OETCS/WP2/?

openETCS

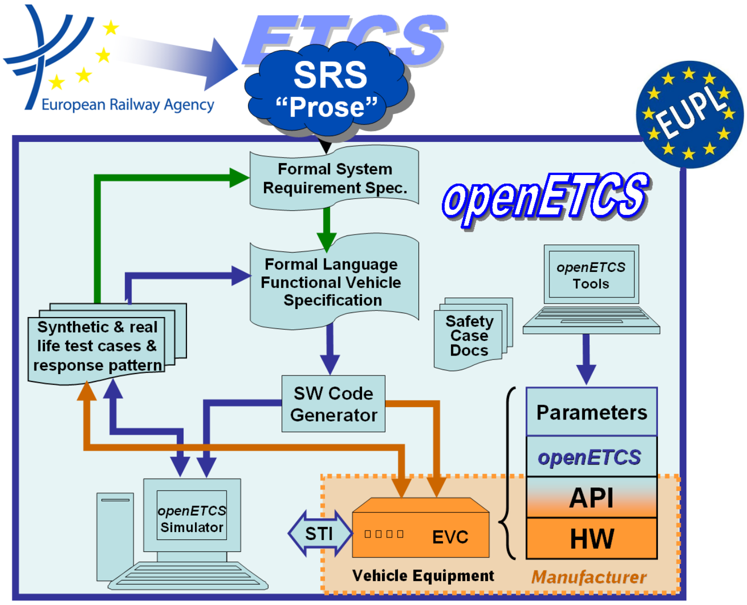
ITEA2 Project
Call 6 11025
2012 – 2015

Work-Package 2

Scrum Process in openETCS Project

Abdelnasir Mohamed and Jan Welte

August 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 2

OETCS/WP2/?
August 2015

Scrum Process in openETCS Project

Document approbation

Lead author:	Technical assessor:	Quality assessor:	Project lead:
location / date	location / date	location / date	location / date
signature Abdelnasir Mohamed (AEbt)	signature ()	signature ()	signature Klaus-Rüdiger Hase (DB Netz)

Abdelnasir Mohamed

AEbt Angewandte Eisenbahntechnik GmbH
Adam-Klein-Str. 26
90429 Nürnberg, Germany
eMail: abdelnasir.mohamed@aebt.de
WebSite: www.aebt.de

Jan Welte

Technische Universität Braunschweig
Institute for Traffic Safety and Automation Engineering
Hermann-Blenk-Str. 42
38108 Braunschweig, Germany
eMail: openetcs@iva.ing.tu-bs.de
WebSite: www.iva.ing.tu-bs.de

Output Document

Prepared for openETCS@ITEA2 Project

Abstract:

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)

Table of Contents

Figures and Tables

Figures

Tables

Document Control

Document information	
Work Package Deliverable ID	
Document title Document version Document authors (org.)	openETCS 01.1 Jan Welte (TU-BS) and Abdelnasir Mohamed (AEbt)

Review information	
Last version reviewed	0.1
Main reviewers (org.)	

Approbation			
	Name	Role	Date
Written by	Jan Welte	WP4-T4.4 Task Leader	March 2015
Approved by	–	–	

Document evolution			
Version	Date	Author(s)	Justification
0.1	18/08/2015	Jan Welte	Document creation
0.2	19/10/2015	Abdelnasir Mohamed	Chapter 1.1
00.1	28/01/2014		

ToDo List

1 Introduction

Nasir

Short introduction to this document.

[AM 1]: Introduction: finish section writing

- extension to QA-Plan and Process Description

1.1 Purpose

The purpose of this document is to define and describe the agile work-flow of the software development process in the openETCS project. As the outcomes of the project, the openETCS software and the openETCS development Tool chain, are precisely defined, the process of how these two artifacts at the beginning of the development process are being produced in detail is not clear. In the conventional software development process, e.g. with a V-Model, the life cycle of the development is implemented in closed non re-definable phases, which they shall be processed consecutively.

