rules are executed when the procedure is called.
- Allow to call runtime functions
- Reviewed actions to allow either variable modifications or procedure calls
- Reviewed preconditions to allow complex expressions
- State diagram :
  - review the display of state diagrams for initial states and overlapping transitions
  - improved the pattern to find transitions for state machines
- Added the type double
- The EFS Workbench allows to handle different several requirement sources. Rule disabling is also available to allow behaviour modification for project specific requirement source.
- The text of chapters 7&8 are present in subset26.efs
- Navigation between types & variables is available
- Newly created elements are placed at the beginning of the displayed collection.
- The workbench allows to review Requirements to adapt them to the required granularity: merge unnecessary split requirements and split big requirements into smaller ones.
- Metrics have been added in the specification view : number of selected requirements, number of implemented, number of implementable, number of implementation required
- Performances issues have been fixed
- XML files are saved as ASCII text files to avoid crashes when saving non-ASCII characters.
- A scope section has been added in the release notes
- Removed the documentation of the “is” operator, since it is no more used

7.4.1 Scope

This release implements 10% of the requirements specified in the Subset26, version 3.2.0. The following sections present the covered functional scope

7.4.1.1 Start of mission

All requirement of section 5.4 are implemented, except

- Degraded situations (section 5.4.6)
- Entry to mode considered as a mission (section 5.4.7)
- Two sub requirements of associated to state S10 which cause interpretation issues.

7.4.1.2 Procedure Train trip

All requirements of section 5.11 are implemented, except

- Degraded situations (section 5.11.4)
- State S010: because this references mode transitions which is not implemented
- State A035: because this references MA, which is not implemented
- State S120: because this references RBC acknowledgement, which is not implemented

7.4.1.3 Train position confidence interval and relocation

50% of the requirements in section 3.6.4 have been implemented. The other requirements depend on external data, which is not yet implemented. Each non implemented requirement has been annotated with a comment indicating why the implementation was not yet available.

7.4.1.4 Position reporting to RBC

40% of the requirements in section 3.6.5 have been implemented. The other requirements depend either on Level3, which is not completely specified yet; or depend on external data, which is not yet implemented

7.4.1.5 DMI depending on mode

All requirements of section 4.7 are implemented, except 10 requirements which depend on external data, which is not yet implemented.

7.5 Version 0.6.1

Delivery of the EFS Workbench for WP2. Main differences are the followings

- Creation of a data dictionary report
- Requirements can be related to tests
- Added the coverage report, the model report and the dynamic test report in the delivery
- Rule unique identifiers are accessible and rules can be searched for using appropriate tool
- Created the EMPTY value and allow to verify that a value is not empty
- Allowed named parameters in a function / procedure call
- Added more model checks and performed a complete model review
- Creation of the functional tests for the modelled elements

7.5.1 Scope

This release implements more than 10% of the requirements specified in the Subset26, version 3.2.0 and performs functional tests on 10% of the requirements specified in Subset26. The following sections present the covered functional scope

7.5.1.1 Start of mission

All requirement of section 5.4 are implemented, except requirements which require the creation of the procedure the “Shunting Initiated by Driver”.
All modelled requirements are tested using functional tests.

7.5.1.2 Procedure Train trip

All requirements of section 5.11 are implemented, except

- Degraded situations (section 5.11.4)
- State S010: because this references mode transitions which is not implemented
- State A035: because this references MA, which is not implemented
- State S120: because this references RBC acknowledgement, which is not implemented

All modelled requirements are tested using functional tests.

7.5.1.3 Train position confidence interval and relocation

95% of the requirements in section 3.6.4 have been implemented. The other requirements depend on external data, which is not yet implemented. Each non implemented requirement has been annotated with a comment indicating why the implementation was not yet available.
All modelled requirements are tested using functional tests.

7.5.1.4 Position reporting to RBC

90% of the requirements in section 3.6.5 have been implemented. The other requirements depend either on Level3, which is not completely specified yet; or depend on external data, which is not yet implemented.

7.5.1.5 DMI depending on mode

All requirements of section 4.7 are implemented, except 10 requirements which depend on external data, which is not yet implemented.
All modelled requirements are tested using functional tests.

7.5.1.6 National values

All requirements of section 3.A3.2 are implemented.

7.5.1.7 Train position

All requirements of section 3.6.1.3 are implemented.