SEVENTH FRAMEWORK PROGRAMME

Collaborative Large-scale Integrating Project

OPENCOSS

Open Platform for EvolutioNary Certification Of Safety-critical Systems

Detailed requirements for the process-specific needs of the OPENCOSS platform

D7.2

Work Package:	WP7: Transparent Certification and Compliance-aware Process
Dissemination level:	PU
Status:	Final
Date:	01 November 2012
Responsible partner:	Alessandra Martelli (INT)
Contact information:	Alessandra.martell@intecs.it

PROPRIETARY RIGHTS STATEMENT
This document contains information, which is proprietary to the OPENCOSS Consortium. Neither this document nor the information contained herein shall be used, duplicated or communicated by any means to any third party, in whole or in parts, except with prior written consent of the OPENCOSS consortium.

Contributors

Names	Organisation
Alessandra Martelli	Intecs
Janusz Studzizba	Parasoft
Jerome Lambourg	Adacore
Francisco Ruiz	Tecnalia
Florent Pages	Inspearit
Martijn Klabbers	Technica University of Eindhoven

Document History

Version	Date	Remarks
V0.1	2012-07-10	ToC
V0.2	2012-09-05	New ToC
V0.3	2012-09-24	New ToC and first contributions
V1.0		Approved by PB

TABLE OF CONTENTS

List of Figures

List of Tables

1 Executive Summary

The objective of WP7 is to define a safety certification management infrastructure to support the certification process. This process has to be interwoven with the development and safety assurance processes by allowing developers to assess where they are with respect to their duties to conform to safety practices and standards, and still to motivate them to see the effective progress of the work and level of compliance. In doing so, WP7 will fulfil the OPENCOSS Objectives ST4 (Transparent Certification) and ST5 (Compliance-Aware Development Process).

This document is the second deliverable of WP 7, where the detailed requirements for the process-specific needs of the OPENCOSS platform are elicited. It contains functional and non-functional requirements of the Transparent Certification and Compliance-Aware Development Process functionalities of the OPENCOSS platform. Functional requirements have been captured by means of use cases that have been collected in deliverable D2.3. The non-functional requirements also include quality metrics.

2 Introduction

2.1 Overview

WP7 aims at defining a safety certification management infrastructure to support the certification process. This process will be interwoven with the development and safety assurance processes by allowing developers to assess where they are with respect to their duties to conform to safety practices and standards, and still to motivate them to see the effective progress of the work and level of compliance. WP7 has the following specific objectives:

- Analyse and assess the state of the art and state of the practice in terms of approaches for certification process specification and execution, also by looking at the development and assurance processes as well. Evaluate Agile approaches and continuous integration approaches.
- Develop detailed technical requirements by refining the high level requirements defined in WP2. This includes the identification of business models and constraints such as legal and technological.
- Identify metrics for the certification and safety assurance processes with the pursuit of dependability as a balancing of costs and benefits and a prioritization of risks.
- Design and implement a set of OPENCOSS platform services for certification life-cycle support, standards-compliance awareness, traceability management of certification requirements, and event triggering infrastructure for certification compliance.
- Provide a methodological guide to integrate the OPENCOSS platform services into other existing ALM or tool integration platforms.

Within WP7, T7.1: Transparent Certification and Compliance-aware Process Baseline and process-specific requirements, has the objective to cover the three first bullets above.
This deliverable is the second deliverable of WP7 produced by Task 7.1. Its objectives are to elicit the detailed requirements for the process-specific needs of the OPENCOSS platform; the high level requirements coming from WP2 are refined in this deliverable. It includes the quality metrics.

The process followed to define the low level requirements is the following:

1. Identification of High Level Use Cases/scenarios and High Level Requirements relevant to the Compliance-Aware and Transparent Certification service infrastructures.
2. Definition of Low Level Use Cases of the Compliance-Aware and Transparent Certification service infrastructures.
3. Identification of possible overlaps with WP4 and WP6.
4. Definition of Low Level Requirements of the Compliance-Aware and Transparent Certification service infrastructures

2.2 WP7 vision

WP7 aims at meeting two of the scientific and technological objectives of OPENCOSS. To do so, WP7 is divided in this document into 6 functional areas:

- To meet ST4: Transparent Certification Process
    - Safety Assurance and Process Metrics

  - Safe Product Metrics
  - Safe Process Metrics
- To meet ST5: Compliance-Aware Development Process
  - Mapping of process models
  - Process execution
  - Estimation of compliance

2.2.1 ST4: Transparent Certification Process

The goal here is to expose and make explicit as much as possible the overall certification process to all involved actors. This will help those actors to take immediate actions to increase safety, and reduce the costs of the certification process.

In order to achieve those functions, the 3 functional areas attached to ST4 will collect and use the project information stored in the CCL formalism in the OPENCOSS platform, and calculate the appropriate metrics from them.