Due to the complex nature of the openETCS as a Research and Development (R&D) Project an agile methodology is needed, which it's framework facilitates continuous refinements and improvements throughout the whole openETCS software development process. In this agile way of development complex Tasks are broken down into smaller Tasks (increments), which shall be executed in short time frames called Sprints. This document will be describing the agile framework in openETCS development process applying SCRUM.

1.2 Reference Documents

This document essentially refers to the following standards, ETCS specification documents and openETCS project documents.

- **ISO 9000** — 12/2005 — *Quality management*
- **ISO 9001** — 12/2008 — *Quality management systems — Requirements*
- **ISO 25010** — 03/2011 — *Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models*
- **CENELEC EN 50126-1** — 01/2000 — *Railways applications — The specification and demonstration of Reliability, Availability, Maintenability and Safety (RAMS) — Part 1: Basic requirements and generic process*
- **CENELEC EN 50128** — 10/2011 — *Railway applications – Communication, signalling and processing systems – Software for railway control and protection systems*
- **CENELEC EN 50129** — 05/2003 — *Railway applications — Communication, signalling and processing systems — Safety related electronic systems for signalling*

- **CCS TSI** — *CCS TSI for HS and CR transeuropean rail has been adopted by a Commission Decision 2012/88/EU on the 25th January 2012*
- **SUBSET-026** 3.3.0 — *System Requirement Specification*
- **SUBSET-091** 3.2.0 — *Safety Requirements for the Technical Interoperability of ETCS in Levels 1 & 2*
- **SUBSET-088** 2.3.0 — *ETCS Application Levels 1 & 2 - Safety Analysis*
- **OpenETCS FPP** — *Project Outline Full Project Proposal Annex OpenETCS* – v2.2
- **OpenETCS D2.2** – Report on CENELEC standard
- **OpenETCS D2.3** – Definition of the overall process for the formal description of ETCS and the rail system it works in
- **OpenETCS D2.4** – Definition of the methods used to perform the formal description
- **The Scrum Guide** – The Definitive Guide to Scrum: The Rules of the Game, July 2013

1.3 Glossary

ACedit	Assurance Case Editor
ARM	Argumentation Metamodel
ETCS	European Train Control System
ERA	European Railway Agency
FMEA	Failure Mode Effect Analysis
GSN	Goal Structured Notation
MoRC	Management of Radio Communication
RAMS	Reliability, Availability, Maintainability and Safety
SIL	Safety Integrity Level
SRS	System Requirement Specification
THR	Tolerable Hazard Rate
V&V	Verification & Validation

1.4 Background Information

1.4.1 Agile Development

Jan

really short introduction to the concept of agile development overall

[AM 2]: Agile Development: finish section writing

1.4.2 SCRUM Definition And Roles

This section should be considered as an introduction for the following chapters to fit the content of this document. For more detailed information about SCRUM the *Scrum Guide* is advisable to read. In the SCRUM Guide developed by Ken Schwaber and Jeff Sutherland, Scrum is defined as "*A framework within which people can address complex adaptive problems, while productively* *and creatively delivering products of the highest possible value*". It is "*simple to understand*" but "*difficult to master*".

SCRUM defines three roles; the Product Owner (PO); the SCRUM Master (ScM) and the Development Team. All three roles together make the SCRUM Team.

1.4.3 General SCRUM Framework

Scrum process starts with a Product Backlog (PB), this Product Backlog contains an ordered list of requirements that is maintained for a product. The Product Backlog Items (PBIs) are ordered by the Product Owner based on considerations like risk, business value, date needed, etc. The Product Owner is the responsible of the product backlog and the prioritization of PBIs.

After the defining of the Product Backlog the next task is to create the Sprint planning meeting. This meeting is organized at the beginning of the Sprint cycle. The objective of this meeting is to define the PBIs to be done in following Sprint. The Sprint is the time period in which development occurs on a set of backlog items that the team has committed to (also commonly referred to as iteration).

The result of PBIs selected for implementation in one Sprint is called the Sprint Backlog. The Sprint Backlog is the list of work the Development Team must address during the next Sprint. This list is derived by selecting product backlog items from the top of the product backlog until the Development Team fills it's capacity and make sure it has enough work to during the CompSprint.

