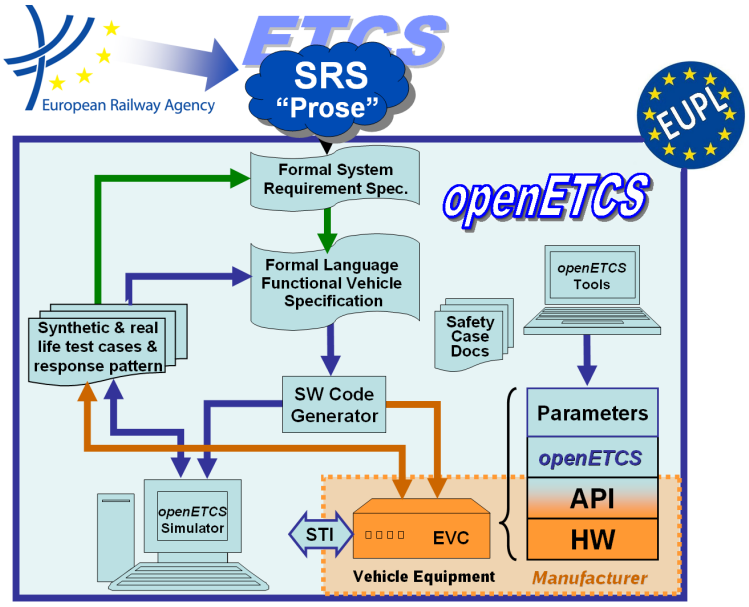
ITEA2 Project
Call 6 11025
2012 – 2015

Work-Package 1: “Management”

# Project Quality Assurance Plan

Izaskun de la Torre

December 8, 2015




Funded by:
Federal Ministry of Education and Research
Région de Bruxelles-Capitale
GOBIERNO DE ESPAÑA
MINISTERIO DE INDUSTRIA, ENERGÍA Y TURISMO

This page is intentionally left blank

**Work-Package 1: “Management”**

**OETCS/WP1/D1.3.1**
**December 8, 2015**

# Project Quality Assurance Plan

Izaskun de la Torre

SQS
Avenida Zugazarte 8
48930 Getxo, Spain

Description of work

Prepared for openETCS@ITEA2 Project

# Contents

# Figures and Tables

## Figures



## Tables

## Document History

**Table 1. Documentation History**

| Version | Date | Chapters modified | Reason | Name |
|---|---|---|---|---|
| 0.0.0 | 15.11.2012 | All | First Steps on frame evaluation | Rico Kaseroni (DB)<br>Peyman Farhangi (DB) |
| 0.1.0 | 27.11.2012 | All | First Steps on Content | Rico Kaseroni (DB)<br>Jan Welte (TUBS)<br>Peyman Farhangi (DB)<br>Matthias Kuhn (DB) |
| 0.1.1 | 29.11.2012 | All | Optimaziation of document structure, Revision of Chapters according to EN 50128, Merging with project specific tasks | Stephan Jagusch (AEbt)<br>Rico Kaseroni (DB)<br>Cyril Cornu (All4tec) |
| 0.2.0 | 30.11.2012 | Baseline Requirements for certification | Extention of Chapter according to EN 50128 | Jan Welte (TUBS)<br>Rico Kaseroni (DB) |
| 0.3.0 | 19.12.2012 | All | Extention of Chapter 0, 1, 2, 3 | All4tec, DB, SQS |
| 0.4.0 | 11.01.2013 | All | Extention to existing and further Chapters | All4tec, DB, SQS |
| 0.6.0 | 28.01.2013 | All | intellectual property (IP) Clean | Rico Kaseroni (DB)<br>Cyril Cornu (All4tec) |
| 0.6.1 | 29.01.2013 | Scrum | Contribution | Bernd Hekele (DB) |
| 0.7.0 | 01.02.2013 | All | More Content | Rico Kaseroni (DB) |
| 0.8.0 | 02.02.2013 | All | Jungle Content -> Smooth | Rico Kaseroni (DB) |
| 0.9.0 | 06.02.2013 | All | Review on 0.8.0 Version | Klaus-Rüdiger Hase (DB) |
| 0.9.1 | 07.02.2013 | Scrum | Optimization | Bernd Hekele (DB) |
| 0.9.2 | 07.02.2013 | All | Restructuring | Rico Kaseroni (DB) |
| 0.9.3 | 11.02.2013 | 1-, 2-, Last Chapter Appendices A and C | Graphic Figure 1, Definition of openETCS Process IP clean Job | Rico Kaseroni (DB) |
| 0.9.4 | 12.02.2013 | All | Optimization | Rico Kaseroni (DB) |
| 0.9.4.5 | 15.02.2013 | Chapter2 | System Testing | Rico Kaseroni (DB) |
| 0.9.4.6 | 15.02.2013 | ALL | Optimization | Rico Kaseroni (DB) |
| 0.9.5 | 22.02.2013 | ALL | Restructuring & Optimization | Rico Kaseroni (DB) |
| 0.9.5.1 | 01.03.2013 | ALL | LaTeX conversion | Peter Mahlmann (DB) |
| 0.9.5.2 | 04.03.2013 | ALL | LaTeX Optimization | Rico Kaseroni (DB) |
| 0.9.5.3 | 10.04.2013 | ALL | New Structure | Izaskun de la Torre (SQS) |

**Table 1 – continued from previous page**

| Version | Date | Chapters modified | Reason | Name |
|---|---|---|---|---|
| 0.9.5.4 | 20.04.2013 | ALL | New Content | Bernd Hekele (DB)<br>Jan Welte (TUBS)<br>Marielle Petit-Doche (Systerel)<br>Stefan Rieger (TWT)<br>Izaskun de la Torre (SQS) |
| 0.10.0 | 26.08.2013 | chapter 1.2 and 8.2 | New Content | Bernd Hekele (DB)<br>Stefan Rieger (TWT) |
| 0.10.0 | 17.09.2013 | chapter CAT1 and CAT2 | Updated Content | Bernd Hekele (DB) |
| 0.10.1 | 25.09.2013 | Chapter 1 | Terminology | Jan Welte (TUBS) |
| 0.10.2 | 14.11.2013 | Chapter 6.3, 7.1, 8.1 | Updated Content | Izaskun de la Torre (SQS) |
| 0.10.3 | 21.11.2013 | Chapter 3.3, 5.2, 5.3, 6.2 | Updated Content | Izaskun de la Torre (SQS) |
| 0.10.4 | 30.01.2014 | All | Fixed issues 4 and 11 of internal assessment | Izaskun de la Torre (SQS) |
| 0.10.5 | 18.02.2014 | All | Fixed issues 3, 10 and 12 of internal assessment | Izaskun de la Torre (SQS) |
| 0.10.6 | 26.02.2014 | 5.2.1, 5.2.2, 11 | Addition of new sections | Izaskun de la Torre (SQS) |
| 0.10.7 | 11.03.2014 | 3.2.1, 11.2 | An outline and new content | Izaskun de la Torre (SQS) |
| 0.10.7 | 16.04.2014 | Appendix E | Methods and Tools Update to cover Modeling needs | Bernd Hekele (DB) |
| 0.10.7 | 08.05.2014 | 5.2.2.1<br>5.2.2.2 | Add information about tools in the tool chain | Cécile Braunstein (Uni Bremen) |
| 0.10.7 | 14.05.2014 | 3.2.2 | Complete the openETCS tool chain lifecycle | Cécile Braunstein (Uni Bremen) |
| 0.10.8 | 25.06.2015 | all | Quality review | Abdelnasir Mohamed (AEbt) |
| 0.10.9 | Chapter 1.5 | 13.07.2015 | Update chapter | Abdelnasir Mohamed (AEbt) |
| 0.10.10 | 17.07.2015 | Chapater 2 | Small corrections | Abdelnasir Mohamed (AEbt) |
| 1.0.0 | 26.10.2015 | Chapter 3, 4 and 5 | Restructuring of chapter 3, small corrections | Abdelnasir Mohamed (AEbt) |
| 1.0.1 | 07.12.2015 | Chapter 1.6.2 and 1.6.3 | Generation of glossaries and abbreviation | Jan Welte (TU Braunschweig) |

# 1 Introduction

## 1.1 Purpose

The purpose of the QA Plan is to define the processes, methods and tools that will be used to develop the openETCS project meeting ITEA requirements, following Open Source principles and practices and applying the agile methodology with SCRUM. Besides, two of the project outcomes, the openETCS software, the openETCS Tool Chain, will have to comply with the CENELEC EN 50128:2011 requirements [3].

Due to the nature of the openETCS project (Research and Development (R & D) an EU project with a complex list of project outcomes and deliverables), the QA Plan is specifically designed to provide a complete, consistent and integrated view of the development process at both project and product level (i.e. the development life-cycle is described partially in two different deliverables, the QA plan should manage to provide an integrated overall view).

The QA Plan also describes the activities to monitor and manage quality in all aspects of the project:

- Defining and ensuring that all processes and products are compliant with the corresponding standard and requirements, according to the required system/software safety integrity level
- Identifying non conformances
- Providing timely quality status feedback to management and affected personnel
- Ensuring noncompliance issues are addressed

Therefore, it describes the QA functions, responsibilities and specific monitoring and control activities.

## 1.2 Goals of the openETCS project

The main goals and deliverables of the openETCS project are:

**A semi-formal reference specification for the ETCS requirements and architecture, completed by strictly formal models of sub-parts**

The first goal of the project is to propose a semi-formal specification of the ETCS on-board unit (OBU) functionalities according to UNISIG SUBSET-026 [10], baseline 3.

The purpose of this semi-formal specification is:

- to enhance the understanding of the subset;
- to have an artifact for testing and analyzing purpose at system level;
- to provide information on the completeness and soundness of the SUBSET-026;
- to be used as a semi-formal reference of specification for the implementation of an on-board unit (by the openETCS project team and by industrial actors);

The output is a model, at least semi-formal, which can be extended to many formal approaches (SCADE, Simulink, B tools, openETCS tool chain, ...) that can be given to all railway actors, and if possible associated to SRS documents in the ERA database.

Thus, strictly formal models can be designed from this semi-formal model, which allows formal proofs of sub-parts of SUBSET-026. This will allow improving the understanding of the system and will provide elements for verification and validation using formal proof.

The final goal is that industrial actors work with this model instead of the natural language specification. The objective is to cover as much as possible of the functionality of the on-board unit described in SUBSET-026 and to show the capabilities of analyses of a complex system using formal approaches.

### Definition of the safety case concept for the full model and application on a subset of the on-board unit

The safety strategy and the safety case concept required for the full validation of the product, compliant to the CENELEC standard EN 50126, shall be taken into account in all steps of the specification and design process. This will allow industrial actors to reuse the models and processes to develop certifiable products.

In particular the definition of the process shall take into account specification as well as verification and validation of the safety properties on the models. The outputs of WP4 (safety plan, safety case concept, verification plan and validation plan) will complete the description of the safety process.

### Providing a tool chain and process/methodologies for developing an on-board software that can fulfill the CENELEC requirements for SIL4 software

The design process of the system and the associated tools of the tool chain shall be suitable to provide a certifiable product. For this purpose all steps of the process and the choice of the methods and tools shall be justified to ensure a safe approach to build an ETCS system.

The full safety process required to make openETCS *certifiable* according to EN 50126, EN 50128 and EN 50129 shall be described in detail. The safety process will detail precisely which activities are required, why they are required, and the choices that are made to claim that a safe design process is guaranteed.

The use of formal methods, supported by tools, is highly recommended in this safety process for specification, design, verification and validation of the certifiable product.

The tool chain should include model editors, code generators, verification tools (including formal provers), validation tools (including test generators, simulators,..), document generation, version management, maintenance facilities, ....

### Provide an executable software package generated from the specification of on-board ETCS

An executable software of the specification shall be provided, as well as a non vital implementation of the on-board unit for laboratory test, simulation and as reference. It will be a non-vital implementation, able to be executed in real-time and in interaction with other components.

## 1.3 Intended Audience

The QA Plan addresses all the stakeholders who are in the position to interact with openETCS project.

| Audience | Use | Role |
|---|---|---|
| openETCS Consortium Members | It provides information and access to the QA procedures and guidelines to be followed/applied during the different phases of the project development life-cycle.<br>It provides a consistent and integrated view of the development process followed. | Consultation Reviewer Contributor or Committer |
| openETCS Quality Manager | It contains the quality targets to be achieved and the corresponding QA activities to be implemented and monitored. | Author |
| Independent Assessors | It shows the SQA strategy conceived and the one effectively implemented. | To assess whether the project results comply to CENELEC standard |
| Open Source Community (Users, Adopters, Contributors, Committers) | Provision of information and access to the QA related procedures and guidelines implemented.<br>Provision of information on the on-going projects<br>Provision of guidelines on how to participate to any of the projects | For consultation and/or engagement |
| ITEA Representative | The QA Plan constitutes a Project Deliverable | For evaluation |

**Table 2. Intended Audience**

## 1.4 Evolution

The first version of the document, prepared at the beginning of the project, will be updated regularly with the evolution of the openETCS project. The methods and tools to be applied during the development of the openETCS software products will be decided based upon the results of the research activities carried out during the project.

The QA Plan document will incorporate such decisions as they are taken with a proper justification of their appropriateness to meet the quality targets. The QA manager will guarantee the document is up to date.

The QA Plan document has been conceived as a reference document. This means that detailed descriptions of procedures, guidelines, methods and/or tools will not necessarily be included in the document but adequately referenced *(chapter 1.5)*. The authors of such documents and/or Wiki pages will be responsible for keeping them updated. The QA manager will monitor such activities and will guarantee changes are appropriately reflected in the QA Plan.

The QA Manager will maintain the QA Plan backlog [6] [Wiki].

Major revisions of the QA Plan will be accomplished by the Committers to the Management Project. Minor review process will be done with the participation of the external community, following procedure [13].

## 1.5 References, Guidelines and Standards

| Standards | | | | |
|---|---|---|---|---|
| Internal Code | Name | Version/ Edition/ Date | Repository | Responsible |
| [1] | EN 50126:2000 | | - | CENELEC |
| [2] | EN 50129:2003 | | - | CENELEC |
| [3] | EN 50128:2011 | | - | CENELEC |
| [11] | ISO 9001 | | - | International Organization for Standardization |
| [9] | SUBSET-023 | 3.0.0 | SSRS | UNISIG |
| [10] | SUBSET-026 | 3.3.0 | SSRS | UNISIG |

**Table 3. Standards**

| References | | | | |
|---|---|---|---|---|
| Internal Code | Name | Version/ Edition/ Date | Repository | Responsible |
| [25] | Full Project Proposal (FPP) | 4.0 | management | Klaus-Rüdiger Hase, Peter Mahlmann |
| [26] | Software Configuration Management Plan | 0.0.2 | governance | Jürgen Weiss |
| [19] | Project Co-operation Agreement | 02e | management | Bernd Hekele |
| [18] | OpenECTS IP Policy | 0.1 | ecosystem | Bernd Hekele |
| [17] | openETCS Assessment Plan | 1.0 | internal-assessment | Frédérique Vallée, Norbert Schäfer |
| [23] | openETCS Validation & Verification Plan | 3.01 | validation | Marc Behrens, Hardi Hungar |
| [6] | QA Plan Backlog | N/A | governance | Izaskun de la Torre |
| [7] | Traceability Matrix | 0.1.0 | governance | Izaskun de la Torre |
| [**?** ] | openETCS Hazard and Risk Analysis and Safety Case Methodology | 0.10 | validation | Jan Welte |

**Table 4. References**

| Procedures | | | | |
|---|---|---|---|---|
| Internal Code | Name | Version/ Edition/ Date | Repository | Responsible |
| [13] | Review Process | 0.2.1 | governance | Ainhoa Gracia |
| [14] | Revision Process | 0.2.1 | governance | Ainhoa Gracia |
| [5] | Change/Problem Management Process | 0.1.0 | governance | Izaskun de la Torre |
| [16] | Grieving Handling Process | | governance | Bernd Hekele |
| [21] | Committer Approvement Process | 2012-11-14 | ecosystem | Jonas Helming |
| [22] | openETCS Development Process | 2012-11-14 | ecosystem | Jonas Helming |
| [8] | Training Process | 0.1.0 | governance | Izaskun de la Torre |
| [12] | Document Control Process | 0.1.0 | governance | Ainhoa Gracia |

**Table 5. Procedures**

| Guidelines | | | | |
|---|---|---|---|---|
| Internal Code | Name | Version/ Edition/ Date | Repository | Responsible |
| [15] | Contribution guidelines | 01 | ecosystem | Bernd Hekele |
| [20] | Committer Election Guideline | 2013-02-06 | ecosystem | Jonas Helming |
| [24] | openETCS Publishing Guideline (see also Sct. 10) | 2013-07-15 | Dissemination | Stefan Rieger |
| [29] | Expert Election Guideline | N/A | governance | *To be defined* |

**Table 6. Guidelines**

| Templates | | | | |
|---|---|---|---|---|
| Internal Code | Name | Version/ Edition/ Date | Repository | Responsible |
| [27] | Competence Matrix Template | 0.1.0 | governance | Jan Welte |
| [28] | Expert database Template | N/A | governance | *To be defined* |

**Table 7. Templates**

## 1.6 openETCS Terminology

The openETCS project deals with topics from different domains like railway vehicles, signaling systems, formal methods and tool development. As every of these domains has their specific

terminology, the identification of all relevant terms and abbreviations is an important part of the openETCS development process. Respectively a terminology process has been defined which collects, defines, analyses and distributes the relevant terminology for all parts of the openETCS project.

