OETCS/WP4/DescriptionOfWork

openETCS

ITEA2 Project
2012 – 2015

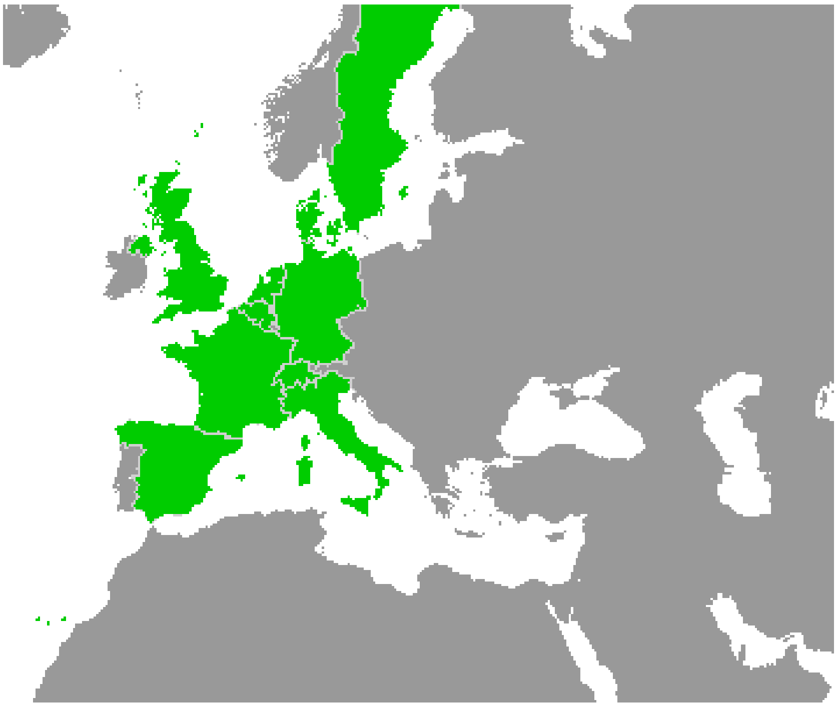
Work Package 4: “Validation & Verification Strategy”

openETCS Validation & Verification Strategy Work Package

Description of Work

Marc Behrens, , , Jens Gerlach, Hansjörg Manz, Jan Welte
and Cyril Cornu

March 2012



This page is intentionally left blank

openETCS Validation & Verification Strategy Work Package

Description of Work

Marc Behrens

WP4 Leader

WP4.1 Task Leader (Idetntification of tools and profile usage)

WP4.2 Task Leader (Verification & Validation of the formal model)

Jens Gerlach

WP4.3 Task Leader (Verification & Validation of the implementation code)

Hansjörg Manz, Jan Welte

WP4.4 Task Leader (Verification of the tools and processes)

Cyril Cornu

WP4.5 Task Leader (Internal Assessment)

Description of work

Prepared for ITEA2 openETCS consortium
Europa

Abstract: This work package will focus on the validation and verification of the model. At the very first beginning the target and requirements of the Verification & Validation strategy have to be described, i.e., what should be checked? Depending on the Modelling framework, the modelling language and formalization of the System requirements a strategy in form of a concept has to be defined how the consistency, coherence of the model as well as the coverage of system requirements will be transparently verified. Additionally, it is important to validate the model, i.e., to evidence the equivalence of the model and the ETCS system requirement specification (Subset-026 et al.). In other words the reliability and acceptance of the model has to be generated, e.g. nothing is lost or added or mutated and so on. Additionally, it has to be checked that the code is consistent with the model. The WP is intended to be performed in parallel with the modelling in order to apply the strategy and to generate feedback to the modelling process as well as to measure the quality and maturity of the model. Beside a subtask will manage the consideration of all relevant safety requirements (e.g. EN 50128/129) in the modelling process.

Table of Contents

Figures and Tables **Figures**

Tables

Introduction

%%To Be Defined%%

0.1 Techniques for Verification & Validation

Verification & Validation techniques can be roughly classified into *dynamic* and *static* techniques. Dynamic verification & validation techniques include:

- software-in-the-Loop
- model-in-the-Loop
- model-based testing
- monitoring
- coverage analysis

Static verification & validation techniques include:

- Checking of coding guidelines
- ...
- formal methods
  - model checking
  - deductive verification (theorem proving)
  - abstract interpretation

%%description on V&V classification non formal-> formal -> formal -> code & description%%