Each day during the Sprint, a project team communication meeting occurs. This is called a daily Scrum Meeting and has specific guidelines. The Scrum Meeting is organized by the Scrum Master and the participants respond to three questions:

1. What have you done since yesterday?
2. What are you planning to do today?
3. And Any impediments/stumbling blocks?.

At the end of the Sprint, the result of the PBIs implemented is called the increment (potentially shippable increment), this is the sum of all the Product Backlog items completed during a Sprint and all previous Sprints. The increment must be in a usable condition regardless of whether the Product Owner decides to actually release it.

At the end of Sprint cycle or iteration, two meetings are held; the Sprint review meeting and Sprint retrospective. The approximate duration of the spring review meeting will be no more than 4 hours and it will be moderated by the Scrum Master (ScM). In this meeting two main activities will take place. One is to review the the planned work that has been completed from the one that wasn't. And the second activity is to present to the Product owner the work done. The other meeting after iteration is the spring retrospective. This meeting will be moderated by the scrum master and it's main objective is the continuous process improvements. Two main questions are asked to all participants in this meeting:

- what went well during the Sprint?

- What could be improved in the next Sprint??

2 Agile Development in openETCS

[AM 3]: Agile Dev oETCS: finish section writing

Jan

Basic concept of agile development in openETCS. Allocation of agile work principles to

- allocating agile work activities to the iterative work in openETCS development life cycle (phases nd artifacts)
- describe overlap between EN 50128 roles and Scrum roles

3 openETCS SCRUM Process

In openETCS there are three levels of processes that apply SCRUM:

- The first level is the openETCS Project level, where Project Coordinator is the Product Owner. In this level the SCRUM meetings are organized every Friday.
- The second level is the Work Package level, where Work Package Leader is the Product Owner. In this level SCRUM meeting depends on WP, but at least two SCRUM meetings are organized every week.
- The last level is the Task level, where the Task Leader is the Product Owner. In this level SCRUM meeting period is variable and depends on the task.

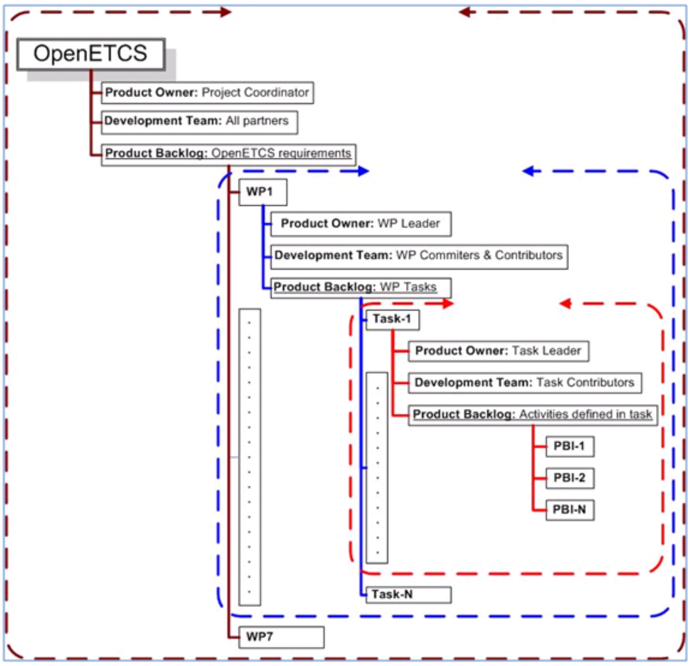

OpenETCS
Product Owner: Project Coordinator
Development Team: All partners
Product Backlog: OpenETCS requirements
WP1
Product Owner: WP Leader
Development Team: WP Commiters & Contributors
Product Backlog: WP Tasks
Task-1
Product Owner: Task Leader
Development Team: Task Contributors
Product Backlog: Activities defined in task
PBI-1
PBI-2
PBI-N
Task-N
WP7