### 1.6.1 Terminology Process

The openETCS terminology process is based on the main openETCS development environement mainly github and Latex. In addition the iglos (https://www.iglos.de/iglos/) environment is used to model and manage terminology relations. The overall process contains the following steps:

1. Term proposals with definition, source and relation proposals via https://github.com/openETCS/glossary/issues or through extraction from project documents
2. Inclusion of term, definition, source and relation proposals into iglos, to manage the terminology work and allow analysis of the terminology (e.g. for consistency)
3. Export of terms and abbreviations and their information as definitions, sources and relations from iglos using a csv-file
4. Transformation of the csv-file into a latex glossary (https://github.com/openETCS/glossary/blob/master/Latex_Glossary/openETCS-Latex-Glossary.tex)for all project documents
5. All glossary files and their documentation is provided in the github glossary repository and continuously updated

Depending on the needs further extractions from the iglos database can be created providing terminology for specific openETCS activities.

The glossary and the list of abbreviations respectively acronyms is then added to any latex document by using the glossaries package. The latex commands, which they do support this can be found in the *User Manual for glossaries.sty v3.07* or in the short description in the glossary wiki at https://github.com/openETCS/glossary.

The following subsections list the important glossary terms and the abbreviations used in this document.

### 1.6.2 Glossary

| | |
|---|---|
| assessor | person or agent appointed to carry out the assessment. |
| designer | entity that analyses and transforms specified requirements into acceptable design solutions which have the required safety integrity level. |
| formal methods | Formal methods are system design techniques that use rigorously specified mathematical models to build software and hardware systems.. |

| | |
|---|---|
| integrator | entity that carries out software integration. |
| intellectual property | part of intellectual assets owned by a person or organization as a result of creations of the mind or the intellect. |
| validator | person or agent appointed to carry out validation. |

### 1.6.3 Abbreviations

| | |
|---|---|
| ASR | assessor. |
| DES | designer. |
| ETCS | European Train Control System. |
| HR | Highly Recommended. |
| IMP | implementer. |
| INT | integrator. |
| IP | intellectual property. |
| OBU | on-board unit. |
| R & D | Research and Development. |
| SCMP | System Configuration Management Plan. |
| SIL | safety integrity level. |
| SW | software. |
| TSI | Technical Specification for Interoperability. |
| TST | tester. |
| V & V | Verification and Validation. |
| VAL | validator. |
| VER | verifier. |
| WP | Work Package. |

# 2 Project Organization

openETCS is a cooperative European-ITEA project. The project plan (objectives, work plan schedule, role of the partners, project organization) is described in the [25] FPP document, which is updated regularly (at least yearly). The project is accomplished according to the Project Co-operation Agreement (PCA) [19] signed by the partners.

The organization of the project has to meet the following constraints and challenges to succeed:

1. As an ITEA project, the project has to meet requirements imposed by the ITEA Office that affect both the organization and the outcomes of the project.
2. As an ITEA project, the effective involvement of the partners is sometimes hampered by external constraints (i.e. local financing, local approvals) so mechanisms to guarantee the "required competence" is available when needed are to be implemented. Besides, openETCS operates in a regulated environment where demonstrating the competence of the personnel assigned to the different activities is required.
3. Some of the results (software & tool chain) have to be certifiable; CENELEC SIL4 requirement [3] have to be followed and the corresponding evidence provided.
4. As an open source project, Open Source principles will be respected; high degrees of engagement from the community are intended.
5. As it is the intention to apply SCRUM, the appropriate responsibilities and mechanisms have to be implemented

The following chapters shows the mechanisms implemented at organizational level to guarantee the above mentioned objectives are achieved.

## 2.1 openETCS project organisation

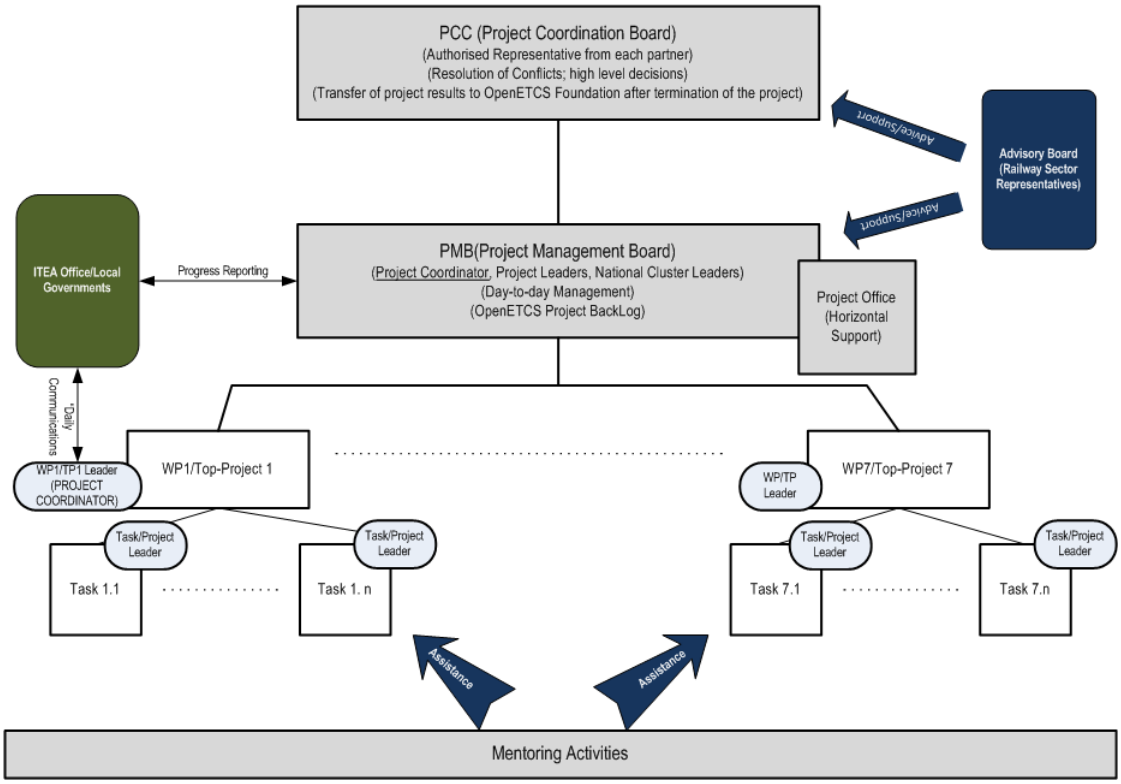



**Figure 1. openETCS Project Structure**

### 2.1.1 Compliance with ITEA Requirements

ITEA rules are documented in the ITEA2 Frame Agreement.

Compliance to ITEA Requirements is achieved by means of:

- The appointment of a Project Coordinator (DB, WP1, supported by the Project Office) who leads the project and is responsible for the communications with the ITEA representatives.
- The appointment of a Local Coordinator per country, National Cluster Leader, who reports to the corresponding National Authorities of the progress of the local partners
- A signed PCA where cooperation rules and principles and working structures are agreed by all the partners.
- An openETCS Foundation e.V. which guarantees sustainability of the project results once the project is finished.

### 2.1.2 Compliance with Open Source Principles

Compliance to the Open Source Principles and related objectives is achieved by means of:

- An openETCS IP Policy and Procedures [18]

- An openETCS Development Process [22] [Wiki] based in the Eclipse Development Process [4], designed to promote dynamism in the development and openness. All the guidelines are maintained and available at the openETCS Ecosystem project [Wiki]:
  - The openETCS project is conceived as a project of projects organized in a hierarchical manner, where the Work Packages, as defined within the Work Program [25], are considered Top-Level Projects with their own charter. The so-called Tasks are projects, sub-projects of the corresponding Top-Level Project.
  - Anyhow, new projects can be launched, if needed and approved; existing projects can be archived, if they become inactive. Therefore the final structure of the openETCS project will very much depend on its evolution.
  - The list of openETCS projects with information on their status is available in [governance]
  - Any project (independently to its position in the hierarchy, and type) has its project leader, scope and maintains its own resources. The project leader is not only responsible to guarantee progress towards the scope of the project but to promote that the most appropriate community is engaged in the project life-cycle with openness and transparency. This community includes committers, contributors, users and adopters.
  - Every Top-project/Work Package (WP) has its own repository under the responsibility of the Top-Project/WP Leader.
  - The PMB (Project Management Board) is responsible for maintaining and assuring the implementation of the openETCS Development Process and for ensuring the required “coordination" among the projects.
  - The Mentoring board is responsible for mentoring projects and advising.
  - The Project Office is responsible for the administrative tasks around the openETCS Development Process and maintains the openETCS Ecosystem project
- The tools to support the openETCS development process are open source tools. A relation of the tools approved by the consortium is in WP7.
- The engagement of the openETCS Advisory Group will not only provide valuable technical insights but visibility of the project within the railway community.

### 2.1.3 Compliance with Agile Development Requirements

Agile Project Management has been introduced to software projects in the 90-ties and is now a de facto industry standard well documented in publications. In the openETCS project the agile software development will be achieved with the SCRUM-Methodology. The document D2.3b [**?** ] describes the agile development in the openETCS project.

### 2.1.4 Compliance with software management and organisation according to EN50128:2011

In principle, two of the openETCS project results (Software and Tool Chain) are to be CENELEC safety integrity level (SIL) 4 certifiable. These are two of the results from WP3 and WP7. The following mechanisms, at organizational level, will help the corresponding project leader to provide evidence of compliance with chapters 5.1 and 5.2. However, evidence that requirements imposed are met will have to be provided for each of the two software projects on a project by project basis.

- Every partner in the consortium is ISO9001 Certified or will be in the position to provide evidence of a quality management process is accordance to ISO9001

- Every partner maintains an updated CV of the staff/experts involved in openETCS
- A Required Competence Matrix (RCM) per role and project will be maintained *(Chapter 4)*.
- A database with the participants per role and task/project will be maintained by the task/project leader.
- Overall, the independence required to develop certifiable results is promoted by the Work Programme which is structured into the following “independent" WorkPackages/Top-Projects, each lead by a different organization.
  - WP2, focused on Requirements Specification is led by SNCF.
  - WP3, focused on the Software Implementation taking as input WP2 and WP7 results is led by Alstom France.
  - WP4 focused on Verification and Validation (V & V) structure, is led by DLR
  - WP5 focused on demonstrating applicability/validity of WP3 and WP7 results is led by ERSA
  - WP7 focused on the development of the Tool Chain is led by Formal Mind taking as input WP2 and WP4 inputs
  - For the purpose of validating/adapting technical approaches, tools and concepts before they are taken into consideration, three Use Cases will be engaged.
  - The Open Development Process facilitates the creation of the necessary projects required to achieve the openETCS project results.
- For each assessable result, CENELEC required software roles will be covered by experts from different WPs. Incompatibilities can be controlled and monitored as active participation to the different projects has to be granted, accepted and is appropriately registered *(Chapter 2.2)*. Evidence of competence can be provided by comparing the CV of each expert with the RCM for the role assigned.
- For each assessable result, if possible, the role of the assessor will be selected from the external community of the project. Meanwhile, an internal independent assessor will be appointed. The role and profile of this assessor is detailed in openETCS/internal-assessment [17] [wiki pages]

One of the mechanisms to guarantee the availability of competence staff when needed will be the design and implementation of a training programme. The training programme will be managed by the Project Office. The identification of needs will be performed by the project leaders, the PMB and the Quality Manager. The training process is detailed in [governance]

## 2.2 Committers assignment and responsibilities

Each Top-Project/WP leader is responsible for establishing and publishing the specific required competence matrix for the Top-Project/WP *(Chapter 4)*. This matrix will be updated in response to the demands imposed by the evolution of the project. The competence matrix template [27]is provided in [governance]

Each Top-Project/WP leader is responsible for developing the most appropriate communities of users, adopters, contributors and committers as required by the Top-Project/WP. A database will be maintained and assessed periodically by the Top-Project/WP Leader. This database

will contain the coordinates of the expert, his/her role in the project and a basic explanation of adequacy. The expert database template is provided in governance.

The required core competences as well as the expected contribution of each of the identified communities are described in Chapter 4.

Only committers have write-access to the project resources. Becoming a committer requires of the acceptance of the project leader and of the rest of the project committers. Guidelines on how to become a committer can be found in [ecosystem wiki pages].

- It is the responsibility of the Project Leader to make sure the required competence to develop a task is covered by the engaged committers.
- It is the responsibility of the Open ETCS Project Leader to guarantee the required competence for the project is covered by the effective committers.

Contributors have read-access to the project resources, and acceptance is not required. Guidelines on how to become a contributor can be found in [ecosystem wiki pages].

An expert can contribute to different projects with different roles. The data from different project will be integrated and analysed to detect potential incompatibilities, if applicable. This activity will be done by the QA Manager. The guideline on how to select expert is detailed in governance.

### 2.3 Project QA Management

QA activities will be under the responsibility of the QA Manager, who reports to the Project Coordinator.

The QA Manager will be responsible for the identification, supervision and control of all the processes, methods and tools required to meet the quality targets of the project. It is also the responsibility of the QA manager to provide the necessary evidence that such activities have been developed.

The activities of the QA Manager will be:

- To maintain the QA Plan and associated procedures and guidelines.
- To maintain, implement and publish a QA Plan Backlog
- To participate in the openETCS Ecosystem project in cooperation with the Project Office
- To perform periodical audits of the maturity of the different on-going projects; propose improvement actions, if necessary.
- To participate in the review processes of the different work products.
- To collaborate with the Project Office in the identification of gaps and in the development of the corresponding Training Programme.
- To perform quantitative and qualitative analysis at process and product levels. To maintain a set of metrics for all the processes.
- To produce and publish the corresponding quality reports.

# 3 Life Cycle

The openETCs project itself is a R & D project running over 3 years which has the goal to deliver products such as the on-board specification model and the corresponding tool chain to generate source code based on this model. While the project life cycle is limited through the project time span, the products shall be used and also developed further after the end of the openETCS project. Respectively, the project only presents the development part of the product life cycle.

## 3.1 Project Life Cycle

The project Life Cycle is implemented through a set of WPs broken down into Tasks. In response of the nature of the project, these WPs are grouped into three purpose driven categories. The first category (WP2, WP4) addresses the specification of the work to be developed and validation of the results to be obtained; the second category (WP3, WP7, WP5) addresses the development itself and the demonstration of the software and the tools chain developed. The third category (WP1, WP6) addresses the project management, the quality assurance and the dissemination of the project. This structure enables both, the development and the integration of conceptual (R & D) and implementation activities to achieve innovative, validated and fit-for-purpose results. The detailed description of the Work Package and an overview plan is covered by the Full Project Proposal [FPP].