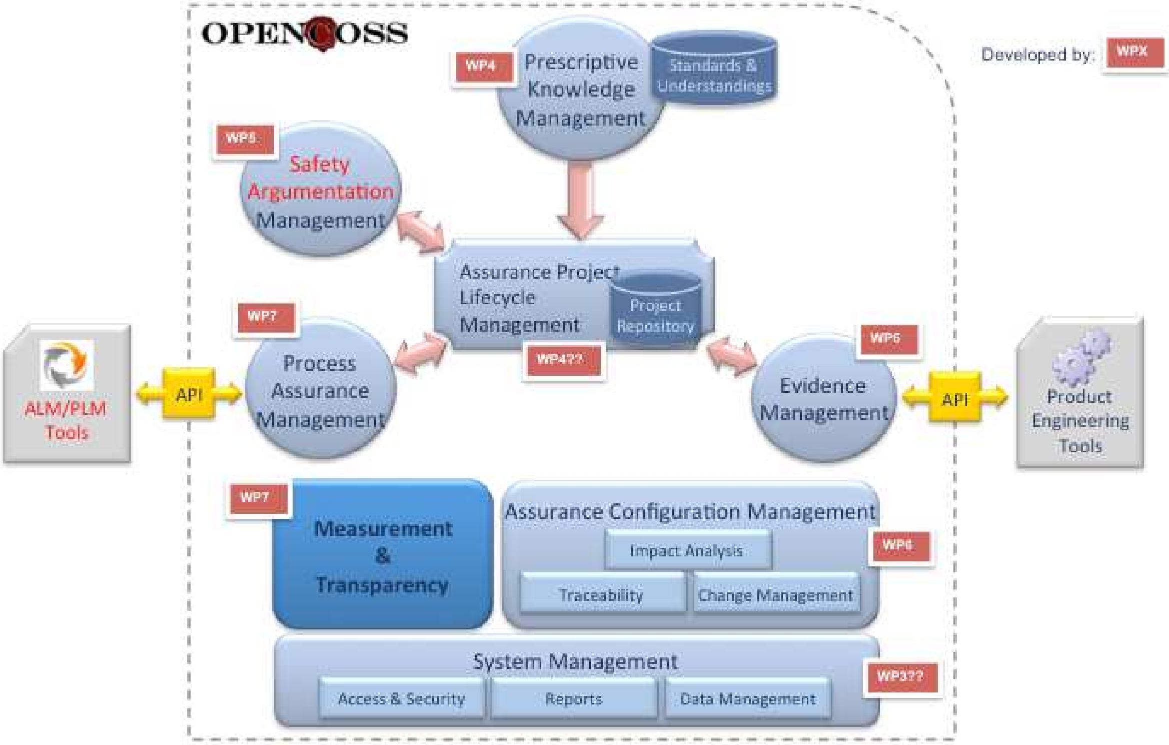
OPENCOSS
WP4
Prescriptive Knowledge Management
Standards & Understandings
Developed by:
WPX
WP5
Safety Argumentation Management
Assurance Project Lifecycle Management
Project Repository
WP4??
WP7
Process Assurance Management
WP6
Evidence Management
ALM/PLM Tools
API
API
Product Engineering Tools
WP7
Measurement & Transparency
Assurance Configuration Management
Impact Analysis
Traceability
Change Management
WP6
System Management
Access & Security
Reports
Data Management
WP3??

Figure 1 Measurement & Transparency in OPENCOSS

2.2.2 ST5: Compliance-Aware Development Process

Concerning the ST5 - Compliance-Aware Development Process: the OPENCOSS platform will rely on external tools to define and execute the process. Such tools typically use BPMN, SPEM or similar process languages to define project processes. The OPENCOSS platform by the mean of WP7 will implement an API that will allow such tool to report to the platform the correct execution of the process. The platform will then be able to compute the conformance of such execution regarding the involved standards.

Following is an overview of the involved mechanisms:

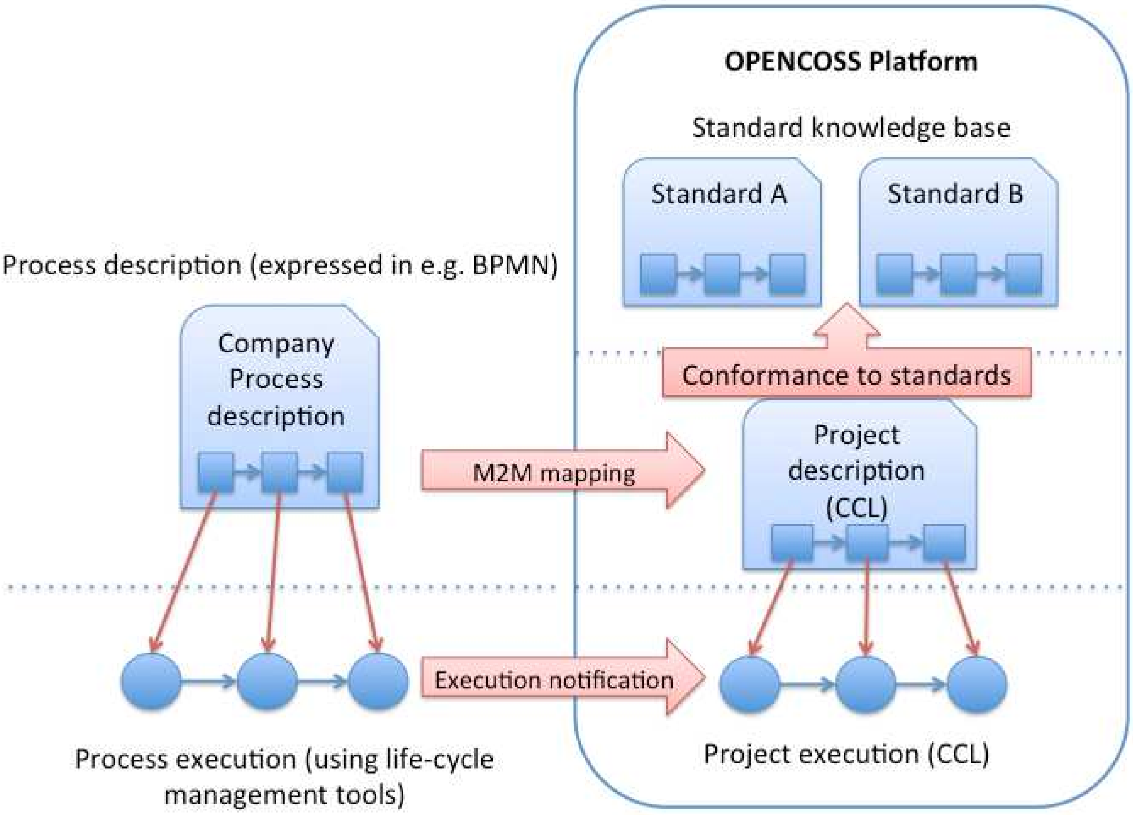
OPENCOSS Platform
Standard knowledge base
Standard A
Standard B
Process description (expressed in e.g. BPMN)
Company Process description
Conformance to standards
M2M mapping
Project description (CCL)
Execution notification
Process execution (using life-cycle management tools)
Project execution (CCL)