1 Identification of Tools and Profile Usage

%%To Be Confirmed%% The field of validation and verification features a huge set of different methods as well as suitable tools. Depending on the results of the modelling process in WP3 not all techniques and tools are suitable for this project. The objective of this task is to evaluate and choose the right techniques and tools for the verification and validation of the formal model as well as the implementation of the formal model developed by WP3 later in this project. Therefore the methodology from WP2 and the requirements and specifications to the tool chain from WP3 as well as the demonstrator in WP5 need to be analysed. The analysis should lead to the creation of a validation and verification plan used throughout the remaining tasks in this work package, which contains:

1. A selection of methods and a list tools suitable for applying the chosen V&V methods for
2. the formal model
3. the implementation of the formal model, i.e. the generated source code from the model
4. the tools themselves – if necessary

Furthermore the analysis result will be used to generate a review feedback for WP2.

1.1 Verification & Validation plan

%%To Be Defined%%

Table 1. T4.1 Inputs, Outputs and Deliverables

T4.1 Identification of Tools and Profile Usage				
Type	Description	Due Date	Due Month	status
D	*D 4.1* Report on V&V Plan & Methodology	Jul-2013	T0+13	started

2 Verification & Validation of the Formal Model

%%To Be Confirmed%% To ensure the correctness and consistency of the model and its implementation, the validation and verification has to be performed alongside with the modelling process. Thus these tasks will be performed repeatedly during WP3 and will provide feedback to it.

This task handles the verification and validation of the formal model. This will be accomplished by applying the methods chosen in WP4 Task 1 onto the formal model from WP3 using the tool chain developed in WP3. Depending on the chosen approach and applicable tools a variety of verification methods can be applied like:

1. proof technique
2. model checking technique
3. Simulation

As the verification and validation is part of the development chain, this task is being applied iteratively in parallel to the development of the formal model in WP3. The feedback given should focus on the consistency and correctness of the model and development process in WP3. The results of this task are the verification and validation specifications (how to perform the V&V on the formal model), the basic materials (the actual tests cases, checklists, etc.) and the V&V report on the formal model.

Table 2. T4.2 Inputs, Outputs and Deliverables

T4.2 Verification & Validation of the Formal Model				
Type	Description	Due Date	Due Month	status
D	*D 4.4* Final report on Verification & Validation of the model	Jul-2013	T0+13	not started

3 V&V of the Implementation & Code

The objective of this task is to verify and validate the actual implementation of the formal model. Therefore the tool chain from WP3 will be used to apply the chosen methods from WP4 Task 1 onto the implementation of the formal model from WP3. The chosen combination of methods and tools in WP4 Task 1 can result in a wide variety of techniques to be used:

1. Software-in-the-Loop
2. Model-in-the-Loop
3. Model-based testing

4. Deductive verification

5. Monitoring

6. Static analysis

Analogue to WP4 Task 2 the verification and validation of the formal model implementation is part of the development chain. Therefore this task runs parallel to the development of the formal model in WP3, and is being applied iteratively. Therefore feedback regarding the validity and correctness is delivered back to the development process in WP3. The results of this task are the verification and validation specifications (how to perform the V&V on the formal model implementation), the basic materials (the actual tests cases, checklists, etc.) and the V&V report on the implementation of the formal model.

As first steps the relevant properties and techniques concerning the code and implementation are to be identified,

3.1 Software Properties

Here we list properties that we think are most relevant for verification & validation:

- functionality
- robustness (absence of runtime errors)
- performance
- real time behaviour
- dataflow
- absence of deadlocks

%%To Be Defined%%

Table 3. T4.3 Inputs, Outputs and Deliverables

T4.3 V&V of the implementation code				
Type	Description	Due Date	Due Month	status
D	*D 4.4* Final Report oon tn Verification & Validation of the code/ Implementation	Jun-2015	T0+36	not started

4 Verification of the Tools and Processes

%%To Be Confirmed%% The software will be developed according to the guidelines specified in the CENELEC Standard 50128. Each of the Lifecycle stages (SW Requirement Specification, SW Design, SW Coding, etc.) must be fully documented and simultaneously verification and validation tasks must be performed. In this task the safety management team draws a safety plan to identify the safety management structure, safety related activities and safety approval milestones. A hazard log will be created and maintained throughout the whole development process. In addition, the safety plan will include plans for verifying that each development phase meets its safety requirements. The safety plan also describes (among others):

1. Roles, responsibilities and competences of the involved bodies

2. Safety-related deliverables with milestones
3. Procedures of preparing the safety case
4. Procedures for maintaining safety documents

All safety principles followed in the development process will be described along with documented quantitative analyses. Evidences of technical safety shall describe the safeguards used for individual safety properties. The V&V reports are to be referred in this part. Concerning the applied tools, 3rd parties may be engaged to perform the V&V. Certainly, the respective results will be referred to in the safety case.