## 3.2 Product Life Cycle

As the openETCS project products shall be part of the train development the reference for their life cycles are the CENELEC standard phases defined in the EN50126. But as the products are in general R & D results, their life cycles do not include any certification or acceptance activities at the moment. The main openETCS products are the openETCS Software model and the openETCS tool chain development, which they have separate development life cycles. For both parts the main development, verification and validation activities are done during the openETCS project. For the software only the demonstrator implementation is part of the openETCS project, while any kind of implementation on a target hardware is out of the project. For the openETCS tool chain the basic implementation is part of the project, but all further steps from qualification on are out of the project. In general long time maintenance is a key concept of these products but it can not be established in the project time span.

### 3.2.1 Life Cycle of the openETCS Software

The software development life-cycle of the openETCS project should comply with the CENELEC EN 50128. Requirements imposed by the standard are analyzed and shown in detail in D2.2, while the software development life cycle and the agile software development process applied in this project is described in Deliverables D2.3a, D2.3b and D2.4.

Figure 2 gives an overview of the overall openETCS development Process.

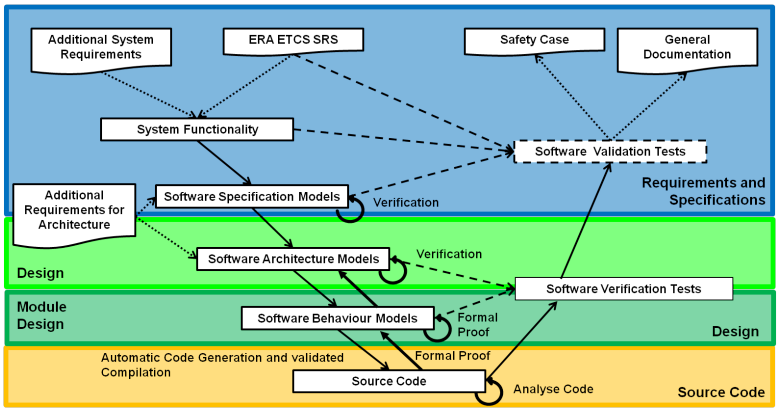



**Figure 2. Overall openETCS development Process**

In the following the development life cycle of the openETCS software is presented in tables. It is a reference to the D2.3a.

## Software Specification Phase

| Software Specification Phase | |
|---|---|
| | |
| Responsible | WP2 |
| Tasks | See D2_3a chapter 3.3.1 |
| Activities incl. QA | See D2_3a ch. 3.3.2 |
| Input | See D2_3a ch. 3.3.3 |
| Output | See D2_3a ch. 3.3.4 |

**Table 10. Software Specification Phase**

## Software Design Phase

| Software Design Phase | |
|---|---|
| | |
| Responsible | WP3, WP5, WP7 |
| Tasks | See D2_3a ch. 3.4.1 |
| Activities incl. QA | See D2_3a ch. 3.4.2 |
| Input | See D2_3a ch. 3.4.3 |
| Output | See D2_3a ch. 3.4.4 |

**Table 11. Software Requirements Specification Phase**

### Software Component Implementation Phase

| SW Component Implementation and Test Phase | |
|---|---|
| | |
| Responsible | WP3, WP5, WP7 |
| Tasks | See D2_3a chapter 3.5.1 |
| Activities incl. QA | See D2_3a ch. 3.5.2 |
| Input | See D2_3a ch. 3.5.3 |
| Output | See D2_3a ch. 3.5.4 |

**Table 12. Component Implementation and Test Phase**

### Software Integration Phase

| Software Integration Phase | |
|---|---|
| | |
| Responsible | WP4 |
| Tasks | See D2_3a chapter 3.6.1 |
| Activities incl. QA | See D2_3a ch. 3.6.2 |
| Input | See D2_3a ch. 3.6.3 |
| Output | See D2_3a ch. 3.6.4 |

**Table 13. Software Integration Phase**

### Software Validation Phase

| Software Validation Phase | |
|---|---|
| | |
| Responsible | WP4 |
| Tasks | See D2_3a chapter 3.7.1 |
| Activities incl. QA | See D2_3a ch. 3.7.2 |
| Input | See D2_3a ch. 3.7.3 |
| Output | See D2_3a ch. 3.7.4 |

**Table 14. Software Validation Phase**

## 3.2.2 Life Cycle of the openETCS Tools chain

The development of the Tool Chain has to comply with EN50128. Requirements imposed by the standard are analyzed and shown in detail in D2.2. The tools chain development life cycle is described in the document WP7-Tool ChainDevelpmentPlan. As the tool chain is a combination

and improvement of already existing tools, which have a specific life-cycle, the tools chain life cycle mainly consists of integration and maintenance activities.

The tool chain lifecycle is depicted in figure 3.

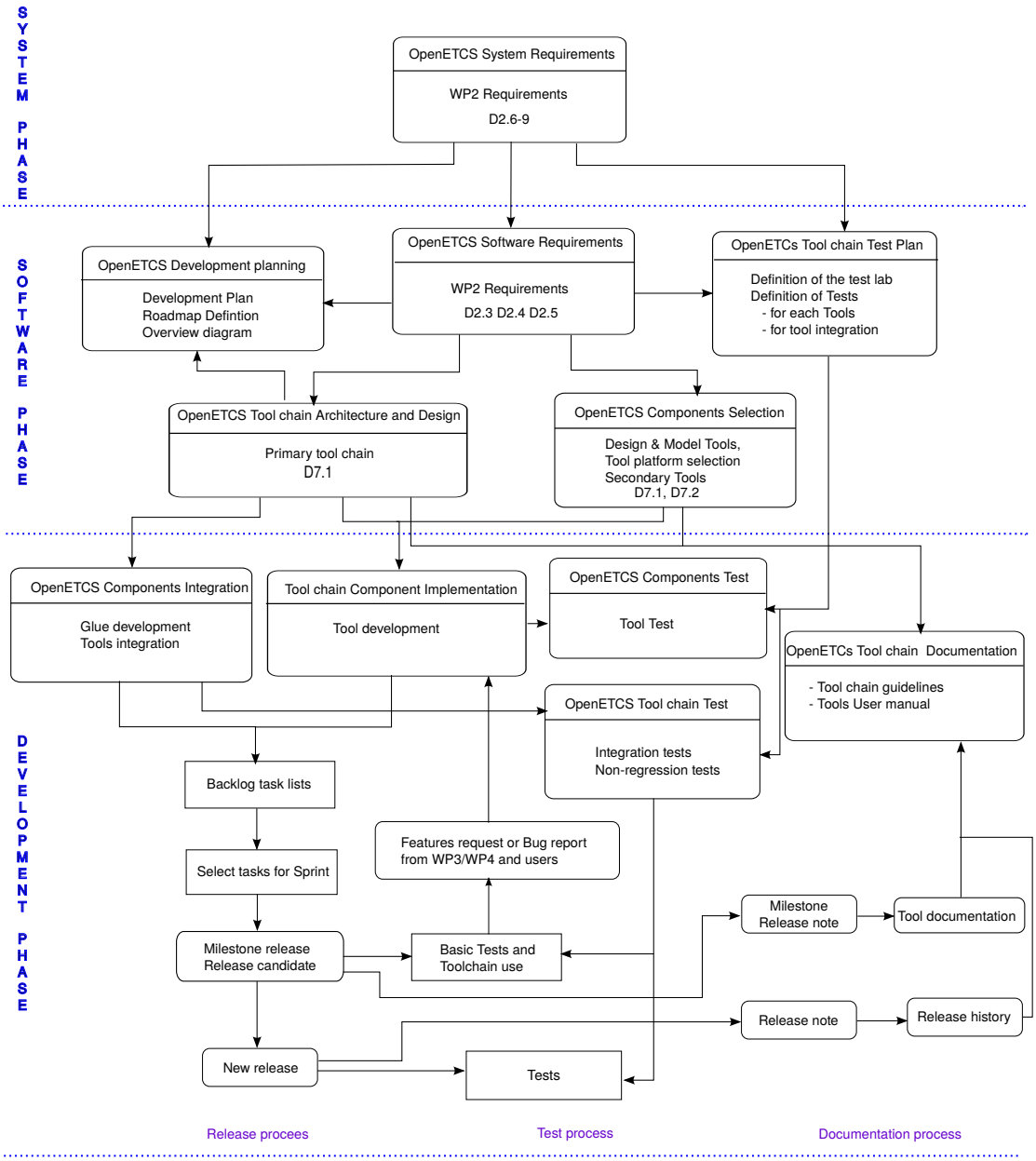



**Figure 3. The openETCS tool chain life cycle**

At following, a more detailed overview of the Tool Chain life cycle is provided. Each life cycle stage includes its input, output and the activities to be done.

### openETCS Tools and Tool Chain Software Requirements

| openETCS Tools and Tool Chain Software Requirements | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Specific requirements for the selection and the development of tools and the Tool Chain development. |
| Input | D2.6-9 |
| Output | D2.3, D2.4, D2.5 |

**Table 15. openETCS Tools and Tool Chain Software Requirements**

### openETCS Development Planning

| openETCS Development Planning | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Elaborate guidelines to develop the tool chain in an agile development process. |
| Input | D2.6-9 D2.3-5 |
| Output | Development Plan, Roadmap definition, Overview Diagram, Release process |

**Table 16. openETCS Tool Chain Development Planning**

### openETCS Tool Chain Test Plan

| openETCS Tool Chain Test Plan | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Elaborate test plans of the Tool Chain. |
| Input | D2.6-9 D2.3-5 |
| Output | Test Lab definition, Integration tests, Performance tests |

**Table 17. openETCS Tool Chain Test Plan**

### openETCS Tool Chain Architecture and Design

| openETCS Tool Chain Architecture and Design | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Selection of the tool chain design, Definition of how the tool chain is implemented. |
| Input | D2.3 D2.4 D2.5 |
| Output | D7.1 Report on all aspects of secondary tooling, Tool Chain Design Specification |

**Table 18. openETCS Tool Chain Architecture and Design**

### openETCS Tool Chain Components Selections

| openETCS Tool Chain Components Selections | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Evaluation and selection of tools according to the different development phases, Selection of the tool platform for the tool integration, Evaluation and selection of tools to answer features requests. |
| Input | D2.3, D2.4, D2.5, Feature requests |
| Output | Primary Tools: Evaluation of the models and tools against the WP2 requirements Primary Tool platform Evaluation: Evaluation of the tool platform against the WP2, requirements, D7.1 Report on the Final Choice of the Primary Tool Chain, Secondary Tools Evaluations:<br>O7-2-1 Management, O7-2-1 Safety, O7-2-1 VnV, O7-2-1 Transformation, D7.2 Report on all aspects, of secondary tooling |

**Table 19. openETCS Tool Chain Components Selections**

### openETCS Tool Chain Components Implementation

| openETCS Tool Chain Components Integration | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Implementation of the tools to be integrated into the tool chain |
| Input | D7.1 D7.2 |
| Output | Tools, Tools User Manual |

**Table 20. openETCS Tool Chain Components Implementation**

### openETCS Tool Chain Components Integration

| openETCS Tool Chain Components Integration | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Integrate tools, Develop glue to make tools talk to each other, Define how to use the tool chain. |
| Input | D7.1 D7.2 |
| Output | Tool chain Releases, D7.4, Release note, Tool chain Guideline |

**Table 21. openETCS Tool Chain Components Integration**

### openETCS Tool Chain Component Test

| openETCS Tool Chain Component Test | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Definition of how to test the Tool Chain, Execution of the Test Cases |
| Input | Test plan, Tools, D7.2 |
| Output | Tools Test report |

**Table 22. openETCS Tool Chain Component Test**

### openETCS Tool Chain Test

| openETCS Tool Chain Test | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Integration tests, Functional tests, Data artifacts integrity tests, Non-regression tests, Performance Test |
| Input | Test plan, Tools, D7.2 |
| Output | Tool chain Test report, Bug report, Feature request |

**Table 23. openETCS Tool Chain Test**

#### openETCS Tool Chain Maintenance

| openETCS Tool Chain Maintenance | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Report bug in tools or in the tool chain, Report missing or wishing feature for the tool chain |
| Input | Bug report, Feature request |
| Output | New Tool Selection, Bug fixes |

**Table 24. openETCS Tool Chain Maintenance**

#### openETCS Tool Chain Documentation

| openETCS Tool Chain Documentation | |
|---|---|
| | |
| Responsible | WP7 |
| Tasks | Document the development process, Document the use of the tool chain, Define the tool chain guidelines and best practice, Report missing or wishing feature for the tool chain |
| Input | Tool chain release, Release notes, Tools selection |
| Output | Tool chain guidelines, Tool chain users documentation, Tool chain developer documentation |

**Table 25. openETCS Tool Chain Documentation**

### 3.3 QA Management

The Software quality assurance will assure that any deviations during the project and product life-cycle from plans and standards are detected, recorded, evaluated, tracked, and resolved.

Software quality assurance will work with the Configuration Management process to assure that proper controls are in place and applied to life cycle data.

The monitoring activities of the Life Cycle control involve a development of criteria for inputs and outputs of the different life cycle phases, reviews and documentation audits to determine if the overall Life Cycle process is correctly followed. As well as assuring the compliance with the CENELEC standard.

The QA Manager will be responsible for:

- Conducting an internal formal audit every month to:
  - assure that the transition criteria to successfully assist in entry from one life-cycle phase to another, is quantifiable, flexible, well documented, and present for every life-cycle phase
  - assure that the different criteria for moving from one step to the next are satisfied (assure transition criteria are adhered to throughout the life-cycle).
  - assure each of the life-cycle phases outputs are verified, assured, and configured as part of the integral processes (sw verification process, configuration management process, sw quality assurance process, change/problem management process, review and revision processes, ...)
  - Verify the correct aplication of the integral processes inside life cycle
  - ensure that the output deliverables of a phase are consistent and meet all requirements established in the input deliverables
  - assure that the as-delivered products matches the as-built and as-verified product
  - Verify life cycle and assure that every objective and requirement of the CENELEC Standard is fulfilled
- Producing and publishing the corresponding quality reports

In order to do the formal review correctly it will be use the Life Cycle Control and Monitoring Activities formal audit checklist. This Checklist template is provided in [governance].

# 4 Roles

## 4.1 openETCS Roles

In view of the nature of the project, roles are grouped into three independent categories:

- CAT1: Open Source Development Process Roles
- CAT2: SCRUM Roles
- CAT3: CENELEC Roles

Therefore, any participant will always adopt a role within CAT1, a role within CAT2 and if he/she is involved in the development of a CENELEC assessable product, a third role in CAT3.

As already mentioned, openETCS is a project of projects. An expert can participate to different projects with different roles. Therefore an expert will have a CAT1, CAT2 and/or CAT3 role per project.

In the Appendices A, B, C and D, the responsibilities and the core competences required by each role are detailed. It is the responsibility of the QA Manager to keep them updated

In the case of CAT 1 roles, specific technical competence will be required depending on the scope of the project. For this reason a new column has been added. In this column, specific technical competences for each project and role are to be included. It is the responsibility of each project leader to provide this information.

According to the open development process followed by Open ETCS, the QA process is also a project. For this reason the QA Manager will have to meet the competences of a Project Leader and the specific competences imposed by CENELEC and the openETCS project to the Quality Manager activities. When needed, specific responsibilities imposed by a project to a role will be detailed too.

As project results affected by CENELEC are already identified, both core and specific required competence per CAT 3 role are included in Appendices C and D.

### 4.2 Roles within the Development process of the openETCS Software

The responsibilities and competences for every role specific to the openETCS Software development are listed in Appendix C. The independence of different roles is the core concept of the quality assurance strategy required be CENELEC standard. As openETCS is a collective project by various independent partners, the project organization already ensures full independence between the roles administrated by experts from different partners.