Figure 2 WP7 Compliance-Aware Development Process overview

The use of an external tool should be made optional for the OPENCOSS platform user. In case no such tool exists, the platform will fall back to a minimal set of features, which will still help reporting some degree of compliance to the standards, even if not all information is available.

2.3 Structure

This document is organized into two main sections:

- Section 3 provides the background of process modelling in OPENCOSS. In particular, it recalls important points concerning existing process modelling languages, using BPMN as the state-of-the-practice language in process modelling. It will also detail our vision on how process modelling will fit in the OPENCOSS platform.
- Section 4 provides the low level requirements organized into the following subsections:
    - 4.1 Mapping of process models
    - 4.2 Process execution
    - 4.3 Estimation of compliance
    - 4.4 Safety Assurance and Process Metrics
    - 4.5 Safe Product Metrics
    - 4.6 Safe Process Metrics

3 Process modelling

3.1 The BPMN example

Process modelling using the Business Process Modelling Notation (BPMN) has already been discussed in Section 5.3 in OPENCOSS Deliverable D7.1 (Baseline for the process-specific needs of the OPENCOSS platform.) Here we summarize the main concepts of process modelling using BPMN and explain the main attention points in making useful business models and assessing whether they have been followed in a correct way [1].

The core part of the BPMN is formed by the flow objects that represent and describe the core of the workflow, the process model. An A*ctivity* is such a flow object and represents work that is performed within a business process. It can be atomic or non-atomic (compound). The types of Activities that are a part of a Process Model are sub-processes and tasks. A *Sub-Process* is a compound activity that is included within a Process. A *Task* is an atomic activity that is included within a Process.

The BPMN process modelling standard can support different methodologies as well as different modelling goals (e.g., orchestration and choreography). It helps in creating correct business models. However, there are some attention points in creating useful business models.

3.2 Modelling a Business process

Process Modelling is capturing an ordered sequence of business activities and supporting information. Business processes describe how a business pursues its objectives. Capturing is considered successful when the process models consist of flow extended with enough information so that the process can be analysed, simulated, and/or executed. Note, however, that process modelling involves choosing between a great many ways of describing a desired behaviour at an acceptable level of precision. George Box therefore advocated that "All Models are Wrong, Some are Useful" **¡Error! No se encuentra el origen de la referencia.**. The key point here is that: many people assume that there is always a correct model (and that somehow other models are wrong). However, there is seldom only one correct model. Validity of models is closely related to correctness. Models may be invalid in that they incorrectly use a given notation.

The modeller is always making decisions about what to include and what leave out. So one needs to maintain a perspective about the uses of the model and who will interpret it. There are some characteristics of a good model:

- Salient—since no model can represent everything, it must selectively represent those things that are most relevant to the task at hand.
- Accurate—the model should precisely encode the actual state of affairs and not an erroneous or biased view.
- Complete yet Parsimonious—the model should be as simple as possible, but no simpler.
- Understandable—once we perceive the model we must be able to make sense of it; it shouldn't be too complicated or unfamiliar for us to understand.

Summarizing, in order to be useful, all models selectively represent some elements of the real world. The modeller consciously excludes different dimensions of the domain that are irrelevant for the intended use of the model (in order to achieve the modelling goals).

Often, the prescribed business model prevents the business from making mistakes. Reality, however, is far more complex than the ideal business model, and exceptions to the workflow of the model occur frequently in practice. The model must often support these kinds of exceptions.

As a result, assessing whether a business model is correctly executed in practice often requires analysis of more than the correctness of the business model alone. Log file information, systems' output files, and audit trails can provide the input for the discovery or absence of harmful patterns [2].