Figure 1. openETCS Project Structure

[AM 4]: finish section writing

3.1 SCRUM Process in Work Packages

- As defined in FPP there are 7 WPs
- WP Roles and it's conventions to conform the EN 50128 Roles
- Grafik of Roles for SW dev. in EN 50128 - Every WP has its own PO
- Webbased CM (GitHub)

Product backlog of the Modeling Team (WP3)

The product backlog is implemented via the issue tracker. The product backlog consists ofUse Cases, marked with the label "US-Operational Scenarios on Utrecht Amsterdam". Each Use Cases has an unique User Story label identifing this Use Case in the backlog. (Sub-) User Stories identified as necessary for the implementation of a Use Cases are marked with the user story label,

The actual work at the user stories is broken down in smaller technical tasks. These tasks are organised in the sprint backlog. Use the label sprint backlog for defining a task. Remark: We will use specific sprint labels when we better start to organise the work in individual sprints.

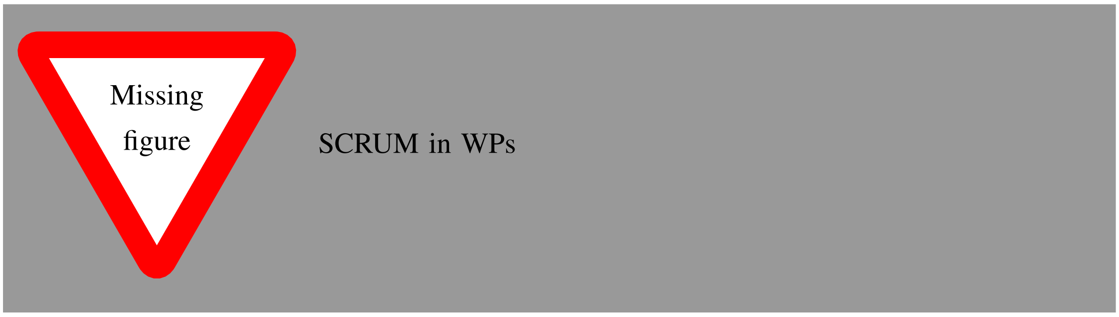
The sprint backlog consists of prioritized tasks. Priorities are integers in the interval [1...1000], with 1000 representing the highest priority. Note that the absolute value is of minor interest, priorization is only needed to classify backlog items by importance.

Priorities are captured in the title of each task respectively issue. Example: US-SpeedMonitoring: Dynamic ASave Calculation Prio: 500 represents a user story with priority 500. The reason to capture the priority in the title is to make priority changes traceable, as the text of github issues may be changed without leaving any trace.

The labels Ready, In Progress, and Deferred are used to capture the state of user stories. Note that closing an issue is equivalent to the state Done.

A user story may only be moved to the state Ready when: A priority has been assigned. The scope is clearly defined and distinguished from other user stories in the states Ready or in Progress. The criteria for reaching the state Done are clear. When setting a user story to "done" a comment is mandatory for documenting the result. Every project member is eligible to propose user stories.

New user stories shall remain in the state (respectively column) Inbox until they have been reviewed by the team in a grooming session.



[AM 5]: SCRUM in WPs GitHub: finish section writing

Here!

3.2 Scrum of Scrums

Nasir

Short introduction to principle of splitting work between different teams, as required by the EN 50128

Principles of grooming work and filling backlogs for distributed Scrum Teams by dynamic work allocation.

Text based on wiki page by Marc Behrens *Objective:* Ongoing verification and validation of openETCS development artifacts are performed within the framework of openETCS scrum process. This page describes the interface of agile verification and validation to the top level customer as well as the scrum process within the verification and validation itself.

