

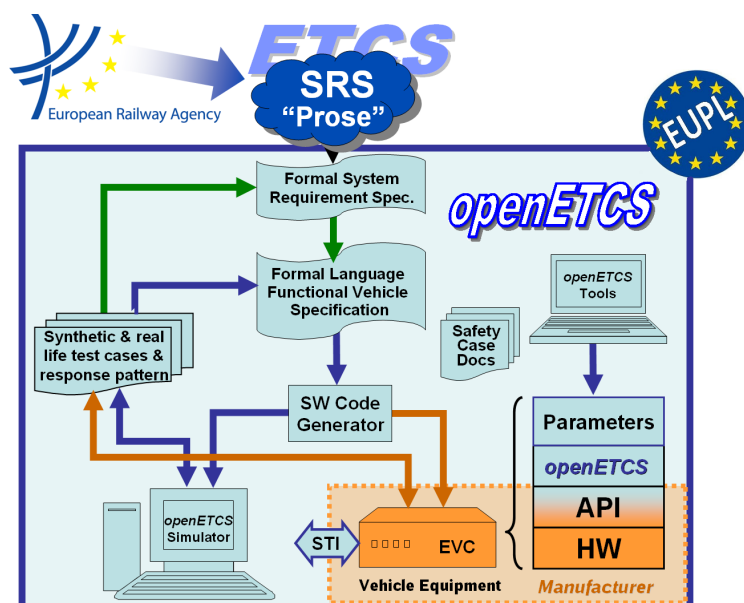
## Work-Package 1: “Management”

# Software Configuration Management Plan

**DRAFT**

Jürgen Weiss

August 9, 2013



**Funded by:**



Federal Ministry  
of Education  
and Research

Région de  
Bruxelles-  
Capitale

GOBIERNO  
DE ESPAÑA



TRIA, ENERGÍA

This page is intentionally left blank

**Work-Package 1: “Management”**

**OETCS/WP1/D1.3  
August 9, 2013**

# Software Configuration Management Plan

**DRAFT**

Jürgen Weiss

Alstom  
Friedrichstrasse 149  
10117 Berlin, Germany

Description of work

Prepared for openETCS@ITEA2 Project

**Disclaimer:** This work is licensed under the "openETCS Open License Terms" (oOLT) dual Licensing: European Union Public Licence (EUPL v.1.1+) AND Creative Commons Attribution-ShareAlike 3.0 – (cc by-sa 3.0)

THE WORK IS PROVIDED UNDER openETCS OPEN LICENSE TERMS (oOLT) WHICH IS A DUAL LICENSE AGREEMENT INCLUDING THE TERMS OF THE EUROPEAN UNION PUBLIC LICENSE (VERSION 1.1 OR ANY LATER VERSION) AND THE TERMS OF THE CREATIVE COMMONS PUBLIC LICENSE ("CCPL"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS OLT LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

<http://creativecommons.org/licenses/by-sa/3.0/>

<http://joinup.ec.europa.eu/software/page/eupl/licence-eupl>

# Contents

Document History .....	5
1 Introduction.....	6
1.1 Purpose .....	6
1.2 Scope .....	6
1.3 Implementation responsibility .....	6
1.4 References, Guidelines and Standards.....	7
1.5 Definitions and acronyms .....	8
2 Configuration Management Tools.....	9
2.1 Tools.....	9
2.2 Repositories .....	9
3 Configuration Identification .....	9
3.1 Configuration items identification .....	10
3.2 Configuration item management .....	10
4 Configuration Management Organization and Planning .....	10
4.1 CM roles assignment.....	10
4.2 Responsibilities and Resources .....	10
5 Configuration Control/Change Management.....	10
5.1 Change management process and procedure .....	10
5.2 Change Request .....	11
5.3 Change Control Board meeting organization.....	11
5.4 Change control board reporting .....	11
6 Configuration Audits .....	11
6.1 Responsibilities .....	11
6.2 Audit content .....	11
6.3 Non Conformance follow up and status reporting .....	12
7 Baselines .....	12
7.1 Baselines formalism .....	12
7.2 Baselines diffusion .....	13
7.3 Baselines schedule .....	13
References .....	13

# Figures and Tables

**Figures**

**Tables**

Table 1. Documentation History ..... 5

Table 2. References ..... 7

Table 3. Guidelines ..... 7

Table 4. Standards..... 8

Table 5. Definitions and acronyms ..... 8

## Document History

**Table 1. Documentation History**

Version	Date	Chapters modified	Reason	Name
0.0.0	09.08.2013	All New	Frame and Basics definition	Jürgen Weiss (Alstom, Berlin)