3.3 Traceability and Dependency

Traceability is the degree to which a relationship can be established between two or more products of the development process, especially products having a predecessor-successor or master-subordinate relationship on another [5]. The importance of traceability is well understood in the software engineering community and adopted across numerous software development standards [3]. Industries are often compelled to implement traceability practices by government regulations. For example, the U.S. Federal Aviation Administration (FAA) that states that software developers need to have ways of demonstrating traceability between design and requirements.

Traceability supports numerous critical activities. For example, pre-requirements traceability is used to demonstrate that a product meets the stakeholders' stated requirements, or that it complies with a set of government regulations. Traceability is also used to establish and understand the relationships between requirements and downstream work products such as design documents, source code, and test cases.

Similar to traceability are the dependency links. While traceability shows how a given artefact has been derived from a higher level artefact, dependency will show the more general constraints that apply to this artefact. As an example, a piece of code will be traceable to a low level requirement (LLR), as there is a specific activity that corresponds to the implementation of the LLR. This same piece of code will also depend on other artefacts such as the coding standard.

In this context, Traceability and Dependency links support tasks such as impact analysis, which helps developers understand how a proposed change impacts the current system, and code verification, which identifies superfluous and unwanted features by tracing all elements of the source code back to specific requirements. Traceability can also support reuse of parts of a software system by identifying the parts that match (new) requirements, and the evolution of software systems.

In practice, traceability links are typically created and maintained either through the use of a requirements management tool, or else in a spread sheet or Word document directly. However, there are numerous issues that make it difficult to achieve successful traceability in practice. These issues include social ones related to communication between project stakeholders, as well as technical issues related to physically creating, maintaining, and using thousands of interrelated and relatively brittle traceability links. As a result, many organisations struggle to implement and maintain traceability links, even though it is broadly recognised as a critical element of the software development life cycle. This is one of the main issues faced by safety projects, as without a proper traceability, impacts of changes are very hard to determine, which leads to a natural 'freeze' of the implementation, even when bugs are found. This has then a potential direct impact on safety.

There are some direct connections between process descriptions and traceability. In fact, activities requiring some work product (WP) as input, and producing an output WP follow in fact a traceability link from the input WP to the output WP. Similarly, a change in a work product used as input for an activity impacts the activity itself, that should be re-run to verify if the produced output is still valid.

3.4 Process modelling in OPENCOSS

The OPENCOSS process modelling formalism will be derived from the CCL formalism, defined by the Work Package 4. This will allow a powerful integration with the other components of the OPENCOSS platform, and will enable an at least partial automated verification of the conformance of a process execution regarding mandatory constraints imposed by safety standards. Using such a common language throughout the platform will allow more powerful checks regarding safety constraints, as well as more accurate metrics to expose progress and conformance measures to the different stakeholders (developers, project and safety managers, reviewers, assessors, etc.).

This however will have the drawback of not using widely used modelling languages such as BPMN, described above. The specific formalism used by the CCL is not available at the moment this document is written, so a precise description of the final semantics of this language is not possible yet.
However, we will try to keep the Common Certification Language as compatible as possible with languages such as BPMN, so that a Model-to-Model translation will remain possible, and keep as much as possible the semantic of the original model.
Such Model to Model translation would still make possible the interaction of external tools such as life-cycle management tools with the OPENCOSS platform, and will allow the user to monitor the execution of its process in its known environment.

Our vision of such interaction is described already in chapter 2.2. The detailed low level requirements that concern this interaction are described in the next chapters **¡Error! No se encuentra el origen de la referencia.**, 4.2 and 4.3.

4 Low Level Requirements

4.1 Mapping of process models

4.1.1 Express user’s processes into the CCL formalism

WP7-01-01	
Description	The OPENCOSS platform should provide means to express user’s processes into the CCL formalism.
Rationale	Each user organization has its own specific formalism and “language” to describe its internal processes. In order to be able to integrate it in the OPENCOSS Platform and to make it “understandable” and “workable” for the platform, the first step is to “traduce” the user’s processes in the OPENCOSS Platform language (“CCL”).
Stakeholders	Project Manager, OPENCOSS Consortium, safety assessor, Quality Manager, Certification authorities
Status	Proposed
Priority	H

4.1.2 Map “CCLized” user’s processes with relevant standard(s)

WP7-01-02	
Description	The OPENCOSS platform should provide means to map “CCLized” user’s processes (e.g. expressed in the CCL formalism, see WP-01-01) with the relevant standard(s) (also expressed in the CCL formalism).
Rationale	The certification or compliance assessment process is performed, tracked and reported regarding the standards requirements. So to be able to assess a user’s process it is necessary to get a mapping between this process and the relevant standard.
Stakeholders	Project Manager, OPENCOSS Consortium, safety assessor, Quality Manager, Certification authorities
Status	Proposed
Priority	H

4.1.3 Keep traceability between user’s processes and the relevant standard

WP7-01-03	
Description	The OPENCOSS platform should provide means to keep traceability between user’s processes (expressed in its initial formalism or native language) and the relevant standard.
Rationale	At the end, the user’s organization and certification bodies needs are to be able to get a direct correspondence between the user’s process to be assessed (in its initial formalism) and the target standard (in its proper formalism). This traceability should be maintained throughout the project lifecycle.
Stakeholders	Project Manager, Quality manager, OPENCOSS consortium
Status	Proposed
Priority	H