### 4.3 Roles within the Development process of the openETCS Tools Chain

See Appendix D

### 4.4 QA Activities

The QA Manager will be in charge of:

- Maintaining the Requirements Competence Matrices updated in response to the evolution of the openETCS project
- Performing periodical audits of the participants' database per project; trace database with the RCM (Required Competence Matrix) for such project
- Identify training needs and provide the required support to the Project Office in the definition and organization of the corresponding training activities.
- In the case of CENELEC related project, provide the necessary evidence of competence and independence between roles. If this is not possible, propose the necessary solutions and support the projects in its implementation

# 5 Methods, measures and tools for quality assurance (product + open ETCS software + Tools chain)

Selection of methods and tools used in each phase of the openETCS process is a part of the WP7 activities. This selection is based on the state of art established by WP2 (D2.1 and D2.2), the set of requirements defined by WP2 (D2.6-9) and the process definition (D2.3, D2.4, D4.1, D4.2.3).

Results of the selection of methods and tools are given in the D7.1 and D7.2 deliverables. Conformance of the methods and tools are going to be discussed in D7.3.

The following table give details of all this deliverables.

## 5.1 Methods, measures and tools for quality assurance openETCS Application Software

It is assumed that the openETCS application software will be SIL4 compliant. Therefore, the methods, techniques and tools shall be suitable to SIL 4.

The selected methods and measures are included in Appendix E

## 5.2 Methods, measures and tools for quality assurance openETCS Tools chain

The Tool Chain will be composed of a set of tools with different levels of interaction. The openETCS tool chain consists on a series of supporting tools that helps in the development of the whole project. Such tools include, but are not limited to, development and design tools, language translators, testing and debugging tools, and configuration management tools. Support tools are further classified according to their influence on the system:

- On-line support tools are tools that can directly influence the safety-related system during its run time
- Off-line support tools are tools that support a phase of the software development lifecycle and that cannot directly influence the safety-related system during its run-time

Following CENELEC criteria, each tool belongs to one of the following classes: T1, T2 and T3. Class 3 and Class 2 Tools are obliged to follow specific development methods, techniques and tools.

### 5.2.1 Selected Tools

Nowdays, by looking different aspects of the possible tools for the toochain like the strengths, weaknesses, opportunities, features and threats, the selection of the below tools is done.

- Eclipse (eclipse Kepler SR1 IDE): tool platform (Eclipse with the modeling framework (EMF))
- Papyrus/SysML to cover the highest level of the openETCS V cycle
- ProR: requirement manager
- SCADE: Low level description and code generation

- EFS to support V&V activities.
- Git: configuration management

Decision has been made to start WP3 and WP4 activities based on SysML/Scade basis, which are involved in the Tool Chain. However, work to define an alternative fully open source chain to provide SIL4 software is encouraged. On going work is thus provided around SysML/ Classical B.

### 5.2.2 Metrics Covered by tools

| T1 Tools | |
|---|---|
| Name | Metrics |
| ProR | requirements management |
| Git | Software Configuration Management |

**Table 26. T1 Tools**

| T2 Tools | |
|---|---|
| Name | Metrics |
| | |

**Table 27. T2 Tools**

| T3 Tools | |
|---|---|
| Name | Metrics |
| Papyrus | System level description |
| SCADE | Software level description |

**Table 28. T3 Tools**

## 5.3 Quality Control and Monitoring Activities

The monitoring activities of the selected Tools, Methods and Techniques implicates a development of criteria, reviews and audits to determine if the overall tools and methods & Techniques selection and implementation is correctly develop and maintain. As well as assuring the compliance with the CENELEC standard.

The Quality Assurance Manager should do the following monitoring activity:

- Conduct an internal formal audit to confirm the methods and tools are ready to use. This audit consists in:
  - Assessing the accept criteria of the tools and methods
  - Assessing the fulfilment of the expectations of the tools and methods
  - Assesing the set of selected methods and tools fulfill CENELEC standard (Benchmarking methods, techniques and tools against CENELEC standard.)

  - Verifying every tool availability and operability
  - Verifying new tool version control
  - Verifying the evaluation of the selected tools, methods and techniques
- Conduct an internal formal audit every month to confirm the methods and tools chain are appropiately implemented.
  - Verifying the correct use of each tool in each WP and Phase
  - Verifying the correct use of each technic and metric in the project

In order to do the formal review correctly it will be use the Tool, Method and Technic Monitoring Activities formal audit checklist. This Checklist template is provided in [governance].

# 6 Documentation

The documentation structure of the openETCS project is composed of:

- Deliverables, which constitute the official outcomes of the different Top-Projects/WPs
  - The relation and scope of the deliverables to be produced along openETCS can be found in the FPP [25].
  - The updated status of development of each Deliverable can be found in [State-of-Deliverables Wiki].
  - The approved and therefore valid version of each Deliverables can be found in the repository of the Top-Project/WP it belongs to.
- Contractual documents, with the Commission and among the project partners
  - The status of development of each contractual document can be found under the repository of Management (WP1).
  - The last approved and therefore valid version of each contractual document can be found under the repository of Management (WP1).
- Periodic Progress Reports, to show progress to ITEA and EC representatives.
  - The state of each Periodic Report can be found under repository of Management (WP1).
  - The last approved and therefore valid version of each Periodic Progress Report can be found under the repository of Management (WP1).
- Supporting Documents, in the form of Templates and Procedures
  - The procedures and templates applicable to a specific Top-Project/WP can be found in the repository of the corresponding TP/WP.
  - The procedures and templates applicable to the whole project can be found in the repository of Governance.
- Internal Reports, in the form of Meeting Minutes
  - The minutes of the weekly scrum meetings are found in the repository of Governance.

The nomenclature used for the naming of the different documents is provided in [governance Wiki].

For each TP/WP the relation of existing documents is provided in the form of a list [Wiki]. This list includes a direct access to the valid version of each document.

### 6.1 Documentation Structure within the development process of the openETCS Software

As a SIL4 software, the documentation structure has to comply with CENELEC requirements. The following table shows the document structure required by CENELEC for a SIL 4 development and the corresponding documents produced in the openETCS project.

**Table 29. Documentation Structure**

| Documentation Structure within the development process of the openETCS Software | | | |
|---|---|---|---|
| **Phase** | **SIL4** | **Document** | **Responsible** |
| Planning | Highly Recommended (HR) | Software Quality Assurance Plan (D1.3.1)<br>Software Quality Assurance Verification Report<br>Software Configuration Management Plan (SCMP)<br>Software Verification and Validation Plan (D4.1)<br>Hazard and Risk Analysis Methodology (D4.2.3)<br>Dissemination Plan (D6.1) | Izaskun de la Torre<br>Todo<br>Peer Jacobsen<br>Hardi Hungar<br>Jan Welte<br>Stefan Rieger |
| Software Requirements | HR | Software Requirements Specification (D2.6_9 Requirements for openETCS)<br>ETCS_OBU_FunctionalStructure -Uwe Steinke-<br>Software Requirements Test Specification<br>Software Requirements Verification Report | Sylvain Baro and Jan Welte<br>todo<br>todo |
| Software Architecture Modeling | HR | System Specification Model (D3.5)<br>Functional Model (D3.6)<br>System Architecture Model with physical allocation (D3.7)-Alstom-<br>Data Dictionary -Bernd Hekele-<br>Data Dictionary Design Specification - Cecile Braunstein-<br>Internal Data Structure -Jan Welvaarts-<br>Data Dictionary Data Catalog -Bernd Hekele-<br>Data Model Description -Marielle Petit Doche-<br>openETCS API Document -Alstom-(D2.7)<br>openETCS SSRS Interface Document<br>EVC External Interfaces-Baseliyos Jacob-<br>SRS analysis and functions -Bernd Hekele-<br>Software Architecture Specification<br>Software Design Specification<br>Software Interface Specification<br>Software Integration Test Specification<br>Software Architecture and design verification report | |

Continued on next page

**Table 29 – continued from previous page**

| Documentation Structure within the development process of the openETCS Software | | | |
|---|---|---|---|
| **Phase** | **SIL4** | **Document** | **Responsible** |
| Component Design | HR | Software Component design specification<br>Software Component Test Specification<br>Software Component design verification report | |
| Component Implementation and Testing | HR | 1st V&V report on model (D4.2.1)<br>1st V&V report on implementation/ code (D4.2.2)<br>Software source code and supporting documentation<br>Software source code verification report<br>Software Component Test Report | Ana Cavalli, João Santos, Huu-Nghia Nguyen, Stefan Rieger, Cécile Braunstein, Uwe Steinke, Benoît Lucet, Matthias Güdemann, Brice Gombault, Marielle Petit-Doche, Alexander Nitsch and Benjamin Beichler<br>Marc Behrens, Jens Gerlach, Kim Völlinger, Andreas Carben and Izaskun de la Torre |
| Integration | HR | Software Integration Test Report<br>Software Integration Verification Report | |
| Overall Software Testing/Final validation | HR | Overall Software Test Report<br>Software Validation Report<br>Tools Validation Report<br>Release Note | |
| Systems configured by Application Data/ algorithms | HR | Application Requirements Specification<br>Application Preparation Plan<br>Application Test Specification<br>Application Architecture and Design<br>Application Preparation Verification Report<br>Application Test Report<br>Source Code of Application Data/Algorithms<br>Application Data/Algorithms Verification Report | |
| Software Deployment | HR | Software Release and Deployment Plan<br>Software Deployment Manual<br>Release Notes<br>Deployment Records<br>Deployment Verification Report | Bernd Hekele |

Continued on next page

**Table 29 – continued from previous page**

| Documentation Structure within the development process of the openETCS Software | | | |
|---|---|---|---|
| **Phase** | **SIL4** | **Document** | **Responsible** |
| Software Maintenance | HR | Software Maintenance Plan<br>Software Change Records<br>Software Maintenance Records<br>Software Maintenance Verification Report | todo |
| Software Assessment | HR | Independent Assessment Plan<br>Software Assessment Report | Cyril Cornu |

## 6.2 Documentation Structure within the development process of the openETCS Tools chain

CENELEC Standard requires that all the off-line support tools have a manual as minimun required documentation. The Standard also established that tools in classes T2 and T3 should have documentation that defines the behaviour of the tools together with instructions and constrains on its use. This documentation should be a justification for use, a potencial failures identification and the ways to avoid or mitigate them, manuals and use restrictions. It is also required that tools to be assessed with the aim to determine the level of reliance that shall be placed on the tool, and potential failure mechanisms that may affect the executable software (T3 only).

For T3 class applications, evidence shall be available that the tool conforms to its specification or manual, so called tool assessment. Such tool assessment shall cover:

- A chronological record of the validation activities
- The version of the tool product manual being used
- The tool functions being validated
- Tools and equipment used
- The results of the validation activity; the documented results of validation shall state either that the software has passed the validation or the reasons for its failure
- Test cases and their results for subsequent analysis
- Discrepancies between expected and actual results

It is important to add, that every new version of a support tool shall be certified. This may rely on evidence provided for earlier versions if sufficient evidence provides that the modifications do not affect tool compatibility with the rest of the tools in the integrated tool chain and that the new version is unlikely to contain significant new, unknown faults.

**Table 30. Documentation Structure**

| Documentation Structure within the development process of the openETCS Software | | | |
|---|---|---|---|
| **Phase** | **SIL4** | **Document** | **Responsible** |
| Planning | HR | Software Quality Assurance Plan (D1.3.1)<br>Software Quality Assurance Verification Report<br>Software Configuration Management Plan (SCMP)<br>Software Verification and Validation Plan (D4.1)<br>Hazard and Risk Analysis Methodology (D4.2.3)<br>Dissemination Plan (D6.1)<br>Tool chain Qualification Process Description | Izaskun de la Torre Todo<br>Peer Jacobsen<br>Hardi Hungar<br>Jan Welte<br>Stefan Rieger<br>Cecile Braunstein, Jan Peleska and Stefan Rieger |
| openETCS Software Requirements | HR | Software Requirements Specification (D2.6_9 Requirements for openETCS)<br>D2.3 Process Definiition<br>D2.4 Methods Definition<br>D2.5 Description of ETCS using those methodologies<br>Requirements Modeling for SSRS Activities with ProR | Sylvain Baro and Jan Welte<br>Marielle Petit-Doche and Matthias Güdemann<br>Marielle Petit-Doche, David Mentre and Matthias Güdemann<br>David Mentre, Stanislas Pinte, Guillaume Pottier<br>Michael Jastram |
| openETCS Development Planning | HR | Tool Chain Development Plan<br>Roadmap definition<br>Release Process | Cecile Braunstein and Jan Peleska<br>Cecile Braunstein<br>ToDo |
| openETCS Tool chain Test Plan | HR | Tool Chain Test Plans<br>Test Lab definition | ToDo |
| openETCS tool chain Architecture and design | HR | Tool chain Design Specification | Cecile Braunstein and Jan Peleska |
| openETCS Components Selections | HR | D7.1 Report on the Final Choice of the Primary Tool Chain<br>Evaluation of the models and tools against the WP2 requirements<br>Evaluation of the tool platform against the WP2 requirements<br>D7.2 Report on all aspects of secondary tooling<br>Secondary Tools Evaluations (O7-2-1 Management, Safety, V&V, Transformation) | Michael Jastram<br>Marielle Petit-Doche<br>Cecile Braunstein<br>Marielle Petit-Doche<br>Marielle Petit-Doche |

Continued on next page

**Table 30 – continued from previous page**

| Documentation Structure within the development process of the openETCS Software | | | |
|---|---|---|---|
| **Phase** | **SIL4** | **Document** | **Responsible** |
| Tool chain Components Implementation | HR | Tools and supporting documentation (User Manual, developer documentation) | |
| openETCS Component Tests | HR | Tools Test Reports | |
| Tool chain Components Integration | HR | Tool chain Releases<br>Tool chain Release note<br>Tool chain Guideline<br>Tool chain Integration Test Report | |
| Overall openETCS Tool chain Test Testing/Final validation | HR | Overall Tool chain Test report<br>Tool chain Validation Report<br>Tools Validation Report<br>Release Note | |
| Tool chain Maintenance | HR | Tool chain Maintenance Plan<br>Tool chain Change Records<br>Tool chain Maintenance Records<br>Tool chain Maintenance Verification Report | |

## 6.3 Quality Control and Monitoring Activities

The monitoring activities of the Documentation Structure control involves a development of criteria, reviews and documentation audits to determine if the overall structure of the documentation is correctly followed. As well as assuring the compliance with CENELEC standard for a SIL 4 product.

The Quality Assurance Manager should do the following monitoring activity:

- Conduct an internal formal audit every month to confirm the documentation structure is maintained correctly. This audit consists in:
  - Verifying each document development in time and its correct classification inside the WP and Phase.
  - Controlling the document version labels and identifier
  - Benchmarking against CENELEC standard (verifying life cycle and assuring that every objective and requirement of the CENELEC Standard is fulfilled)
  - Assessing the document timeline's creation
- Produce and publish the corresponding quality reports

In order to do the formal review correctly it will be use the Document Structure Control and Monitoring Activities formal audit checklist. This Document Structure Checklist template is provided in [governance].

# 7 Documentation Control

The Documentation Control procedure describes the steps to follow to ensure that the documentation developed whiting the openETCS project is current and suitable for use by the Eclipse community, the project members and the key customers. The main control activities covered by the procedure include the document creation and review, the approval, dissemination, archiving, modification and update due to a change request or the monitoring of the evolution among the time among others.

The implementation of this procedure, shall ensure that openETCS documents can be located easily, be periodically reviewed, have the nomenclature updated when needed, be available at any time, and be moved and archived when they are labelled ad obsolete.

The whole procedure is fully described in the [Document Control Procedure] .

There is a signature copy inside the project office mandatory for all official deliverables and selected additional documents. The selection of the documents to be signed signed is in the resposibility of the workpackage leader

## 7.1 Quality Control and Monitoring Activities

The monitoring activities of the Documentation Control, carried on by the Quality Assurance Manager, implicate an in-depth analysis of the document development process. This analysis involves developing criteria, conducting reviews and examining documentation to determine how the process is conducted.