Table 4. T4.4 Inputs, Outputs and Deliverables

T4.4 Verification of the Tools and Processes				
Type	Description	Due Date	Due Month	status
D	emphD 4.4 Final report concerning the Safety Case	Jun-2015	T0+36	%%To Be Defined%%

5 Internal Assessment

One of the major point for a SIL4 compliant Software is the Whole Software Development Project Assessment by a Safety Authority (e.g CERTIFER in France, TÜV in Germany). As none of these companies are involved in openETCS Software Development assessment, the Internal Assessment objective is to simulate a real Assessor's tasks during the whole Open ETCS Software Development activities.

An assessment is a ¨ Process of analysis to determine whether software, which may include process, documentation, system, subsystem hardware and/or software components, meets the specified requirements, and to form a judgment whether the software is fit for its intended purpose. Safety assessment is focused on but not limited to the safety properties of a system.¨

5.1 Assessment tasks

The Assessor shall write a Software Assessment Plan. It is like an assessment process which is linked to the software development process. More precisely, he shall explain the tasks needed to assess the software of the project OpenETCS.

Note: The Verifier shall write a Software Assessment Verification Report, as required in the standard EN50128, to verify in the first time that the Software Assessment Plan meets the general requirements for readability and traceability.

During the software development, he shall evaluate the software verification and validation activities. We propose that the Assessor intervenes at least seven times during the software development process (this is equivalent to one time at least by Work Product).

Note: the numbers of WPs are not given in the chronological order, e.g. WP1 is performed during all the development process and WP5 occurs before the end of WP4.

During WP1: Project Management.

The Assessor is able to assess:

- The Quality Assurance

- The capability of the Project Manager and the quality of his deliverables

The Assessor shall assess the Software Quality Plan. We propose that he gives a formal approval of this document.

During WP2: Requirements for Open Proof.

The Assessor is able to assess:

- The System requirements specification, including:
  - functions and interfaces;
  - application conditions;
  - configuration or architecture of the system;
  - hazards to be controlled;
  - safety integrity requirements;
  - apportionment of requirements and allocation of SIL to software and hardware;
  - timing constraints
- The software requirements specification,
- The software architecture and design specification,
- The software component specification,
- The personnel key roles, responsibilities and competence,
- The Quality Assurance

Nevertheless, he shall assess the implementation of both activities and deliverables of WP 2.

During WP3: Modeling of (part of) ETCS specification.

The Assessor shall evaluate the software implementation respectively the software modeling. Furthermore, he is able to assess:

- A part of the lifecycle and the documentation,
- The Quality Assurance,
- The personnel roles and responsibilities and competence.

The Assessor shall assess the implementation of both activities and deliverables of WP 3.

During WP4: Validation & Verification Strategy.

The Assessor shall assess:

- the Software Verification Plan and the Software Validation Plan,
- the Quality Assurance.

We propose that he gives a formal approval on these documents. He shall mainly evaluate the verification activities and the implementation of both activities and deliverables of the WP 4.

During WP5: Demonstrator.

The Assessor shall assess the specific openETCS software. Indeed, before the beginning of the validation activity (WP4), the Assessor shall assess the Software Integration Test Report to give or not the approval for software validation. This point is the Validation first step (the previous steps are related to the verification).

During WP6: Dissemination, Exploitation and Standardization.

The Assessor shall verify that the software maintenance plan is written and compliant with the software safety integrity level (SIL4).

During WP7: Language, ToolChain and Opensource Ecosystem.

The Assessor shall assess the developed tool chain according to Tool class T3 of EN 50128:2011. The other tools (T2) have to be assessed as well, but the effort is liter regarding the T3 assessment effort.