4.1.4 Provide and maintain all standards of the OPENCOSS Platform expressed in the CCL formalism

WP7-01-04	
Description	The OPENCOSS platform should provide and maintain (e.g. standards evolutions and new versions) all standards – which are in the scope of the OPENCOSS Platform – expressed in the CCL formalism ("CCLized").
Rationale	To be able to map a CCLized user's process with the relevant standard, this last one has to be expressed/translated in CCL language. This translation must be validated beforehand and maintained throughout the lifecycle of the different standards.
Stakeholders	OPENCOSS Consortium, Certification authorities
Status	Proposed
Priority	H

4.2 Process execution

4.2.1 Provide a process-viewer

WP7-02-01	
Description	OPENCOSS platform should provide a process-viewer which visualizes the status of each process item
Rationale	Regardless the fact whether the process is executed automatically in external process execution tool or not executed automatically by any tool, users should be able to see their process basic data: • process actions status - not-done/in-progress/done • who and when performed specific action, • traceability from process actions to work products produced by them
Stakeholders	Project Team, External Safety Assessor
Status	Proposal
Priority	M

4.2.2 Provide API to automatically update the status of process execution

WP7-02-02	
Description	OPENCOSS platform should provide API to automatically update the status of execution of the process actions
Rationale	This functionality supports the use case when external tool executes the process. In such case information regarding each action (activity) should be sent to OPENCOSS via a dedicated API, and should contain process-specific information, such as: • status (not-done/in-progress/done) • who and when performed the action • traceability to work products produced by the action
Stakeholders	Process Execution Tool
Status	Proposal
Priority	M

4.2.3 Provide means to manually update the status of process execution

WP7-02-03	
Description	OPENCOSS platform should provide means to manually update the status of execution of the process actions
Rationale	This functionality supports the use case when there is no process execution external tool. In such case OPENCOSS platform should let the user manually enter the process actions status information in OPENCOSS.
Stakeholders	Project Team, External Safety Assessor
Status	Proposal
Priority	M

4.2.4 Deduce basic process information from work products

WP7-02-04	
Description	OPENCOSS platform should be able to deduce basic process information from work products
Rationale	OPENCOSS platform should be able to deduce process information (who and when performed specific action) from the work product information created by this action. Note: the work product information is sent to OPENCOSS platform via API described by D6.2 requirements.
Stakeholders	Project Team, External Safety Assessor
Status	Proposal
Priority	M

4.3 Estimation of compliance

4.3.1 List Safety Assessment Items

WP7-03-01	
Description	OPENCOSS Platform shall list project process-related safety assessment items
Rationale	There should be available to project team a list of project artefacts (tasks/activities) that are related with safety management. There project artefacts are key for managing the safety compliance.
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.2 List Safety Requirements/Objectives

WP7-03-02	
Description	OPENCOSS Platform shall list required requirements and objectives
Rationale	A set of standard requirements or objectives are collected from Prescriptive Knowledge Management. The set of requirements/objectives that apply to current project process are listed to project management team.

Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.3 View details of project artefacts

WP7-03-03	
Description	OPENCOSS Platform shall show details of project artefacts
Rationale	OPENCOSS Platform shall made available all details of project process items, including all process record data: who, when and how performed what action or produced the product artefacts (requirements, design, code, tests, reviews)
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.4 Show status of compliance

WP7-03-04	
Description	OPENCOSS Platform shall display graphical/textual status of compliance
Rationale	The status of compliance can be displayed as non-compliant, partially-compliant or fully-compliant. There should be some kind of graphical item showing project artefacts in this status. This shall be displayed both in project artefact listing and in details.
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.5 Non-compliant status report

WP7-03-05	
Description	OPENCOSS Platform shall display a textual report for non-compliant project artefacts.
Rationale	The information about non-compliance items shall provide hints for turning the non-compliance into a compliant-item. This should include both targeted objectives/requirements, a textual description about the non-compliance, and sources or hints to make project artefact compliant.
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.6 Compliance status report

WP7-03-06	
Description	OPENCOSS Platform shall display a graphically an estimation on compliance
Rationale	OPENCOSS Platform shall collect all information of project artefacts (and associated compliance items) and provide an up-to-date compliance report, stating what project artefacts need to be completed, reviewed, or amended
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.7 Compliance means transparency

WP7-03-07	
Description	OPENCOSS Platform shall show compliance means for process-related artefacts
Rationale	OPENCOSS Platform shall reflect the agreements, interpretation and compliance means agreed with Independent Assessor, associated to each project artefact
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.8 Availability of project artefacts

WP7-03-08	
Description	OPENCOSS Platform shall made available safety compliance-related project artefacts
Rationale	Project artefacts shall be made available for people assessing compliance of the project. This shall include read permission for Independent Assessor or Certification Authorities. The set of project artefacts is collected on a Safety Standard Compliance Estimation Report
Stakeholders	Independent Safety Assessor, Internal Safety Assessor
Status	Proposed
Priority	M