The Quality Assurance Manager should do the following monitoring activity:

- Conduct an internal formal review every 2 months to confirm the documentation control is done correctly. This review consists in:
  - Verifying each document correct location and labeling in the GITHUB repository
  - Verifying the roles of the documentation.
  - Verifying that the terminology, acronyms or abbreviations have the same meaning in every document
  - Verifying the document schedule fulfillment
  - Controlling obsolete documentation labels and location
  - Benchmarking compliance with CENELEC standard
  - Assessing that every document applies the conditions and requirements of the preceding document with which it has a hierarchical relationship. It should not be contradictions among the documentation and its preceding documents
  - Assessing that there is a reference with the same name and description for each element or concept in every document
- Conduct an internal informal review every month using the plans, goals and objectives established in the project to verify control documentation and documentation development is done satisfactorily.

- Participate in the review processes of the different documents
- Maintain a set of metrics for the Document Control process.
- Produce and publish the corresponding quality reports.

In order to do the formal review correctly it will be use the Document Control and Monitoring Activities formal review checklist. This Document Control Checklist template is provided in [governance].

# 8 Tracking and tracing of deviation

## 8.1 Traceability (openETCS software + Tools chain)

The monitoring activities of the Traceability help to determine how the traceability among the different elements of the project is conducted. In order to have a good traceability, it has been established to develop a traceability matrix.

During the whole project different traceability matrix will be used and all of them will monitor in the same way.

The Quality Assurance Manager should do the following monitoring activity:

- Verify that the means to demonstrate traceability throughout all phases of the lifecycle are provided
- Verify that the output of the traceability process is the subject of formal configuration management
- Verify that the requirements traceability is covered completely
- Assure that any untraceable material (requirement, model, code, ...) to have no bearing upon the safety or integrity of the system
- Ensure that each specific CENELEC role is responsible for establishing and maintaining traceability to and from the specific elements.
- Monitor the different matrix with informal reviews every month and with formal reviews every 3 months. These reviews will consist in assuring that the generated matrix table has well traced every element of the project.

  In order to do this the following relations among elements will be reviewed:

  1. Traceability between requirements and models (design)
  2. Traceability between models and the generated code
  3. Traceability among requirements, models, test plans, specifications to the test or other reports which record the results of their application and tool chain.

In order to do the formal review correctly it will be use the Traceability Activities formal review checklist. This Traceability Checklist template is provided in [governance].

## 8.2 Configuration Management

Configuration Management (CM) is used to handle changes systematically so that a system maintains its integrity over time. The Software Configuration Management Plan [SCMP] [26] defines the procedures, techniques, and tools that are required to manage the software development, evaluate proposed changes, trace the status of changes, and to support an inventory of the system.

The main points to perform the configuration management process are:

- Configuration Management Tools
- Configuration Items
- Configuration Management Organization
- Configuration Control/Change Management
- Configuration Audits
- Baselines

The Quality Assurance Manager is accountable for the implementation of the System Configuration Management Plan (SCMP).

The QA Manager will be in charge of:

- perform periodical audits
  - Audits to verify the process itself: the correct implementation of the process and the compliance of the process with CENELEC Standard
- Produce and publish the corresponding quality reports.

## 8.3 Fault Management

A failure is a deviation of the component or system from its expected delivery, service or result. A failure is the consequence of a fault or error in a system but not all faults result in failures.

Faults, failures and errors encountered during the review activities (QA. Verification, Validation, Assessment) planned in the software development life-cycle, problems reported by users and customers as well as change requests initiated by any of the system stakeholders will be reported and managed following the Change/Problem Management Process [5] detailed in [governance] and through the Change/Problem Management Tool. This tool will be integrated with the Configuration management tool *GIT* and will be configured to implement and record all the information generated during the process.

The integration with the Configuration management tool *GIT* will permit:

- Traceability between Change/Problem Requests and the configuration items where the problem was located.

- Traceability between the configuration items modified and the corresponding Change/problem request.

The implementation of the workflow will permit:

- A complete history trail of the Change Request/Problem Report

The purpose of the Change/Problems Management implementation at openETCS project is to ensure that standardized methods and procedures are used for efficient and prompt handling of all changes/problems associated with the openETCS products, in order to minimize the number and impact of any related changes/problems. Changes/problems in the products may arise reactively in response to incidents, or proactively from seeking improved efficiency and effectiveness, as well as to enable or reflect openETCS initiatives, or products improvements.

The QA Manager will be in charge of:

- perform periodical audits and quality assessments of the bugs received
  - Audits to verify the process itself
  - Quality Assessments to verify the evolution of the product quality
- Assist in determining QA impacts
- Support Problem owner in analysis

### 8.4 Grievance Handling

It is a good culture to solve concerns as close as possible to the root cause of an problem or a misunderstanding. This means, the team where a problem is seen first is empowered to search for a solution of the problem first.

If the partners cannot agreed on a solution, the impediment is escalated to the next level in the project hierarchy.

When a member of the openETCS community has a concern about a Project, the member will raise that concern with the Project's Leadership (e.g., task leader in openETCS). If the member is not satisfied with the result, the member can raise the concern with the parent Project's Leadership, typically the workpackage leader.

The Member can continue appeals up the Project Leadership Chain and, if still not satisfied, thence to the project management board PMB, then the openETCS project lead, and finally to the project co-operation committee (PCC). All appeals and discussions will abide by the Guiding Principles of being open, transparent, and public.

Member concerns may include:

- Out of Scope. It is alleged that a Project is exceeding its approved scope.
- Dysfunctional. It is alleged that a Project is not functioning correctly or is in violation of one or more requirements of the Development Process.

- Contributor Appeal. It is alleged that a Contributor who desires to be a Committer is not being treated fairly.
- Invalid Veto. It is alleged that a -1 vote on a Review is not in the interests of the Project and/or of Eclipse.

A variety of grievance resolutions are available to the PMB up to, and including, rebooting or restarting a project with new Committers and leadership.

The issues seen during a sprint shall be taken to the sprint retrospective in order to help the team find an easy way in the future.

## 8.5 Software Maintenance

### 8.5.1 Software Maintenance Plan

Software Maintenance Plan introduces the approach that the openETCS-Software and openETCS-Tool Chain project adopts for the maintenance of the software components.

The procedures for software maintenance will be contained in the Software Maintenance Plan for openETCS-Software (SMP-SW) and the Software Maintenance Plan for openETCS-Tool Chain (SMP-Tool Chain). These procedures should also contain information about:

- Control of the error report, the error log, maintenance records, authorizations to make changes and software configuration / system and the techniques for estimation impact analysis and record and data analysis.
- Evaluation, Verification and Validation of every change
- Definition of software modification process (definition of the Authority which approves the changed software, etc...)

### 8.5.2 Modification and change control

A change is the addition, modification, or removal of a configuration item (CI), product, or product component, and/or its associated elements

The change requests initiated by any of the system stakeholders will be reported and managed following the Change/Problem Management Process [5] detailed in [governance] and through the Change/Problem Management Tool.

The Change/problem Management process aims to evaluate and plan the change/problem process to ensure that, if a change is made, it is done in the most efficient way possible, following the established procedures and ensuring the quality and continuity of the openETCS project and products at all times.

The Change/problem Management process should define at least the following:

- the necessary documentation to report a problem
- analysis of the collected information

- practices to be followed to report, track and resolve identified problems
- responsabilities
- controls to ensure that corrective actions have been taken and are effective
- impact analysis
- approval before implementation

### 8.5.3 Quality control and monitoring activities

For Software Maintenance the following metrics to follow in order to control the Maintenance phase are identified:

- Maintenance efficiency
- Maintenance effectiveness

Any trends and changes that occur provide an analytical basis for managerial decision making, regarding issues such as; examining resource requirements and initiation of corrective and preventive actions.

The QA Manager will be in charge of:

- perform periodical audits and quality assessments of the change request received
  - Audits to verify the processes themself: the correct implementation of the processes and the compliance of the processes with CENELEC Standard
  - Quality Assessments to verify the evolution of the product quality
- Assist in determining QA impacts
- Support Change owner in analysis

## 9 Supplier Control

This section describes what openETCS consortium expects its suppliers to do to ensure that all openETCS products' requirements and expectations are met.

This Supplier control applies to all Suppliers providing openETCS project with materials, products, processing, and related services.

At following, the expected Suppliers' general requirements are listed:

- Supplier shall ensure the confidentiality of openETCS project and products under development, and related product information, as well as intellectual property shared as a result of the working relationship.
- Suppliers are expected to have an effective quality system that ensures conforming product is delivered.

  - Suppliers shall maintain a Quality Management System suitable to the products and services provided to openETCS, that is certified by an accredited third-party certification body, i.e. ISO9001. This letter of accreditation should be provided to the respective QA personnel
  - In the absence of third-party certification, depending on the product, its application, value, and criticality, the openETCS community and Quality Assurance Manager may authorize the acceptance of other evidence of compliance

- Supplier should assure that all performance, endurance, maintenance, safety and warning requirements are met.
- The Supplier shall maintain documented procedures for identification of product from receipt and during processes of production and delivery. When traceability is a specified requirement, the Supplier shall establish and maintain a documented procedure for unique identification of individual product or batches
- The supplier shall provide and maintain suitable gauges, measuring instruments and test equipment to measure/test all material for conformance to openETCS requirements.
- Copies of quality conformance inspection data pertinent to material inspection must be provided by the supplier if required for each shipment or retained at Suppliers premises for future verification.
- The Supplier shall provide evidence that the following verifications required by the design record and control plan have been completed and that results indicate compliance with specified requirements
- Suppliers will be responsible for corrective action when changes to product specifications without prior notification to QA result in non- conformity to product or processes.
- openETCS requires all Suppliers to be approved prior to the issuance of contracts

### Supplier Approval Process

- **Registration:** New suppliers must complete a registration form. This form initiates the approval
- **Evaluation:**
  - Ensure Supplier Risk Assessment considers both:
    * Quality risk
      · Finished Device Quality implications
    * Supply risk
      · Including implications of supplier going out of business
  - Evaluate Suppliers using Questionnaires, Self surveys and Audits techniques
  - Supplier's grading based upon evaluation results and assigned an evaluation status: Approved, conditional or not approved
- **Certification**: Classify Suppliers based on both QUALITY Risk and SUPPLY Risk.

The QA Manager will be in charge of:

- inspect records/evidence of a supplier's quality management systems at their facility
- monitoring and feedback processes: Include periodic review of critical product/process data
- documents problem issues and requirements for the supplier
- analyses the supplier operating conditions,
- Establish Minimal performance for Quality and Delivery
- When a supplier provides a product/part, apply supplemental controls to further mitigate risk
  - Product Acceptance Activities
  - Supplier Performance and Monitoring: augmented frequency of reviews
- creates a corrective development profile together with the supplier,

# 10 Publishing Guideline

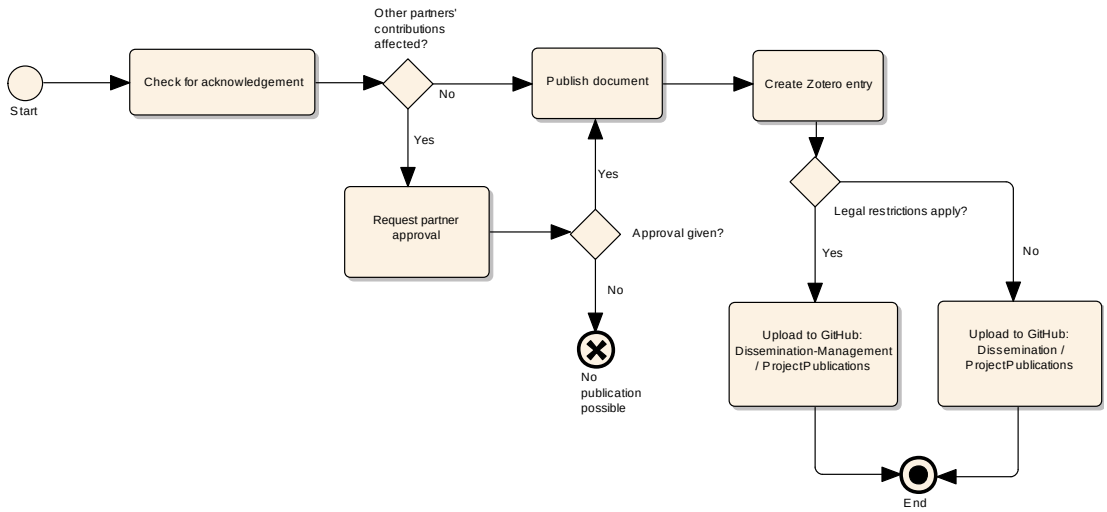



**Figure 4. The publishing process as BPMN diagram**

When publishing in the context of the openETCS project authors shall adhere to this guideline. Figure 4 depicts the steps as graphical BPMN process. The individual steps are described in detail in the following.

1. It must be ensured that the *project*, the *funding authority* and the *grant number* is mentioned in the paper/presentation. The following acknowledgements can be used:

   **Germany** This work was funded by the German Federal Ministry of Education and Research (Grant No. 01IS12021) in the context of the ITEA2 project openETCS.

   **Belgium/Brussels region** This work was funded by the Région de Bruxelles-Capitale / Brussels Hoofdstedelijk Gewest (Grant No. RBC/12 R 11) in the context of the ITEA2 project openETCS.

**Belgium/Walloon region** This work was funded by the Walloon Region (DG06) (Grant No. 6921) in the context of the ITEA2 project openETCS.

**France** This work was funded by the "Direction Générale de la compétitivité, de l'industrie et des services" (DGCIS) (Grant No. 112930309) in the context of the ITEA2 project openETCS.

**Spain** This work was funded by the "Gobierno de España, Ministerio de Ciencia e Innovación" in the context of the ITEA2 project openETCS

2. Publications potentially affecting project contributions of other partners require explicit approval. A request for approval shall be accompanied with a reasonable deadline (e.g., two weeks). Please consider a joint publication with the involved partners.

3. An entry with the details of the publication should be added to the Zotero group *openETCS Publications* by using the Zotero tool or the website zotero.org. A how-to regarding the use of Zotero in openETCS is provided here. A link to an official and public webpage where the publication can be obtained/purchased should be included.

4. The final document should be uploaded to GitHub to one of the following directories:

   - To Dissemination-Management/ProjectPublications if legal restrictions apply for publication.
   - To Dissemination/ProjectPublications if it can be published freely under the openETCS Open License.