# 1 Introduction

## 1.1 Purpose

This document is the Software Configuration Management Plan (SCMP) for the OpenETCS project. It identifies the functional attributes of the software at various points in time. The SCMP performs systematic control of changes to the identified attributes for the purpose of maintaining software integrity and traceability. This document describes the applicable software configuration management directives which shall be applied by the software development team. Configuration Management (CM) is used to handle changes systematically so that a system maintains its integrity over time. The Software Configuration Management Plan (SCMP) 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 a configuration management process are:

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

This Software Configuration Management Plan (SCMP) has to be written in accordance to the IEEE Std 828<sup>TM</sup>-2005 and must meet the requirements of CENELEC EN 50128.

## 1.2 Scope

The SCMP provides technical and supervising direction to the development and implementation of tools, processes, services, procedures, functions, and resources required to successfully develop and support a complex system. During the OpenETCS project the Scrum methodology will be used. Scrum is an iterative, incremental framework for projects and products or application development. It structures development in cycles of work called Sprints. These iterations are short in time, no more than one month each, and take place one after the other without interruption.

## 1.3 Implementation responsibility

Responsible: Software Configuration Manager  
Accountable: Quality Assurance Manager



#### 1.4 References, Guidelines and Standards

References				
Internal Code	Name	Version/ Edition/ Date	Repository	Responsible
[2]	Quality Assurance Plan	0.9.5.4	governance	Izaskun de la Torre
[13]	Full Project Proposal (FPP)	3.0	management	Klaus-Rüdiger Hase
[8]	OpenECTS IP Policy	0.1	ecosystem	Bernd Hekele
[7]	OpenETCS Internal Assessment Plan		internal-assessment	Cyril Cornu
[11]	OpenETCS Validation & Verification Plan	01	validation	Marc Behrens Hardi Hungar

**Table 2. References**

Guidelines				
Internal Code	Name	Version/ Edition/ Date	Repository	Responsible
[6]	Contribution guidelines	01	ecosystem	Bernd Hekele
[9]	Committer Election Guideline		ecosystem	Jonas Helming
[12]	openETCS Publishing Guideline		Dissemination	Stefan Rieger
[? ]	Expert Election Guideline		governance	<i>To be defined</i>

**Table 3. Guidelines**

Standards				
Internal Code	Name	Version/ Edition/ Date	Repository	Responsible
[10]	IEEE Std 828 Standard for Software Configuration Management Plans	IEEE 828-2012	-	Institute of Electrical and Electronics Engineers
[1]	EN 50128		governance	CENELEC
[5]	ISO 9001		governance	International Organization for Standardization
[3]	SUBSET-023	3.0.0	SSRS	UNISIG
[4]	SUBSET-026	3.3.0	SSRS	UNISIG

Table 4. Standards

## 1.5 Definitions and acronyms

Table 5. Definitions and acronyms

Abbreviation	Meaning
CCB	Change Control Board
CI	Configuration Item
CM	Configuration Management
CR	Change Request
ERTMS	European Rail Traffic Management System Train signalling system equipment based on a single Europe-wide standard for train control and command systems.
ERA	European Railway Agency
ETCS	European Train Control System It is a signalling, control and train protection system designed to replace the many incompatible safety systems currently used by European railways
EUPL	European Union Public Licence
EVC	European Vital Control
FM	Formal Methods
GNU	General Public License
IEEE	Institute of Electrical and Electronics Engineers
NA	Not Applicable
NC	Non Conformance
PMP	Project Management Plan
QAP	Quality Assurance Plan

**Table 5 – continued from previous page**

<b>Abbreviation</b>	<b>Meaning</b>
R&D	Research and Development
SCM	Software Configuration Management
SCMP	Software Configuration Management Plan
SIL	Safety Integrity Level
SRS	System Requirements Specification
SW-SIL	Software-Safety Integrity Level (EN 50128:2011)
V&V	Verification and Validation
WP	Work Package

## 2 <sup>35</sup> Configuration Management Tools

In this section the Tools per Configuration Item (CI) and the related repositories will be defined.