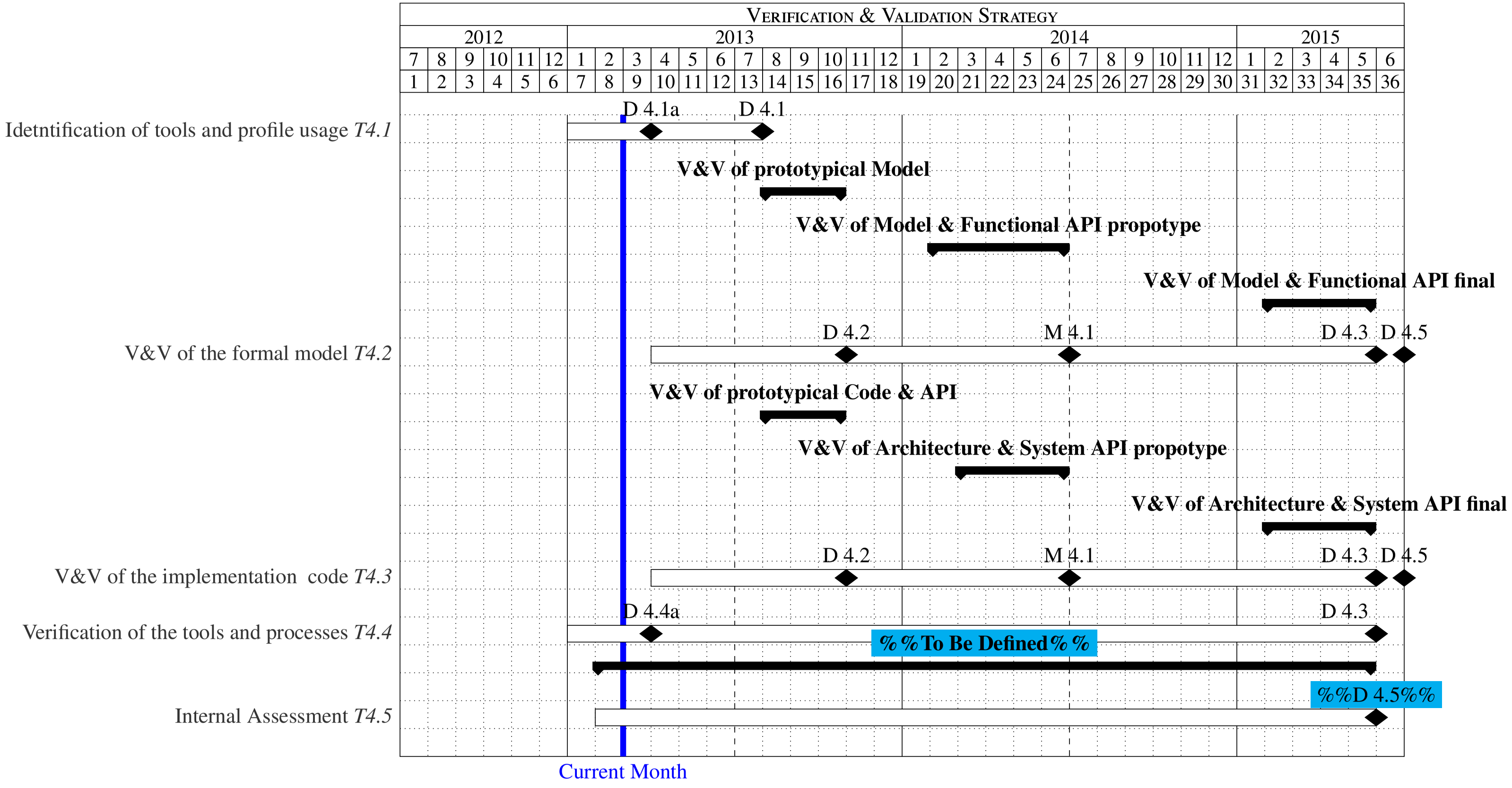
At the end of the software development process, the Assessor shall perform a final assessment. Indeed, he shall evaluate that the lifecycle processes and products resulting are such that the software is of the defined software safety integrity level and fits for its intended application. All the steps of assessment performed during the software development process shall be gathered in the Software Assessment Report. This report could be updated all along the process.

Note: the Software Assessment Verification Report permit to verify the internal consistency of the Software Assessment Report.

Table 5. T4.5 Inputs, Outputs and Deliverables

T4.5 Internal Assessment				
Type	Description	Due Date	Due Month	status
D	*D 4.5* Quality recommendation to prepare the Assessment	29/03/2013	March	

6 GANTT chart


VERIFICATION & VALIDATION STRATEGY
Idetntification of tools and profile usage T4.1
D 4.1a
D 4.1
V&V of prototypical Model
V&V of Model & Functional API propotype
V&V of Model & Functional API final
V&V of the formal model T4.2
D 4.2
M 4.1
D 4.3
D 4.5
V&V of prototypical Code & API
V&V of Architecture & System API propotype
V&V of Architecture & System API final
V&V of the implementation code T4.3
D 4.2
M 4.1
D 4.3
D 4.5
Verification of the tools and processes T4.4
D 4.4a
D 4.3
%%To Be Defined%%
Internal Assessment T4.5
%%D 4.5%%
Current Month