# 11 Perimeter of the System

## 11.1 List of Functions

**Table 31. Functions**

| Functions | | | |
|---|---|---|---|
| Number | Name | Block/ Function | Complexity |
| 1 | DataPreparation | B | |
| 1.1 | Board_External_Interface | F | |
| 1.2 | GATC TRAINBORNE SUB SYSTEM | B | |
| 1.2.1 | Filter_information_from_ERTMS_trackside (including linking) | F | 3 |
| 1.3 | Provide_automatic_train_protection | B | |
| 1.3.1 | Manage_STMs | F | 3 |
| 1.3.2 | Determine_train_location_information | F | 3 |
| 1.3.3 | Control_route_suitability | F | 1 |
| 1.3.4 | Manage_track_conditions | F | 2 |
| 2 | Ensure_train_protection | B | |
| 2.1 | Manage_reception_of_MA_information | F | 2 |
| 2.2 | Manage_TSR | F | 1 |

Continued on next page

**Table 31 – continued from previous page**

| Functions | | | |
|---|---|---|---|
| Number | Name | Block/ Function | Complexity |
| 2.3 | Manage_Speed_Supervision_Inputs | F | 2 |
| 2.4 | Active_and_Manage_train_protection | B | |
| 2.4.1 | Activate_train_protection_in_FS | F | 3 |
| 2.4.2 | Activate_train_protection_in_OS | F | 3 |
| 2.4.3 | Activate_train_protection_in_LS | F | 3 |
| 2.4.4 | Activate_train_protection_in_SR | F | 2 |
| 2.4.5 | Activate_train_protection_in_UN | F | 2 |
| 2.4.6 | Activate_train_protection_in_SH | F | 2 |
| 2.4.7 | Activate_train_protection_in_TR | F | 2 |
| 2.4.8 | Activate_train_protection_in_SF | F | 1 |
| 2.4.9 | Activate_train_protection_in_SB | F | 2 |
| 2.4.10 | Activate_train_protection_in_PT | F | 2 |
| 2.4.11 | Activate_train_protection_in_RV | F | 2 |
| 2.4.12 | Activate_train_protection_in_IS | F | 1 |
| 2.4.13 | Perform train protection | F | 3 |
| 2.4.14 | Perform train protection related actions | F | 3 |
| 2.4.15 | Activate_train_protection_in_SN | F | 2 |
| 2.4.16 | Activate_train_protection_in_PS | F | 2 |
| 2.4.17 | Activate_train_protection_in_NP | F | 1 |
| 2.5 | Manage_emergency_stop_messages | F | 2 |
| 3 | Manage_mode_and_level_and_procedures_and_ancillary_fu | B | |
| 3.1 | compute_mode | F | 2 |
| 3.2 | compute_level | F | 2 |
| 3.3 | Manage_procedures | B | |
| 3.3.1 | capture_data_for_mission | F | 2 |
| 3.3.2 | handle_mode_profile_procedure | F | |
| 3.3.3 | handle_SH_procedure | F | |
| 3.3.4 | handle_RV_procedure | F | |
| 3.3.5 | handle_override_EOA | F | |
| 3.4 | Perform_ancillary_functions | B | |
| 3.4.1 | Manage_track_ahead_free | F | 2 |
| 3.4.2 | Display_text_messages_coming_from_trackside | F | 1 |
| 3.4.3 | Display_geographical_position | F | 1 |

Continued on next page

**Table 31 – continued from previous page**

| Functions | | | |
|---|---|---|---|
| Number | Name | Block/ Function | Complexity |
| 3.4.4 | End_mission | F | 1 |
| 3.4.5 | Manage_national_values | F | 1 |
| 3.4.6 | store_configuration_data | F | 1 |
| 3.4.7 | Management of MA request | F | 2 |
| 3.4.8 | Sending of position report | F | 1 |
| 3.4.9 | Determine train integrity | F | 1 |
| 3.4.10 | Detect_change_of_orientation | F | 2 |
| 3.4.11 | Manage_cold_movement_detection | F | 1 |
| 3.4.12 | Provide_train_movement information | F | |
| 3.4.13 | Manage_radio_sessions | F | |
| 3.4.14 | Record_juridical_data | F | 2 |
| 3.4.15 | Interface with train | F | 2 |
| 3.4.16 | Interface with DMI | F | 2 |

## 12 List of Risks

**Table 32. Risks**

| Risks | |
|---|---|
| Event Id. | Event Description |
| MMI-1a | False acknowledgement of mode change to less restrictive mode |
| MMI-1b | False command to enter Non-leading mode |
| MMI-1c | False command of Override request |
| MMI-1d | False acknowledgement of Level Transition |
| MMI-1e | False acknowledgement of Train Trip |
| MMI-1f | False acknowledgement of Track Ahead Free |
| MMI-1g | False shunting request |
| MMI-1h | False acknowledgement of undesired train movement (RAP, RMP and SSS) |
| MMI-2a.1 | False presentation of train speed on the DMI |
| MMI-2a.2 | False presentation of speed (except train speed) or distance on the DMI, including supervision status |
| MMI-2b | False presentation of mode on the DMI |
| MMI-2c | False presentation of track adhesion |

Continued on next page

**Table 32 – continued from previous page**

| Risks | |
|---|---|
| Event Id. | Event Description |
| MMI-2d | Failure to present Entry in FS/OS informa-tion |
| MMI-2e | False presentation of train data/additional data |
| MMI-2f | False presentation of Override status, including false enabling of override selection |
| MMI-2g | Failure to present acknowledgement message to a less restrictive mode |
| MMI-2h | False presentation of TAF request |
| MMI-2i | Failure to present LX “not protected” infor-mation |
| MMI-2j | False presentation of reversing allowed |
| MMI-2k | False presentation of level transition an-nouncement |
| MMI-3 | Falsification of driver’s train data / addi-tional data input stored onboard |
| MMI-4 | Falsification of SR speed/distance data |
| MMI-5 | Falsification of train integrity input |
| MMI-6 | Falsification of Virtual Balise Cover |
| ODO-1 | Incorrect standstill indication |
| ODO-2 | Speed measurement underestimates trains actual speed |
| ODO-3 | Incorrect actual physical speed direction |
| ODO-4 | The confidence interval for distance measurement does not include the real position of the train |
| KERNEL-1 | Balise linking consistency checking failure |
| KERNEL-2 | Balise group message consistency checking failure |
| KERNEL-3 | Failure of radio message correctness check |
| KERNEL-4 | Radio sequencing checking failure |
| KERNEL-5 | Radio link supervision function failure |
| KERNEL-6 | Manage communication session failure |
| KERNEL-7 | Incorrect LRBG |
| KERNEL-8 | Emergency Message Acknowledgement Failure |
| KERNEL-9 | Speed calculation underestimates train speed |
| KERNEL-10 | Functional failure of standstill detection |
| KERNEL-11 | Incorrect traction/braking model (e.g. brake use restrictions) |
| KERNEL-12 | Failure of standstill supervision |
| KERNEL-13 | Failure of backward distance monitoring |
| KERNEL-14 | Failure of reverse movement protection |
| KERNEL-15 | Incorrect cab status (TIU failure) |
| KERNEL-16 | Incorrect train status TIU sleeping/cab status |

Continued on next page

**Table 32 – continued from previous page**

| Risks | |
|---|---|
| Event Id. | Event Description |
| KERNEL-17 | Wrong Acceptance of MA |
| KERNEL-18 | Failure to manage RBC/RBC |
| KERNEL-19 | Failure of train trip supervision in OS, LS and FS |
| KERNEL-20 | Failure of train trip supervision, shunting and SR |
| KERNEL-21 | Incorrect supervision of stop in SR |
| KERNEL-22 | Incorrect current EoA |
| KERNEL-23 | Incorrect train position / train data sent from on-board to trackside |
| KERNEL-24 | Failure of message acknowledgement |
| KERNEL-25 | Incorrect traction/braking model (Accelera-tion only) |
| KERNEL-26 | Deleted |
| KERNEL-27 | Incorrect System Data (e.g. current level) |
| KERNEL-28 | Incorrect confidence interval |
| KERNEL-29 | Failure to shorten MA |
| KERNEL-30 | Incorrect shortening of MA |
| KERNEL-31 | Deleted |
| KERNEL-32 | Failure of loop message consistency checking |
| KERNEL-33 | Wrong processing of MA information |
| KERNEL-34 | Incorrect supervision of MA time-outs (sections and overlaps) |
| TI-1 | Service brake / emergency brake not commanded when required |
| TI-2 | Service brake / emergency brake release commanded when not required |
| TI-3 | Inappropriate sleeping request |
| TI-4 | Incorrect brake status (TIU failure) |
| TI-5 | Incorrect direction controller position report (TIU failure) |
| TI-6a | Loss of Cabin Active signal |
| TI-6b | Wrong Cabin considered as Active |
| EUB-H1 | A balise group is not detected, due to fail-ure of a balise group to transmit a detectable signal |
| EUB-H4 | Transmission of an erroneous telegram interpretable as correct, due to failure within a Balise |
| EUB-H7 | Erroneous localisation of a Balise Group, with reception of valid telegrams, due to failure within Balises (too strong up-link signal) |
| EUB-H8 | The order of reported Balises, with reception of valid telegram, is erroneous due to failure within a Balise (too strong up-link signal) |

Continued on next page

**Table 32 – continued from previous page**

| Risks | |
|---|---|
| Event Id. | Event Description |
| EUB-H9 | Erroneous reporting of a Balise Group in a different track, with reception of valid telegrams, due to failures within Balises (too strong up-link signal) |
| BTM-H1 | A balise group is not detected, due to failure within the onboard BTM function |
| BTM-H4 | Transmission to the on-board kernel of an erroneous telegram, interpretable as correct, due to failure within the onboard BTM function |
| BTM-H7 | Erroneous localisation of a Balise Group, with reception of valid telegrams, due to failure within the on-board BTM function (erroneous threshold function or significantly excessive Tele-powering signal) |
| BTM-H8 | The order of reported Balises, with reception of valid telegrams, is erroneous due to failure within the on-board BTM function (erroneous threshold function or significantly excessive Tele-powering signal) |
| BTM-H9 | Erroneous reporting of a Balise Group in a different track, with reception of valid telegrams, due to failure within the on-board BTM function (erroneous threshold function or significantly excessive Tele-powering signal) |
| OB-EUR-H4 | Radio message corrupted in onboard Euroradio, such that the message appears as consistent |
| TR-EUR-H4 | Radio message corrupted in trackside Euroradio, such that the message appears as consistent |
| LEU-H4 | Transmission of an erroneous telegram / telegrams interpretable as correct, due to failure within the LEU function |
| EUL-H4 | Transmission of an erroneous telegram / telegrams interpretable as correct, due to failure within a Loop |
| LTM-H4 | Transmission of an erroneous telegram / telegrams, interpretable as correct, due to failure within the on-board LTM function |
| RBC-2 | Incorrect radio message sent from RBC Kernel, such that the message appears as consistent |
| RBC-3 | Incorrect radio message from an adjacent RBC, causing incorrect message to ETCS onboard |

# Appendices

## A CAT1: Open Source Development Process Roles and Competence Matrix

**Table 33. CAT1: Open Source Development Process Roles/Competences**

| CAT1: Open Source Development Process Roles/Competences | | | | |
|---|---|---|---|---|
| Code | Role | Responsibilities | Core Competences | Specific Competences /Responsibilities per project |
| OPL | openETCS project Leader | Responsible to guarantee progress<br>Promote that the most appropriate community is engaged in the project life-cycle<br>Ensure that all personnel involved in all phases of the software, tool chain (products) and project life-cycle, including management activities, have the appropriate training, experience and qualifications | Good Project Management Skills<br>Communication Skills<br>Presentation Skills<br>Moderation Skills<br>Risk Management Skills | Not Applicable |
| WPL | WP Leader/Top-level project leader | Make sure the required competence to develop a task is covered by the engaged committers<br>To ensure that all personnel who have responsibilities for the software are competent to discharge those responsibilities<br>Ensure that the parties involved throughout the product life-cycle are independent, to the extent required by the software safety integrity level, in accordance with CENELEC | Project Management Skills<br>Good Technical Knowledge of the Workpackage<br>Communication Skills<br>Presentation Skills<br>Moderation Skills<br>Risk Management Skills | Good Technical Knowledge of the specific Workpackage |

Continued on next page

**Table 33 – continued from previous page**

| CAT1: Open Source Development Process Roles/Competences | | | | |
|---|---|---|---|---|
| Code | Role | Responsibilities | Core Competences | Specific Competences /Responsibilities per project |
| TL | Task Leader/ project leader | Maintains the corresponding backlog | Project Management Skills<br>Technical Knowledge of the Workpackage<br>Communication Skills<br>Presentation Skills<br>Moderation Skills<br>Risk Management Skills | Technical Knowledge of the specific Workpackage. For example:<br>Project: QA activities responsible for the identification, supervision and control of all the processes, methods and tools required to meet the quality targets of the project |
| US | User | Not Applicable | | |
| AD | Adopter | Reuse of the frameworks (within the companies that are contributing to the project and outside of the project), Reuse of the tools (within the companies that are contributing to the project and outside of the project, | Not in the scope of this document | Not in the scope of this document |

Continued on next page

**Table 33 – continued from previous page**

| CAT1: Open Source Development Process Roles/Competences | | | | |
|---|---|---|---|---|
| Code | Role | Responsibilities | Core Competences | Specific Competences /Responsibilities per project |
| CTB | Contributor | Contribute content, code, fixes, tests, documentation, or other work that is part of the Project<br>Provide feedback<br>Help new users<br>Test, report or fix bugs<br>Request new features<br>Write or update documentation<br>Write and update software | Not relevant | See "how to become a committer in openETCS"<br>Good technical skills for the task of the workpackage |
| CMT | Committer | Have the exclusive right to elect new Committers to their Project–no other group, including a parent Project, can force a Project to accept a new Committer.<br>Monitor and contribute to the mailing lists<br>Proactively report problems in the task tracking system, and annotating problem reports with status information, explanations, clarifications, or requests for more information from the submitter | Not relevant | Good technical skills for the task of the workpackage |

# B CAT2: SCRUM Roles and Competence Matrix

**Table 34. CAT2: SCRUM Roles/Competences**

| CAT2: SCRUM Roles/Competences | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Core Competences |
| POw | Product Owner | Managing and prioritizing the Product Backlog<br>Planning the release<br>Software and Tool chain acceptance<br>Understand the value of the project<br>Stakeholder Management<br>We expect the WP-/Task Leader to act in this role | Agile Product Owner Training and Certificate is highly recommended<br>Customer Orientation<br>Deep Technical Knowledge of the Product he/she is responsible for<br>Good knowledge of the use-case of the product<br>Project Management Skills<br>Risk Management Skills |
| ScM | Scrum Master | Team Coach<br>Change Agent<br>Owner of the Impediment Backlog<br>Manage the development process<br>Prepare Burndown charts<br>Identify and eliminate obstacles that prevent the team from achieving their goals<br>Ensures that the team is fully functional and productive<br>Enables close cooperation across all roles and functions<br>Ensure clear communication among everyone involved in the project | Agile Scrum Master Training and Certificate is highly recommended<br>Moderation Skills<br>Team Coaching Skills<br>Experiences in the tasks the team is responsible for |

Continued on next page

**Table 34 – continued from previous page**

| CAT2: SCRUM Roles/Competences | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Core Competences |
| ScT | Scrum Team | Self organizing (organizes itself and its work)<br>Identify obstacles and informing the Scrum Master<br>Development to achieve sprint goals.<br>Implementing test cases<br>Unit and initial Acceptance testing | Some Basic Scrum Training is needed<br>Team needs to cover skills for all tasks needed to develop and release the product<br>Looking at CENELEC, the roles to be covered in each team are<br>Requirement Manager, Designer, Implementer, Tester, Verifier, and Integrator. |