4.3.9 Annotation on project artefacts

WP7-03-09	
Description	OPENCOSS Platform shall allow to associate annotations to project artefacts
Rationale	During compliance assessment, assessors could provide annotations over project artefacts. These annotations shall be stored in OPENCOSS platform to later usage for the compliance assessment decision
Stakeholders	Independent Safety Assessor, Internal Safety Assessor
Status	Proposed
Priority	M

4.3.10Highlight compliance items to be audited

WP7-03-10	
Description	OPENCOSS shall provide means to highlight compliance items to be audited
Rationale	Audits are the main tool for validating that a given process execution is aligned to a given target. As in the project, the accounting responsibilities (RACI matrix) are established beforehand, once the process item is executed, the OPENCOSS Platform should inform about the availability of this item to be reviewed.
Stakeholders	Project Team
Status	Proposed
Priority	M

4.3.11Request to upload missing process items

WP7-03-11	
Description	OPENCOSS shall provide means to request to upload missing process items
Rationale	The OPENCOSS platform should provide some kind of formal requests to collect all missing information required for certification dossier.

Stakeholders	Project Manager, Safety Assessor, QM
Status	Proposed
Priority	M

4.3.12Audit of process-specific items

WP7-03-12	
Description	OPENCOSS shall provide means to audit process-specific items
Rationale	Before presenting any information to the certification authorities, QA people should be able to review and audit the completeness of process-related items. This requirements is met when the “auditing” responsible is able to access to the documentation
Stakeholders	Project Manager, QM
Status	Proposed
Priority	M

4.3.13Feedback on certification estimation report

WP7-03-13	
Description	OPENCOSS shall provide means to provide a feedback on certification estimation report
Rationale	The certification dossier, before being submitted to the certification authorities, the QM should be able to provide in the OPENCOSS platform a feedback about the status of the certification dossier, at any stage of the process of collecting information. The result should be in the range of complete, partially complete or incomplete.
Stakeholders	Project Manager, QM
Status	Proposed
Priority	M

4.3.14Editor for creating the assessment report

WP7-03-14	
Description	OPENCOSS shall provide an editor for creating the assessment report
Rationale	OPENCOSS Platform shall provide an editor (text, checklists, and questionnaire) to support the filling of the assessment report. This editor should support basic operations (copy & paste, import, export, editing, deleting, etc.).
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

*4.3.15*Publish the certification assessment report

WP7-03-15	
Description	OPENCOSS shall provide means to publish the certification assessment report
Rationale	The OPENCOSS Platform shall include an action for publishing the assessment report. The assessment report could be published in typical formats, including MS Word, MS Excel, and Webpage or in a view of OPENCOSS Platform.
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

*4.3.16*Make available the certification assessment report

WP7-03-16	
Description	OPENCOSS shall provide means to make available the certification assessment report
Rationale	The success or failure in certification result should be informed to all stakeholders in the project. Once the assessment report is published, the member of the project team should have access to the result of this assessment report.
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4 Safety Assurance and Certification Process Metrics

4.4.1 Time efficiency metrics

WP7-04-01	
Description	OPENCOSS shall allow setting and monitoring of time efficiency metrics for the assurance/certification process.
Rationale	OPENCOSS shall allow to set time plans (calendar working days) for the assurance/certification activities, to monitor progress, set new forecast, and to evaluate actuals against plans. The preferred indicator shall be the SPI “Schedule Performance Indicator”. Goal: plan and control the certification process Question: to what extent we are meeting our deadlines ?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.2 Resource efficiency metrics

WP7-04-02	
Description	OPENCOSS shall allow setting and monitoring of resource efficiency metrics for the assurance/certification process
Rationale	OPENCOSS shall allow setting resources plans (man/hours) for the assurance/certification activities, to monitor progress, set new forecast, and to evaluate actuals against plans. The preferred indicator shall be the CPI “Cost Performance Indicator”. The assurance/certification process is mainly a human activity (occasionally supported by software tools). Hence man/hours do represent the most significant metrics for resource usage. Goal: plan and control the certification process Question: to what extent we are meeting our effort budget?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.3 Support metrics estimation

WP7-04-03	
Description	OPENCOSS shall support the estimation of support metrics for the assurance/certification process
Rationale	OPENCOSS shall support the estimation of schedule and resources of a new assurance/certification process executed in a particular project as the result of a parametric model and/or statistical historical data. Goal: plan and monitor the certification process Question: what will be the schedules and efforts for this new certification project ?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.4 Completeness metrics estimation

WP7-04-04	
Description	OPENCOSS shall support the estimation of completeness (effectiveness) metrics for the assurance/certification process
Rationale	OPENCOSS shall support completeness by collection of metrics on safety problems and safety problems removal efficiency, organized by risks levels. Goal: improve the certification process Question: to what extent are we effective in finding safety related problems?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.5 Accuracy metrics estimation

WP7-04-05	
Description	OPENCOSS shall support the estimation of the accuracy (effectiveness) metrics for the assurance/certification process
Rationale	OPENCOSS shall support completeness by collection of metrics on safety problems and safety problems removal efficiency, organized by risks levels. Goal: improve the certification process Question: to what extent are we effective in finding safety related problems ?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.6 Assurance/certification metrics presentation