[AM 6]: SCRUM of SCRUMS Wiki in WP3 GitHub: finish section writing

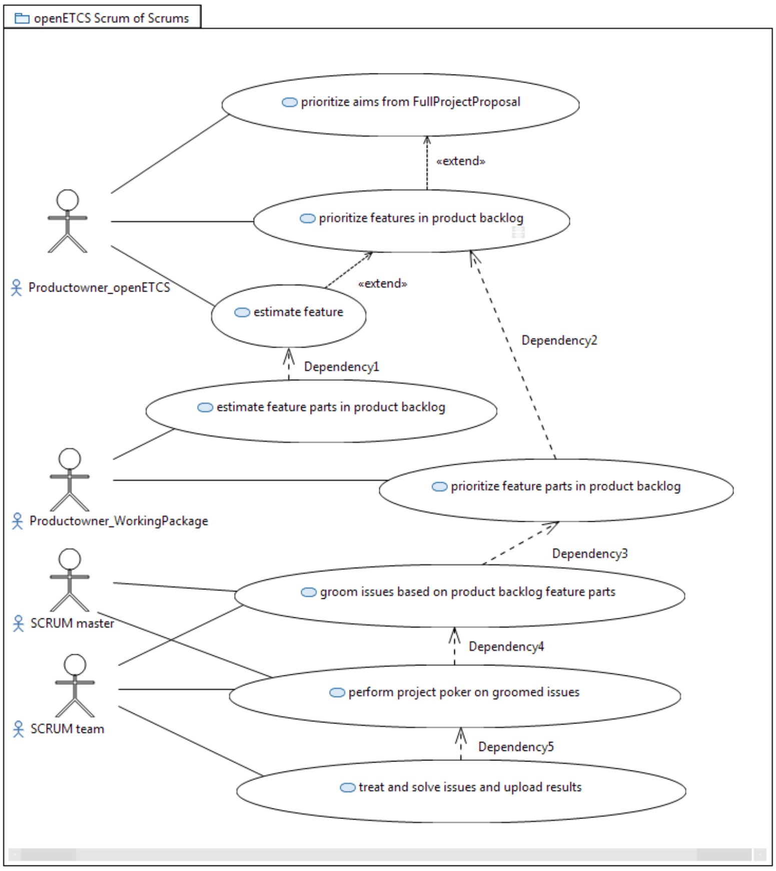

openETCS Scrum of Scrums
prioritize aims from FullProjectProposal
«extend»
prioritize features in product backlog
Productowner_openETCS
«extend»
estimate feature
Dependency2
Dependency1
estimate feature parts in product backlog
prioritize feature parts in product backlog
Productowner_WorkingPackage
Dependency3
groom issues based on product backlog feature parts
SCRUM master
Dependency4
perform project poker on groomed issues
SCRUM team
Dependency5
treat and solve issues and upload results

Figure 2. openETCS scrum of scrums

Within the agile scrum process of openETCS different backlogs are used to synchronise the top level project goals with the tasks performed within the project:

- Each backlog itself consists of items which are represented within the github issue tracker.
- For each backlog a Product Owner is assigned to coordinate and prioritize the items.
- Each Story Point (SP) is estimated to be worth one person day.

The Product Backlog (Product Owner: Klaus-Rüdiger Hase) The Product Backlog consists of top level *Features* by which the product is described.

- Each Feature itself is estimated by the product owner on how many Story Points this Feature is worth.
- A Feature itself is divided into **Feature Parts**.

- A Feature Part is related directly to a working package and the repository the work is documented in.
- Each Feature Part itself is estimated and prioritized by the product owner on how many Story Points this Feature Part is worth.

By estimating the feature the feature becomes accepted on product Level. The Product Backlog is maintained during a regular meeting.

Transferring the Features from the *Product Backlog* to the Verification and Validation Sprint Backlog During regular grooming sessions the scrum team extracts the items with highest priority out of the Product Backlog and refines them to issues within the repository mentioned within the Feature Part. On these issues Project Poker is performed on and by this Story Points are assigned. The decision on which items enter the Sprint Backlog is based on the maximum benefit analysis taking into account the priority as well as the story points.

Sprint Backlog The *Sprint Backlog* consists of issues which are processed during the Sprint. During the Sprint the success is tracked against the Sprint Backlog. The Story Points are earned once the complete result of what has been groomed is uploaded to the github.

4 Release Principals

Jan

principles of artifact releases (specially models) and their verification and validation configuration management of design artifacts, test specifications and test reports change management via issues, status of documents, criteria for releases and assessment

5 Conclusion

can be left open for now

final conclusion of content