# C CAT3: CENELEC Roles and Competence Matrix for openETCS software product

**Table 35. CAT3: CENELEC Roles/Competences for openETCS application software project**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| PM | openETCS software Project Manager | Identify which roles are needed for the project<br>Verify that at least one person fulfills an identified project role<br>Guarantee the required competence for the project is covered by the effective committers<br>Initialize the distribution of roles between partners to ensure independence of the roles<br>Ensure compliance with the quality management system<br>Responsible to guarantee progress according to scheduled plans<br>Responsible for the delivery and implementation of the software<br>Ensure the compliance and the delivery of safety requirements<br>Approve full and partial products to be delivered by the development process<br>Ensure that records and traceability are maintained throughout the decision making and project<br>Ensure appropriate validation for the project through project partners | Understand requirements of software development process<br>Understand quality, competencies, organizational and management requirements according to relevant standards<br>Understand the requirements of the verification, validation and safety process<br>Able to evaluated the impact of different options for the performance concerning implementation, validation and safety |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| RQM | Requirement manager | Responsible for the software model and source code requirement specification<br>Establishes and maintain traceability to and from the system-level requirements<br>Ensure that software and derived specifications requirements are under system<br>configuration and changes management control.<br>Ensure consistency and completeness of the software requirements specification<br>Develop and maintain documents related to software requirements | experience in railways sector and safety attributes in the railway domain<br>experience with requirements management process and tools<br>knowledges of Technical Specification for Interoperability (TSI) and related CENELEC requirements |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| designer (DES) | Designer | Transform software requirements on acceptable solutions<br>Derive the requirements for the system and software architecture<br>Identify the key design issues that must be resolved to support successful development of the software<br>Allocate the software and derived requirements to the chosen architecture components and interfaces<br>Maintain requirement traceability for the software architecture's requirements, and to and from software requirements<br>Identify suitable derived requirements that address the effectiveness and cost of life-cycle phases following development, such as production and operation<br>Develop and maintain design documentation<br>Ensure that the design documents are under system configuration and changes management control.<br>Design or select design methods and support tools<br>Apply principles and suitable design standards<br>Develop component specifications if it is applicable | Competent in software development in the railway domain<br>Competent in safety design principles<br>Familiarity with methods and tools for design analysis and design testing<br>Ability to work with design constraints for safety relevant software in On-Board systems<br>Understanding of the system constraints created through the TSI<br>Understanding of the relevant parts of EN 50128 like design methods |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| implementer (IMP) | Implementer | Transform design solutions in data, models, source code and finally executable code for the demonstrator<br>Apply safety design principles<br>Apply specific rules for data preparation/codification<br>Perform analysis to verify intermediate results<br>Develop and maintain implementing documents comprising the methods, types of data, models and listings applied<br>Maintain traceability to and from the design<br>Maintain the generated or modified data/codes/models under system configuration and changes management control. | Competent in safety relevant software implementation for embedded systems<br>Competent in the implementation language and supporting tools<br>Capable of applying the specified coding standards and programming styles<br>Understanding of the system constraints created through the On-Board hardware respectively the demonstrator<br>Understanding of the relevant parts of EN 50128 like design methods |
| tester (TST) | Tester | Ensure the test activities planning<br>Develop tests specification (goals and cases)<br>Ensure traceability of test objectives to specified software requirements<br>Ensure traceability of test cases to the specified tests objectives<br>Ensure that the planned tests are implemented and performed<br>Identify deviations from the expected results and record in the test reports<br>Communicate deviation to the authority in charge of the changes management for evaluation and decision making<br>Record the results reports<br>Select the equipment for testing the software | Competent in ETCS specification, used means of description (model/ source code), used train and track parameter and other application data source<br>Competent in various test approaches/methods to identify to identify the most appropriate method or combination of methods for every aspect of an artifact<br>Capable of deriving test cases from TSI (specifically Subset 26) and the specification model<br>Understanding of the relevant parts of EN 50128 like test methods |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| verifier (VER) | Verifier | Develop a software (SW) Verification Plan<br>Check the documented test suitability (completeness, coherency, relevance, traceability) with the verification objectives<br>Identify anomalies, evaluate in terms of the risk, record them and communicate them to the authority in charge of the changes management for evaluation and decision making<br>Manage the verification process (revision, integration and testing) and ensure the independence of the activities as needed<br>Develop a verification report with the results of the verification activities | Competent in ETCS specification, used means of description (model/ source code), used train and track parameter and other application data source<br>Competent in various verification approaches/methods to identify the most appropriate method or combination of methods for every aspect of an artifact<br>Capable of deriving verification procedures from TSI (specifically Subset 26) and the specification model<br>Understanding of the relevant parts of EN 50128 like verification methods |
| integrator (INT) | Integrator | Manage the integration process using software baselines<br>Develop sw and sw /hw integration test specification for sw components based on the specifications and on the designer's components architecture<br>Develop and maintain records of the integration activities<br>Identify integration anomalies; record them and communicate them to the authority in charge of the changes management for evaluation and decision making<br>Develop a report of components and the overall system integration covering the integration results | Competent in ETCS specification, used programming language, used API and demonstrator hardware<br>Competent in various integration approaches/methods to identify the most appropriate method or combination of methods for the demonstrator implementation<br>Understanding the design and functionality requirements for intermediated development levels<br>Capable of deriving integrator tests from the set of integrated functions<br>Understanding of the relevant parts of EN 50128 like integration tests |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| validator (VAL) | Validator | Develop a Validation Plan specifying the main tasks and activities for the sw validation<br>Agree on the Validation Plan with the assessor<br>Review Sw requirements in relation to their intended use/environment<br>Ensure sw fulfill all sw requirements<br>Evaluate the assessment of the software process and of the software according to CENELEC requirements and the assigned SIL<br>Review the verification and tests correctness, consistency and suitability<br>Check the correctness, consistency and suitability of the test cases and executed tests<br>Ensure that all validation plan activities are carried out<br>Review and classify deviations, evaluate in terms of the risk, record them and communicate them to the authority in charge of the changes management for evaluation and decision making<br>Provide recommendation about sw suitability<br>Record Validation Plan deviations<br>Conduct audits, inspections or reviews of the overall project at various stages of development as may be appropriate<br>Review and analyse validation reports of the previous sw<br>Check whether the developed solutions are traceable to the sw requirements<br>Ensure that records associated hazardous situations and nonconformances are reviewed<br>Ensure that all dangerous situations are appropriately resolved<br>Develop a Validation Report<br>Express their agreement or disagreement about the sw version | Competent in ETCS On-Board units<br>Experience in safety attributes for train control systems<br>Competent in various validation approaches/methods to identify the most appropriate method or combination of methods for the demonstrator implementation<br>Capable of deriving types of validation evidence required for the TSI with respect to the train control functionality<br>Capable to combine different sources and types of evidence and synthesize an overall view about fitness for purpose or constraints and limitations of the On-Board application<br>Overall software understanding and perspective including the general railway environment<br>Understanding the requirements of EN 50128 |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| assessor (ASR) | Assessor | Develop an assessment plan and communication with safety authority and client organization<br>Evaluate the assessment of the software process and of the software according to CENELEC requirements and the assigned SIL<br>Assess the project team and the organization competences for the sw development<br>Evaluate the Verification & Validation activities and the supporting evidences<br>Evaluate quality management systems adopted for the sw development<br>Evaluate the changes management and the Configuration Management Systems and their use<br>Identify and assess risk in terms of any deviation from the sw requirements in the evaluation report<br>Ensure the evaluation Plan is implemented<br>Performs independent checks of: The development process (audits) and the products safety functions (spot checks) during different development phases.<br>Should perform audits, based on the Safety plan, of the Quality and Safety management systems of the Supplier, the Infrastructure owner and the Operator and be convinced that these systems works<br>The Assessor can also perform spot checks on detailed technical issues to see that safety functions are correctly implemented. The safety functions key documentation (Hazard Log, Safety Requirements and Safety Case) should be examined too.<br>Give an opinion on the validity of sw developed for its intended use detailing any constraints, application conditions and observations for risk control appropriate<br>Develop an assessment report and maintain records about the assessment process | Competences in the railway domain and technology specifically concerning On-Board systems<br>Acceptance/License from a recognized safety authority<br>Continually gained sufficient level of experience in the safety principles and the application of these principles within the railway domain<br>Competence to evaluated that a suitable method or combination of methods in a given context have been applied<br>Understanding the relevant safety, human resource, technical and quality management processes to fulfill the requirements of the EN 50128<br>Competence in assessment approaches/ methods<br>Capable to combine different sources and types of evidence and synthesize an overall view about fitness for purpose or constraints and limitations of the On-Board application<br>Overall software understanding and perspective including the general railway environment<br>Ability to judge the adequacy of all development processes (like quality management, configuration management, validation and verification processes)<br>Understanding the requirements of EN 50128 |

Continued on next page

**Table 35 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS application software project | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| CM | Configuration Manager | Responsible for the configuration management plan [26] System configuration management owner<br>Establish that all sw components are clearly identified and have independent versions within the system configuration management<br>Prepare the published release notes mentioning incompatible versions of sw components | Competences in software configuration management<br>Understanding the requirements of EN 50128 |

# D CAT3: CENELEC Roles and Competence Matrix for openETCS Tool Chain product

**Table 36. CAT3: CENELEC Roles/Competences for openETCS Tool Chain product**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| PM | openETCS project Manager | Guarantee the required competence for the project is covered by the effective committers<br>Identify which roles are needed for the project<br>Verify that at least one person has been identified per project role<br>ensure the independence of the roles according to CENELEC<br>ensure compliance with the quality management system<br>Responsible to guarantee progress according to scheduled plans<br>devote sufficient resources to perform the task, including security tasks<br>responsible for the delivery and implementation of the software<br>ensure the compliance and the delivery of security requirements<br>provide enough time for proper implementation and enforcement of security tasks<br>approve full and partial products to be delivered by the development process<br>ensure that records and traceability are maintained throughout the decision making and project<br>ensure that it has appointed an appropriate validator for the project according to CENELEC | Understand requirements of software development process<br>Understand quality, competencies, organizational and management requirements according to relevant standards<br>Understand the requirements of the verification, validation and safety process<br>Able to evaluated the impact of different options for the performance concerning implementation, validation and safety |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| RQM | Requirement manager | Responsible for the Software requirement specification<br>Establishes and maintain traceability to and from the system-level requirements<br>ensure that tool chain and derived specifications requirements are under system configuration and changes management control.<br>ensure consistency and completeness of the tool chain requirements specification<br>develop and maintain documents related to tool chain requirements | experience in railways sector and safety attributes in the railway domain<br>experience with requirements management process and tools<br>knowledges of TSI and related CENELEC requirements |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| DES | Designer | Transform software requirements on acceptable solutions<br>Derive the requirements for the system and software architecture<br>Identify the key design issues that must be resolved to support successful development of the software<br>Allocate the tool chain and derived requirements to the chosen architecture components and interfaces<br>Maintain requirement traceability for the software architecture's requirements, and to and from software requirements<br>Identify suitable derived requirements that address the effectiveness and cost of life-cycle phases following development, such as production and operation<br>Develop and maintain design documentation<br>Ensure that the design documents are under system configuration and changes management control.<br>Design or select design methods and support tools<br>Apply principles and suitable design standards<br>Develop component specifications if it is applicable | Competent in software development in the railway domain<br>Competent in safety design principles<br>Familiarity with methods and tools for design analysis and design testing<br>Ability to work with design constraints for safety relevant software in On-Board systems<br>Understanding of the system constraints created through the TSI<br>Understanding of the relevant parts of EN 50128 like design methods |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| IMP | Implementer | Transform design solutions in data, source code, models and / or other design representations<br>Apply design principles<br>Apply specific rules for data preparation/codification<br>Perform analysis to verify intermediate results<br>Develop and maintain implementing documents comprising the methods, types of data, models and listings applied<br>Maintain traceability to and from the design<br>Maintain the generated or modified data/codes/models under system configuration and changes management control. | Competent in safety relevant software implementation for embedded systems<br>Competent in the implementation language and supporting tools<br>Capable of applying the specified coding standards and programming styles<br>Understanding of the system constraints created through the On-Board hardware respectively the demonstrator<br>Understanding of the relevant parts of EN 50128 like design methods |
| TST | Tester | Ensure the test activities planning<br>Develop tests specification (goals and cases)<br>Ensure traceability of test objectives to specified software requirements<br>Ensure traceability of test cases to the specified tests objectives<br>Ensure that the planned tests are implemented and performed<br>Identify deviations from the expected results and record in the test reports<br>Communicate deviation to the authority in charge of the changes management for evaluation and decision making<br>Record the results reports<br>Select the equipment for testing the software | Competent in ETCS specification, used means of description (model/ source code), used train and track parameter and other application data source<br>Competent in various test approaches/methods to identify to identify the most appropriate method or combination of methods for every aspect of an artifact<br>Capable of deriving test cases from TSI (specifically Subset 26) and the specification model<br>Understanding of the relevant parts of EN 50128 like test methods |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| INT | Integrator | Manage the integration process using software baselines<br>Develop sw and sw /hw integration test specification for sw components based on the specifications and on the designer's components architecture<br>Develop and maintain records of the integration activities<br>Identify integration anomalies; record them and communicate them to the authority in charge of the changes management for evaluation and decision making<br>Develop a report of components and the overall system integration covering the integration results | Competent in ETCS specification, used programming language, used API and demonstrator hardware<br>Competent in various integration approaches/methods to identify the most appropriate method or combination of methods for the demonstrator implementation<br>Understanding the design and functionality requirements for intermediated development levels<br>Capable of deriving integrator tests from the set of integrated functions<br>Understanding of the relevant parts of EN 50128 like integration tests |
| VER | Verifier | Develop a SW Verification Plan<br>Check the documented test suitability (completeness, coherency, relevance, traceability) with the verification objectives<br>Identify anomalies, evaluate in terms of the risk, record them and communicate them to the authority in charge of the changes management for evaluation and decision making<br>Manage the verification process (revision, integration and testing) and ensure the independence of the activities as needed<br>Develop a verification report with the results of the verification activities | Competent in ETCS specification, used means of description (model/ source code), used train and track parameter and other application data source<br>Competent in various verification approaches/methods to identify the most appropriate method or combination of methods for every aspect of an artifact<br>Capable of deriving verification procedures from TSI (specifically Subset 26) and the specification model<br>Understanding of the relevant parts of EN 50128 like verification methods |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| VAL | Validator | Develop a Validation Plan specifying the main tasks and activities for the sw validation<br>Agree on the Validation Plan with the assessor<br>Review Sw requirements in relation to their intended use/environment<br>Ensure sw fulfil all sw requirements<br>Evaluate the assessment of the software process and of the software according to CENELEC requirements and the assigned SIL<br>Review the verification and tests correctness, consistency and suitability<br>Check the correctness, consistency and suitability of the test cases and executed tests<br>Ensure that all validation plan activities are carried out<br>Review and classify deviations, evaluate in terms of the risk, record them and communicate them to the authority in charge of the changes management for evaluation and decision making<br>Provide recommendation about sw suitability<br>Record Validation Plan deviations<br>Conduct audits, inspections or reviews of the overall project at various stages of development as may be appropriate<br>Review and analyse validation reports of the previous sw<br>Check whether the developed solutions are traceable to the sw requirements<br>Ensure that records associated hazardous situations and nonconformances are reviewed<br>Ensure that all dangerous situations are appropriately resolved<br>Develop a Validation Report<br>Express their agreement or disagreement about the sw version | Competent in ETCS On-Board units<br>Experience in safety attributes for train control systems<br>Competent in various validation approaches/methods to identify the most appropriate method or combination of methods for the demonstrator implementation<br>Capable of deriving types of validation evidence required for the TSI with respect to the train control functionality<br>Capable to combine different sources and types of evidence and synthesize an overall view about fitness for purpose or constraints and limitations of the On-Board application<br>Overall software understanding and perspective including the general railway environment<br>Understanding the requirements of EN 50128 |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| ASR | Assessor | Develop an evaluation Plan<br>Evaluate the assessment of the software process and of the software according to CENELEC requirements and the assigned SIL<br>Assess the project team and the organization competences for the sw development<br>Evaluate the Verification & Validation activities and the supporting evidences<br>Evaluate quality management systems adopted for the sw development<br>Evaluate the changes management and the Configuration Management Systems and their use<br>Identify and assess risk in terms of any deviation from the sw requirements in the evaluation report<br>Ensure the evaluation Plan is implemented<br>Performs independent checks of: The development process (audits) and the products safety functions (spot checks) during different development phases.<br>Should perform audits, based on the Safety plan, of the Quality and Safety management systems of the Supplier, the Infrastructure owner and the Operator and be convinced that these systems works<br>The Assessor can also perform spot checks on detailed technical issues to see that safety functions are correctly implemented. The safety functions key documentation (Hazard Log, Safety Requirements and Safety Case) should be examined too.<br>Give an opinion on the validity of sw developed for its intended use<br>Develop an evaluation report and maintain records about the evaluation process | Competences in the railway domain and technology specifically concerning On-Board systems<br>Acceptance/License from a recognized safety authority<br>Continually gained sufficient level of experience in the safety principles and the application of these principles within the railway domain<br>Competence to evaluated that a suitable method or combination of methods in a given context have been applied<br>Understanding the relevant safety, human resource, technical and quality management processes to fulfill the requirements of the EN 50128<br>Competence in assessment approaches/ methods<br>Capable to combine different sources and types of evidence and synthesize an overall view about fitness for purpose or constraints and limitations of the On-Board application<br>Overall software understanding and perspective including the general railway environment<br>Ability to judge the adequacy of all development processes (like quality management, configuration management, validation and verification processes)<br>Understanding the requirements of EN 50128 |

Continued on next page

**Table 36 – continued from previous page**

| CAT3: CENELEC Roles/Competences for openETCS Tool Chain product | | | |
|---|---|---|---|
| Code | Role | Responsibilities | Competences |
| CM | Configuration Manager | Responsible for the configuration management plan [26] System configuration management owner<br>Establish that all sw components are clearly identified and have independent versions within the system configuration management<br>Prepare the published release notes mentioning incompatible versions of sw components | Competences in software configuration management<br>Understanding the requirements of EN 50128 |