### 2.1 Tools

List of Tools to be used

### 2.2 Repositories

- <sup>40</sup> Git is a Web-based revision control hosting service for software development and code sharing (<http://git-scm.com/>). Git is used by large projects to be capable of handling the versioning and incremental development. The Git working directory is a complete repository with the whole history and full revision tracking capabilities. It is compatibility with existing systems/protocols and not dependent on network access or a central server. Git is free software and distributed under the terms of the GNU General Public License. Additional advantages are: broader participation, changes can be reverted, peer-2-peer model: different contributors can work simultaneously and independently (distributed). Extra “features” can added independently of mainline development with re-integration later. Git supports rapid branching and merging, and includes specific tools for visualizing and navigating a nonlinear development history.

## 3 <sup>50</sup> Configuration Identification

- The Configuration Identification makes sure that the items are properly stored for traceability, and procedures for placing configuration items under software configuration management by means of the definition of a hierarchical directory structure will be established and identification scheme for the components to uniquely identify each individual component will be created. A proper configuration identification schema identifies each component of the system and provides traceability between the component and its configuration status information. The Configuration Identification is the basis by where changes to any part of a system are identified, documented, and later tracked through design, development, testing, and final delivery. Configuration Identification incrementally establishes and maintains the definitive current basis for configuration status controlling of a system and its Configuration Items (CIs).

### 3.1 Configuration items identification

In this section the Configuration Items (CIs) have to be selected which together form the product structure. The CIs must be compatible with each other. The CIs have to be defined, their granularity, a numbering system to identify the CIs and a commitment to the Baseline Measurement. These CIs will be selected during the sprints of the scrum methodology and have to be defined in this section.

### 3.2 Configuration item management

Configuration Management (CM) is the process of identifying and defining the configuration items in a system, controlling the release and change of these items throughout the system lifecycle, recording and reporting the status of configuration items. The term configuration item can be applied to anything designated for the application of the elements of configuration management and treated as a single entity in the configuration management system. The entity must be uniquely identified so that it can be distinguished from all other configuration items.

## 4 Configuration Management Organization and Planning

In this section it is planned to guide the Configuration Management program that includes items such as: Conventions, Personnel, Responsibilities and Resources, Training requirements, Supervisory meeting and guidelines. The Configuration Management Organization makes sure that the final delivered software has all of the planned enhancements that are supposed to be included in the release. Configuration Tools, Configuration control thru Audits and Reviews, and the Change Process will be under a different scope of this document.

### 4.1 CM roles assignment

The configuration management roles referenced in this table are CM roles defined in the software configuration management instruction.

### 4.2 Responsibilities and Resources

The configuration management has to define the responsibilities and plan the resources for the project. To prevent confusion about who will perform given Software Configuration Management (SCM) activities or tasks, organizations to be involved in the SCM process need to be clearly identified. Specific responsibilities for given SCM activities or tasks also need to be assigned to organizational entities, either by title or organizational element. The overall authority and reporting channels for SCM should also be identified, although this might be accomplished in the project management or quality assurance planning. Any training requirements necessary for implementing the plans and training new staff members are also specified.

## 5 Configuration Control/Change Management

Configuration change control is a set of processes and approval stages required to change a configuration item's attributes and to re-baseline them. This includes the evaluation of all change requests and change proposals, and their subsequent approval or disapproval. It is the process of controlling modifications to the system's design and documentation.

### 5.1 Change management process and procedure

From the perspective of the implementer of a change, the CI is the "what" of the change. Altering a specific baseline version of a configuration item creates a new version of the same configuration item, itself a baseline. In examining the effect of a change, two of the questions that must be asked are:

1. What configuration items are affected?
2. How have the configuration items been affected?

Configuration items, their versions, and their changes form the basis of any configuration audit.

## 5.2 Change Request

The first step in managing changes to controlled items is determining what changes to make. The software change request process provides formal procedures for submitting and recording change requests, evaluating the potential cost and impact of a proposed change, and accepting, modifying or rejecting the proposed change. Requests for changes to software configuration items may be originated by anyone at any point in the software life cycle and may include a suggested solution and requested priority. One source of change requests is the initiation of corrective action in response to problem reports.

## 5.3 Change Control Board meeting organization

The authority for accepting or rejecting proposed changes rests with an entity typically known as a Change Control Board (CCB). The Change Control Board will be the group which manages change for the project. Change Requests will be reviewed, approved or rejected, incorporated into a new baseline and then delivered only after the Change Control Board has given its approval in accordance with contract requirements.