WP7-04-06	
Description	OPENCOSS shall support the estimation of the assurance/certification presentation metrics for the assurance/certification process
Rationale	OPENCOSS shall support presentation of the assurance/certification process metrics as a dashboard providing all essentials indicators and highlighting (e.g. red colour) values that needs attention. This includes historical data permitting trend analysis. Goal: plan and control the certification process Question: what is the current status of the certification process ?
Stakeholders	Safety Assessor, Certification Authorities

Status	Proposed
Priority	M

4.4.7 Compliance coverage metrics estimation

WP7-04-07	
Description	OPENCOSS shall support the estimation of the compliance coverage metrics
Rationale	OPENCOSS Platform shall support progress indicators that will provide informative data about the status of the compliance to selected standard. The metrics is a ratio of completed and compliant items/requirements against overall expected items. It would be desirable that items are weighted in terms of required effort. Goal: plan and control the certification process Question: how much progress we have?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.8 Certification costs collection

WP7-04-08	
Description	OPENCOSS shall support the collection of certification costs as a ratio over the total engineering costs.
Rationale	One of the targets for OPENCOSS is to demonstrate an improvement in certification costs. The definition of a measurement associated to the cost will show the effectiveness of the OPENCOSS approach. The ration of certification costs over overall engineering costs provides a sound basis to measure improvement. Goal: improve the certification process Question: how efficient is my certification process?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.9 Claims coverage metric estimation

WP7-04-09	
Description	OPENCOSS shall support the estimation of the claims coverage metric
Rationale	OPENCOSS Platform shall support the collection of data about claims demonstrated and ratio over all claims. Goal: plan and control the certification process Question: how much progress we have?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.4.10Safety goals coverage metric estimation

WP7-04-10	
Description	OPENCOSS shall support the estimation of the safety goals coverage metric
Rationale	OPENCOSS Platform shall support the collection of data about safety goals satisfied and ratio over all safety goals. Goal: plan and control the certification process Question: how much progress we have?
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.5 Safe Product Metrics

4.5.1 Software Complexity metrics

WP7-05-01	
Description	OPENCOSS shall support the collection and presentation of the software Complexity metrics
Rationale	All safety standards have strong requirements on the level of complexity
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.5.2 PDF/PFH metrics

WP7-05-02	
Description	OPENCOSS shall support the collection and presentation of the PDF/PFH metrics
Rationale	Probability of Dangerous Failure on Demand (PFD) or Probability of Dangerous Failure per Hour (PFH)
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.5.3 Product architecture metrics

WP7-05-03	
Description	OPENCOSS shall support collection and presentation of product architecture metrics
Rationale	OPENCOSS shall support collection and presentation of the following product architecture metrics: • Safe Failure Fraction metrics • Single Point Fault metrics • Latent Fault metrics
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.5.4 Product architecture metrics presentation

WP7-05-04	
Description	OPENCOSS shall support presentation of the product architecture metrics as a dashboard
Rationale	OPENCOSS shall support presentation of the product architecture metrics in the previous requirement as a dashboard providing all essentials indicators and highlighting (e.g. red colour) values that needs attention. This includes historical data permitting trend analysis.
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.6 Safe Process Metrics

4.6.1 Process metrics

WP7-06-01	
Description	OPENCOSS shall support the collection and presentation of process metrics
Rationale	OPENCOSS shall support the collection and presentation of the following process metrics: • Software test coverage (all safety standards have a strong requirements on the depth of testing) • Safety related defects found by different V&V activities and their trends analysis
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

4.6.2 Process metrics presentation

WP7-06-02	
Description	OPENCOSS shall support presentation of the process metrics as a dashboard
Rationale	OPENCOSS shall support presentation of the process metrics in the previous requirement as a dashboard providing all essentials indicators and highlighting (e.g. red colour) values that needs attention. This includes historical data permitting trend analysis.
Stakeholders	Safety Assessor, Certification Authorities
Status	Proposed
Priority	M

5 Conclusions

In this document it was analysed the set of High Level Requirements that apply to transparent certification and compliance management. Among these, we have reviewed the approach for ensuring that compliance items correspond to those applied to the project process applying prescriptive knowledge for assessment of safety compliance.

There were five key areas in which the focus of WP7 has been divided:

- Mapping of process models.
- Estimation of process compliance.
- Safety Assurance and Compliance Metrics
- Safety Process Metrics
- Safety Product Metrics.

Mapping of process models is responsible for creating a link between business processes into the prescriptive standard. This link associates both activities/tasks and work product of business models into CCL items modelling standards. This information shall be used to link project execution to compliance items.

Estimation of process compliance focus on the management of certification items evolution during the project execution, and how compliant is the project process. Here, a liaison with WP6 is required, as many proofs of compliance are stored as evidences.

Safety Assurance metrics and compliance metrics perform estimations over the overall project process to get information about the evolution in time of the process-associated compliance items, and how these items are contributing to the safety assessment.

Safety process metrics and safety product metrics are objective measurements over the compliance items to check if required metrics coming from standards are met.

Low level requirements for these areas were obtained according the context for assessment, by following the approach stated in D2.3. In this deliverable, we have helped in the definition of the use cases that are associated to WP7, mainly Process Assurance Management.