# E Methods & Tools for Application Software

| Software Requirements Specification Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Formal Methods | HR | Yes | Formal Methods are applied where possible. However, we will start in the ITEA project only with a small portion to proof the concept. |
| 2 | Modelling | HR | Yes | The models in the requirements specification phase are to be provided in SysML (Papyrus). The link to traceability is given based on ProR extension of the openETCS tool set. |
| 3 | Structured Methodology | HR | No | Not relevant since we are using Formal Methods and Modeling. |
| 4 | Decision Table | HR | No | Not relevant since we are using Formal Methods and Modelling. |
| Justification: **(To be fulfilled)** | | | | |
| *openETCS uses a combination of natural language Software Requirement Specifications, which are derived from the SUBSET-26 SRS, and Modelling of these Requirements in SysML and SCADE. Thereby, the approach fulfils the EN 50128 Table A.2 requirements for Techniques for Software Requirements Specification* | | | | |

**Table 37. Software Requirements Specification Phase**

**Table 38. Software Architecture Phase**

| Software Architecture Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Defensive Programming | HR | Yes | Inherent with the chosen Scade Suite solution. |
| 2 | Fault Detection & Diagnosis | HR | Yes | Inherent with the chosen Scade Suite solution. |
| 3 | Error Correcting Codes | - | No | |
| 4 | Error Detecting Codes | HR | No | Error Detecting codes are not relevant in the scope of the openETCS Application. They might be implemented in a product being based around openETCS. |
| 5 | Assertion Programming | HR | Yes | Handling of assertions will be part of the scade model. |
| 6 | Safety Bag Techniques | R | No | |

Continued on next page

**Table 38 – continued from previous page**

| Software Architecture Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 7 | Diverse Programming | HR | No | Diverse Programming is an option for a real openETCS product. Since it as to be supported by the architecture of the product the method can not be implemented in the generic approach of openETCS. |
| 8 | Recovery Block | R | No | Recovery Block is an option for a real openETCS product. Since it as to be supported by the implementation approach the method can not be implemented in the generic approach of openETCS. |
| 9 | Backward Recovery | NR | No | Not suited for openETCS. |
| 10 | Forward Recovery | NR | No | Not suited for openETCS. |
| 11 | Retry Fault Recovery Mechanisms | R | No | Retry Fault Recovery Mechanisms is an option for a real openETCS product. Since it as to be supported by the implementation approach the method can not be implemented in the generic approach of openETCS. |
| 12 | Memorising Executed Cases | HR | No | Graceful degradation is an option for a real openETCS product. Since it as to be supported by the architecture of the product the method can not be implemented in the generic approach of openETCS. |
| 13 | Artificial Intelligence - Fault Correction | NR | No | |
| 14 | Dynamic Reconfiguration of software | NR | No | |
| 15 | Software Error Effect Analysis | HR | No | SEEA is not superior compared to the chosen formal approach of openETCS. |
| 16 | Graceful Degradation | HR | No | |
| 17 | Information Hiding | - | No | |
| 18 | Information Encapsulation | HR | Yes | Information Encapsulation is covered by the modelling guideline. |

Continued on next page

**Table 38 – continued from previous page**

| Software Architecture Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 19 | Fully Defined Interface | M | Yes | This concept is part of both, the architecture model and the actual implementation in Scade. In Scade the method is inherent. |
| 20 | Formal Methods | HR | Yes | According to the openETCS project proposal formal methods are a the main part part of the concept for safety critical software. However, not the full scope will be possible to implement in the Framework of the Itea project. A proof of concept is required for this option. |
| 21 | Modelling | HR | Yes | The architecture model is to be implemented in SysML / Papyrus. |
| 22 | Structured Methodology | HR | No | Not relevant since we are using Formal Methods and Modelling. |
| 23 | Modelling supported by computer aided design and specification tools | HR | Yes | The openETCS Tool Chain include interfaced design and specification tools. |
| Justification: **(To be fulfilled)** | | | | |
| *Techniques use in openETCS cover all relevant architecture and design issues, but due to the generic concept of openETCS Recovery issues are not covered by any methodology. The applied combination of Modelling and Formal Methods guaranties a design consistency and completeness.* | | | | |

**Table 39. Software Design and Implementation Phase**

| Software Design and Implementation Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Formal Methods | HR | Yes | In openETCS formal methods are a the main methodology to design safety critical software. However, not the full scope will be possible to implement in the Framework of the Itea project. A proof of concept is required for this option. |
| 2 | Modelling | HR | Yes | The design model is implemented in Scade. |

Continued on next page

**Table 39 – continued from previous page**

| Software Design and Implementation Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 3 | Structured Methodology | HR | No | Not relevant since we are using Formal Methods and Modelling. |
| 4 | Modular Approach | M | Yes | Well supported with Scade. Different Moduls are designed and implement as independent blocks based on the overall architecture. |
| 5 | Components | HR | Yes | Well supported with Scade. All internal interfaces are fully defined and the data access is limited due to data flows. |
| 6 | Design and Coding Standards | M | Yes | Part of D2.4 and the Modelling Description of Work. |
| 7 | Analysable Programs | HR | Yes | Static Analysis is supported by the Scade Suite Code generator and is part of the development guideline. |
| 8 | Strongly Typed Programming Language | HR | Yes | The chosen Scade Language is strongly typed. |
| 9 | Structured Programming | HR | Yes | The Scade Suite encourages for a reduction of complexity in modelling (1 page approach). The structure of the code is as such is made by the Scade code generator. The modeller has an indirect impact on the generation. |
| 10 | Programming Language | HR | Yes | Scade approach is a certified language for SIL 4 implementation. |
| 11 | Language Subset | HR | Yes | Not relevant here since the Scade approach is a certified language for SIL 4 implementation. |
| 12 | Object Oriented Programming | R | No | |
| 13 | Procedural Programming | HR | No | Not relevant here since the Scade approach is a certified language for SIL 4 implementation. |
| 14 | Metaprogramming | R | No | |
| Justification: **(To be fulfilled)** | | | | |
| *The openETCS techniques based on use of a structured, strongly typed programming language in SCADE fulfils design and implementation requirements.* | | | | |

**Table 40. Verification and Testing Phase**

| Verification and Testing Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Formal Proof | HR | Yes | Various formal methods are used to check the SysML and SCADE modells and to prove required properties. Specifics are documented in he V&V plan and all corresponding reports. |
| 2 | Static Analysis | HR | Yes | Static checks covering consistency of interfaces and data types are implemented in Papyrus. Further static analysis as boundary checks and data flow analysis are covered during the SCADE design. Details are presented in the V&V plan |
| 3 | Dynamic Analysis and Testing | HR | Yes | openETCS uses a variety of dynamic analysis and testing approach mainly on SCADE modell and C-Code level e.g. Test cases from boundary Values and Performance Modelling. Details are presented in the V&V plan. |
| 4 | Metrics | R | Yes | Various metrics are evaluated on the C-Code level. Further metrics are already addressed during the SCADE modelling. |
| 5 | Traceability | M | Yes | Traceability between all abstraction levels is covered by tool supported traces and manual traceability reviews. |
| 6 | Software Error Effect Analysis | HR | Yes | The Error Analysis is performed during the modelling process on various levels using the modell-based approach for related component analysis. |
| 7 | Test Coverage for code | HR | Yes | As openETCS uses a modell based test generation the coverage follows a complete coverage principal for all relevant data flows. Details are given in the V&V plan. |

Continued on next page

**Table 40 – continued from previous page**

| Verification and Testing Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 8 | Functional/ Black-box Testing | M | Yes | Functional test are the main focus for all parts, which are not covered by formal proofs. Black-box test are performed for validation aspects. Details are given in the V&V plan. |
| 9 | Performance Testing | HR | Yes | Performance test mainly response timing are checked where generic timing constrains can be identified. Due to the generic concept of openETCS further tests are required to demonstrated a sufficient implementation. |
| 10 | Interface Testing | HR | Yes | Interface tests are performed on various levels to ensure consistence and correctness. Hardware tests are not part of openETCS. |
| Justification: **(To be fulfilled)** | | | | |
| The openETCS methodology covers at least in parts all relevant verification techniques. Due to the generic concept of openETCS hardware related tests are not part of openETCS. Several tests will only be implemented on parts of the systems as a proof of concept. | | | | |

**Table 41. Integration Phase**

| Integration Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Functional and Black-box Testing | HR | No | Due to the generic approach of openETCS implementation is only done in a demonstrator context, which does not satesfy any needs for systematic implementation tests. |
| 2 | Performance Testing | HR | No | Due to the generic approach of openETCS implementation is only done in a demonstrator context, which does not satesfy any needs for systematic implementation tests. |
| Justification: **(To be fulfilled)** | | | | |

Continued on next page

**Table 41 – continued from previous page**

| Integration Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| Implementation is only part of the generic functional openETCS development up to the API level. A follow-up product has to deal with all aspects required to proof its specific hardware/software implementation. | | | | |

**Table 42. Overall Software Testing Phase**

| Overall Software Testing Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Performance Testing | M | Yes | Performance test mainly response timing are checked where generic timing constrains can be identified. Due to the generic concept of openETCS further tests are required to demonstrated a sufficient implementation. |
| 2 | Functional and Black-box Testing | M | Yes | Functional test are the main focus for all parts, which are not covered by formal proofs. Black-box test are performed for validation aspects. Details are given in the V&V plan. |
| 3 | Modelling | R | Yes | As the openETCS software development is based on SysML and SCADE modells, overall software tests are also performed based on test modells. Therefore, various methods are used to address different validation aspects. Deatils are given in the V&V plan. |
| Justification: **(To be fulfilled)** | | | | |
| The openETCS project uses different model-based testing approaches for validation in combination with performance and functional tests. Due to the development constraints not all ETCS parts will be completely validated during the project. | | | | |

**Table 43. Software Analysis Techniques**

| Software Analysis Techniques | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Static Software Analysis | HR | Yes | Static Software Analysis performed based on SysML and SCADE modells. Details are presented in the V&V plan. |
| 2 | Dynamic Software Analysis | HR | Yes | Dynamic Software Analysis performed by tests and simulation of SCADE modells. Details are presented in the V&V plan. |
| 3 | Cause Consequence Diagrams | R | No | Indirectly covered by SysML and SCADE modell analysis |
| 4 | Event Tree Analysis | R | No | Indirectly covered by SysML and SCADE modell analysis |
| 5 | Software Error Effect Analysis | HR | Yes | The Error Analysis is performed during the modelling process on various levels using the modell-based approach for related component analysis. |
| Justification: **(To be fulfilled)** | | | | |
| Software Analysis is performed based on SysMl and SCADe modells and additional model-based tests. | | | | |

**Table 44. Software Quality Assurance Techniques**

| Software Quality Assurance Techniques | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Accredited to EN ISO 9001 | HR | No | The openETCS project itself can not be accredited, but most partners are. |
| 2 | Compliant with EN ISO 9001 | M | Yes | Overall project process follow those basic guidelines. |
| 3 | Compliant with ISO/IEC 90003 | R | No | Standard cn not be applied directly to research projects as openETCS. |
| 4 | Company Quality System | M | Yes | openETCS has defined project specific quality principals and uses various review systems to enforce these rules. |
| 5 | Software Configuration Management | M | Yes | Has been specified in detail in the Configuration Management plan |

Continued on next page

**Table 44 – continued from previous page**

| Software Quality Assurance Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 6 | Checklists | HR | Yes | Checklists are use in certain phases during the development, but due to the modelling do not present the main work documentation. |
| 7 | Traceability | M | Yes | Traceability is ensured between all development phases by the use of different traces implemented in the openETCS Tool Chain and manual traceability reviews. |
| 8 | Data Recording and Analysis | M | Yes | All development activities are performed and respectively recorded via the github system. |
| Justification: **(To be fulfilled)** | | | | |
| As openETCS is a research project specific quality processes and tools have been applied to ensure respective standards. | | | | |

**Table 45. Software Maintenance Phase**

| Software Maintenance Phase | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Impact Analysis | M | Yes | Performed during every mayor software iteration. |
| 2 | Data Recording and Analysis | M | Yes | Performed via github. |
| Justification: **(To be fulfilled)** | | | | |
| Due to the agile work in openETCS via github maintenance is performed during every iteration and controlled via standard change control. | | | | |

**Table 46. Data Preparation Techniques**

| Data Preparation Techniques | | | | |
|---|---|---|---|---|
| Code | Method/Technique | SIL 4 | Applied (Yes/No) | Details and References |
| 1 | Tabular Specification Methods | R | No | As ETCS is a dynamic system with complex data relations tables provide only limited options for data specifications. |
| 2 | Application specific language | R | No | Not relevant in the openETCS model-based development process based on SCADE modells. |
| 3 | Simulation | HR | Yes | Using SCADE modells and further validation modells. |
| 4 | Functional testing | M | Yes | Using SCADE modells and further validation modells. |
| 5 | Checklists | M | Yes | Depending on typ of input and configuration data. |
| 6 | Fagan inspection | R | No | Traditional review process are used in openETCS. |
| 7 | Formal design reviews | HR | Yes | Based on formal modells used. |
| 8 | Formal proof of correctness (of data) | HR | Yes | Based on formal modells used. |
| 9 | Walkthrough | HR | Yes | Applied in different variations of the review process. |
| Justification: **(To be fulfilled)** | | | | |
| As openETCS is a genric research project data is only prepared for specific aspects. Details are defined in the corresponding sections of the V&V reports. | | | | |

**Table 47. Quality mechanisms for Safe deployment**

| Quality mechanisms for Safe deployment | Technique & Approach |
|---|---|
| Software Self-identification Mechanisms (9.1.4.11) | **(To be fulfilled)** |
| Error detection and/or avoidance mechanisms during deployment process (store, transfer, transmission and/or duplication of code operations) (9.1.4.20) | **(To be fulfilled)** |
| Automatic detection and safe management of incompatible components/versions (9.1.4.8, 9.1.4.9) | **(To be fulfilled)** |
| Provision of appropriate and accurate diagnostic information | **(To be fulfilled)** |
| Safe Roll back capabilities | **(To be fulfilled)** |

## References

[1] CENELEC. EN50126 railway applications - the specification and demonstration of reliability, availability, maintainability and safety (rams), March 2000.

[2] CENELEC. EN50129 railway applications - communication, signalling and processing systems - safety related electronic systems for signalling, December 2003.

[3] CENELEC. EN50128 railway applications - communication, signalling and processing systems - software for railway control and protection systems, June 2011.

[4] Eclipse Community. Eclipse development process.

[5] Izaskun de la Torre. Change/problem management process.

[6] Izaskun de la Torre. Qa plan backlog.

[7] Izaskun de la Torre. Traceability matrix.

[8] Izaskun de la Torre. Training process.

[9] ERTMS. UNISIG subset 023, glossary of terms and abbreviations.

[10] ERTMS. UNISIG subset 026, system requirements specification - baseline 3, January 2010.

[11] International Organization for Standardization.

[12] Ainhoa Gracia. Document control process.

[13] Ainhoa Gracia. Review process.

[14] Ainhoa Gracia. Revision process.

[15] Bernd Hekele. Contribution guidelines.

[16] Bernd Hekele. Grieving handling process.

[17] Bernd Hekele. openetcs internal assessment.

[18] Bernd Hekele. Openetcs ip policy.

[19] Bernd Hekele. Project co-operation agreement.

[20] Jonas Helming. Commiter election guideline.

[21] Jonas Helming. Committer approvement process.

[22] Jonas Helming. openetcs development process.

[23] Hardi Hungar Marc Behrens. Openetcs validation & verification plan.

[24] Stefan Rieger. openETCS publishing guideline.

[25] openETCS Projekt Team. openETCS full project proposal, 2012.

[26] Jürgen Weiss. Software configuration management plan.

[27] Jan Welte. Competence matriz template.

[28] xxx. Expert database template.

[29] xxx. Expert election guideline.