## 5.4 Change control board reporting

A typical CR form contains a narrative description of the change or problem, information to identify the source of the report, and some basic information to aid in evaluating the report. A CR may be submitted by anyone associated with the project, but usually is submitted by a member of the software development team and during testing and verification phases.

# 6 Configuration Audits

A review of the software with the purpose of assessing the compliance with established performance, requirements, standards, and functions. Configuration audits verify the system and subsystem configuration complies with their functional and physical performance characteristics before acceptance into an architectural baseline.

## 6.1 Responsibilities

## 6.2 Audit content

Audits are conducted according to a well-defined process consisting of various auditor roles and responsibilities. Consequently, each audit must be carefully planned. An audit can require a number of individuals to perform a variety of tasks over a fairly short period of time.

### 135 6.3 Non Conformance follow up and status reporting

All Non Conformances (NCs) identified through the audits are required to be documented in the audit report. The follow up consist of preparing a corrective action plan, listing each of the non-conformances identified.

## 7 Baselines

140 In this section the different Baselines should be scheduled and defined. A Baseline should end in a release. OpenETCS has really concrete Work Packages (WPs) SRS, Model- Code, Safety and Validation and Verification). These WPs are in close communication among each other, but their working schedule has different speed, so the versions of the CI of one WP could not match with the versions of related CIs of other WPs. Due to this, it was decided to define Baselines of each  
145 WP.

### 7.1 Baselines formalism

A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and can be changed only through formal change control procedures. Baselines provide a stable basis for continuing evolution of configuration items.  
150 Baselines are added to the configuration management system as they are developed. Changes to baselines and the release work products built from the configuration management system are systematically controlled and monitored thru the configuration control, change management, and configuration auditing functions of configuration management. OpenETCS team goal is having a complete control of the final developed product and assure the quality of the product, following  
155 the requirements specify. In order to achieve this objective a configuration process has been defined. This process has established 6 different baselines: SRS, Model/Code, Safety, V&V, Product and Archived baselines. The baselines describe the functional and physical attributes of these CIs, in order to maintain control of changes occurring to existing items and new “end items” or deliverables within the projects. The project processes result in establishing approved  
160 baselines and related descriptions in a timely manner. SCMP will set when a baseline will be created. The creation of a baseline will depend on the status of the CIs and its versions.

Definition of each baseline:

- **SRS baseline** will contain the specific version of the components requirements and the supported tools
- 165 • **Model/Code baseline** will be created as soon as it is consider that concrete code and model version could be integrated. This baseline will also contain the supported tools to create the models, code and so on.
- **Safety baseline** will contain the specific version of documentation of the safety items, as well as, the required tools for granting the safety of the project.
- 170 • **V&V baseline** will contain the supported tools for testing, as well as the test cases, use cases, test data, test environment and the whole stuff required to assure the quality of the project and product.
- **Product baseline** will integrate all the different baseline of the WPs to create de tagged release of the product, as well as, any other software or documentation that is needed for the  
175 release of the product.
- **Archived baseline** will contain the back-up of the project.

## 7.2 Baselines diffusion

A Baseline Criteria elaborates the conditions and constraints that must be fulfilled prior to establish/create a baseline. Baseline criteria may vary from baseline to baseline due to the nature of baselines contents, so, specific criteria for each baseline are documented in the Configuration Management Plan.

## 7.3 Baselines schedule

In this Chapter the detailed scheduled Baselines shall be given. All planed baselines in the Configuration Management plan should be reflected as milestones in project schedule in order to avoid them being skipped.

## References

- [1] CENELEC. EN50128 railway applications - communication, signalling and processing systems - software for railway control and protection systems, June 2011.
- [2] Izaskun de la Torre. Quality assurance plan.
- [3] ERTMS. UNISIG subset 023, glossary of terms and abbreviations.
- [4] ERTMS. UNISIG subset 026, system requirements specification - baseline 3, January 2010.
- [5] International Organization for Standardization.
- [6] Bernd Hekele. Contribution guidelines.
- [7] Bernd Hekele. openetcs internal assessment.
- [8] Bernd Hekele. Openetcs ip policy.
- [9] Jonas Helming. Committer election guideline.
- [10] IEEE. IEEE 828 standard for software configuration management plan, March 2012.
- [11] Hardi Hungar Marc Behrens. Openetcs validation & verification plan.
- [12] Stefan Rieger. openETCS publishing guideline.
- [13] openETCS Projekt Team. openETCS full project proposal, 2012.

## APPENDIX A – Definitions and References