The context for extracting the WP7 LLR is depicted in the Fig. 4. In this context, we are taking into account both external information (business process and project processes) and the interaction between OPENCOSS Platform internals. Business processes are mapped into prescriptive standards. This information is used for describing both Project Lifecycle Management (WP4 context) and the creation of required assessment items for a project process. The linkage of project actions (e.g. execution of a given task or activity) and conformance with compliance items is managed at this level, so that, the data collection is made jointly with WP6.

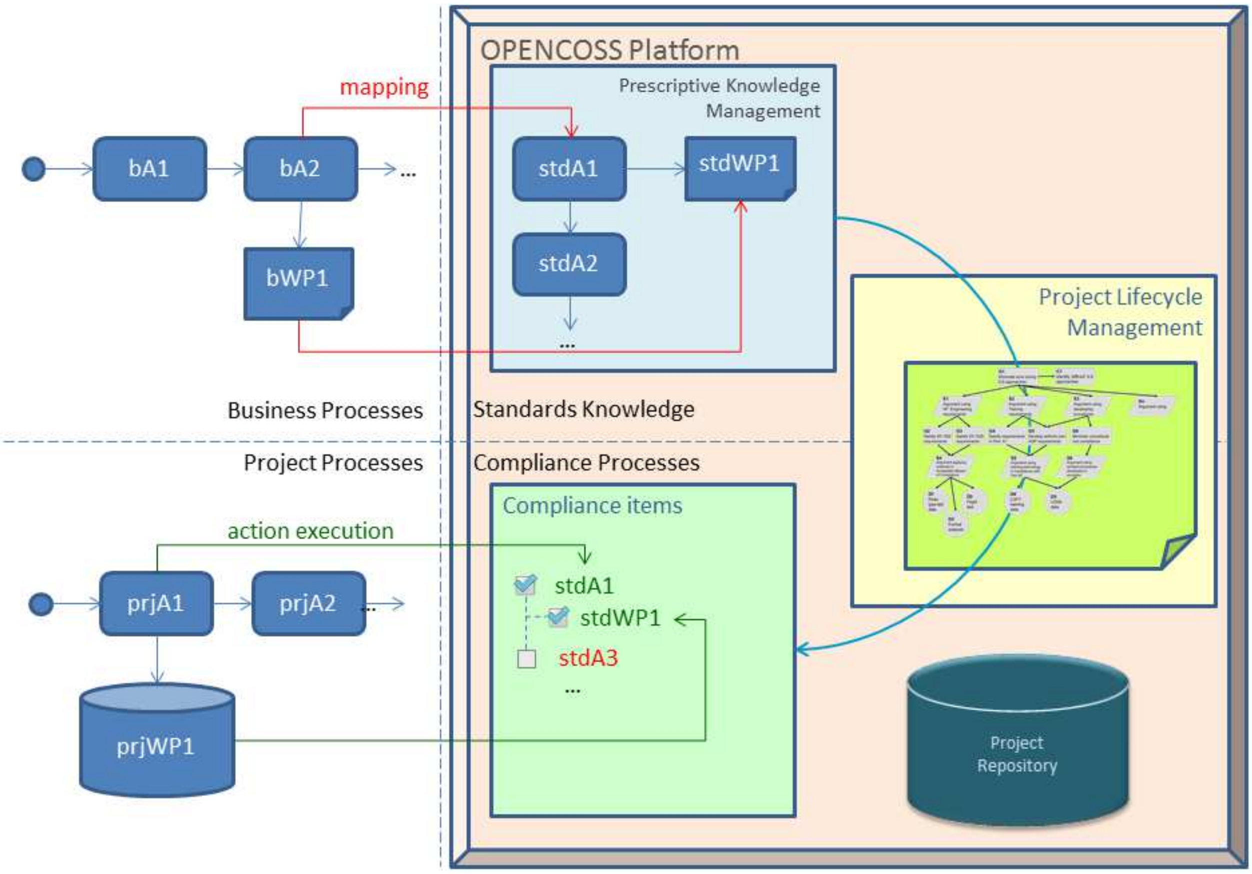

Figure 3. Process Assurance Management context

6 Abbreviations and Definitions

ARTEMIS	Advanced Research & Technology for EMbedded Intelligence and Systems
WP	Work Package
DoW	Description of Work
CCL	Common Certification Language
BPMN	Business process Management Notation
PMOD	Process MODelling
SPEM	Software Process Engineering Modelling
QM	The Qualifying Machine

7 References

[1] White, S.A., Miers, D., 2008, BPMN modeling and reference guide. Future Strategies Inc., Lighthouse Point, Florida, USA

[2] Aalst, W.M.P. van der, Hee, K. van, 2000, Workflow Management, Models, Methods and Systems, Eindhoven University of Technology.

[3] Cleland-Huang, J., Gotel, O., Zisman, A., 2012, Software and Systems Traceability, Editors, Springer Verlag, London.

[4] Box, George E. P.; Norman R. Draper, 1987,. Empirical Model-Building and Response Surfaces, Wiley.

[5] IEEE Computer Society Software Engineering Standards Committee, 1990, IEEE Standard Glossary of Software Engineering Terminology, IEEE Std 610.12-1990,