Collaborative Large-scale Integrating Project

Open Platform for EvolutioNary Certification Of Safety-critical Systems

Detailed requirements for evidence management of the OPENCOSS platform D6.2

Work Package:	WP6: Evolutionary Evidential Chain
Dissemination level:	Public
Status:	Final
Date:	1 November 2012
Responsible partner:	Jose Luis de la Vara (Simula Research Laboratory)
Contact information:	jdelavara@simula.no

Contributors

Names	Organisation
Fabien Belmonte and Laurent Pitot de la Beaujardiere	ALSTOM Transport
Giorgio Tagliaferri and Vincenzo Manni	RINA Services SpA
Jerome Lambourg	AdaCore
Alexander Serebrenik, Martijn Klabbers, and Luc Engelen	Eindhoven University of Technology
Janusz Studzizba	Parasoft SA
Jose Luis de la Vara, Sunil Nair, and Mehrdad Sabetzadeh	Simula Research Laboratory
Karel van de Meent	Inspearit (before DNV ITGS)
Alberto Melzi	Centro Ricerche Fiat S.C.p.A.

Document History

Version	Date	Remarks
V0.1	2012-04-18	First ToC
V0.2	2012-5-15	ToC update and initial content
V0.3	2012-10-5	ToC and content update
V0.4	2012-10-26	Content update
V0.5	2012-10-31	Requirements specification completion
V1.0	2012-11-1	Deliverable finalization

TABLE OF CONTENTS

List of Figures

List of Tables

Abbreviations

BPD	Business Process Diagram
BPMN	Business Process Model and Notation
CCL	Common Certification Language
CENELEC	Comité Européen de Normalisation Electrotechnique (European Committee for Electrotechnical Standardization)
GSN	Goal Structuring Notation
HTML	HyperText Markup Language
IEEE	Institute of Electrical and Electronics Engineers
IS	Information System
ISO	International Organization for Standardization
MS	Microsoft
OMG	Object Management Group
PDF	Portable Document Format
RAM	Requirements Abstraction Model
RE	Requirements Engineering
RTF	Rich Text Format
R&D	Research and Development
SCMS	Software Certification Management System
SPEM	Software & Systems Process Engineering Metamodel Specification
UML	Unified Modeling Language
V&V	Verification and Validation
WP	Work Package

Executive Summary

This document (D6.2) is the second deliverable of OPENCOSS WP6. This WP aims to define a safety certification management infrastructure for an evolutionary evidential chain. The overall goal of D6.2 is to set the context for and document the detailed requirements for evidence management of the OPENCOSS platform.

Different inputs and aspects have been taken into account for discovery of detailed requirements for evidence management, aiming to complement and complete the work performed for OPENCOSS deliverable D6.1 (Baseline for the evidence management needs of the OPENCOSS platform). Situations in which evidence evolves have been determined. How evidence management relates to OPENCOSS CCL, the OPENCOSS platform shall manage evidence and artefacts, and tools for evidence management have also been analysed. A survey on the state of the practice concerning evidence management has been conducted, and other OPENCOSS deliverables have been reviewed in depth. In addition, the overall, high-level business process for evidence management in assurance projects has been analysed, as well as the goals, expected features, and use cases related to evidence management for the OPENCOSS platform.

A set of 111 detailed requirements has been specified and divided into five functional areas for evidence management of the OPENCOSS platform. *Evidence storage* is concerned with the determination, specification, and structuring of the evidence items of an assurance project. *Evidence traceability* is concerned with the specification and adequate maintenance of traceability between evidence items of an assurance project. *Evidence evaluation* is concerned with the assessment of the completeness and adequacy of the body of evidence of an assurance project. *Evidence change impact analysis* is concerned with the identification and analysis of possible effects resulting from changes in the body of evidence of an assurance project. Finally, *Integration with external tools* is concerned with the possibility of importing and exporting information from and to external tools. In addition, several user interface mock-ups have been created to show how the OPENCOSS platform might display the information related to evidence management of an assurance project and facilitate users' work.

The results presented in this deliverable will serve as basis for the next WP6 task (T6.2 - Design of the evidence management service infrastructure). In addition, D6.2 will be used as input for definition of the OPENCOSS CCL, specification of integration requirements, and specification of test cases. In relation to the work to perform in WP6, the two main aspects that will have to be studied during the design task are (1) evidence traceability and (2) evidence change impact analysis. It is necessary to provide practitioners with new means that help them deal with these activities in an effective and more efficient way. It must also be further analysed how the conceptual frameworks developed in WP4 and WP5 affect evidence management in the OPENCOSS platform, once these frameworks are finished. Another important challenge to address for evidence management is to determine how to integrate the OPENCOSS platform with the external with which evidence (i.e., information used as evidence) can be exchanged.

1 Introduction

1.1 Scope and Purpose

Safety assurance and certification are amongst the most expensive and time-consuming tasks in the development of safety-critical embedded systems. European innovation and productivity in this market is curtailed by the lack of affordable (re)certification approaches. Major problems arise when evolutions to a system entail reconstruction of the entire body of certification arguments and evidence. Further, market trends strongly suggest that many future embedded systems will be comprised of heterogeneous, dynamic coalitions of systems of systems. As such, they will have to be built and assessed according to numerous standards and regulations. Current certification practices will be prohibitively costly to apply to this kind of embedded systems.

The OPENCOSS project aims to devise a common certification framework that spans different vertical markets for railway, avionics and automotive industries, and to establish an open-source safety certification infrastructure (hereafter referred to as OPENCOSS platform). The infrastructure is being realised as a tightly integrated solution, supporting interoperability with existing development and assurance tools. The ultimate goal of the project is to bring about substantial reductions in recurring safety certification costs, and at the same time increase product safety through the introduction of more systematic certification practices. Both will boost innovation and system upgrades considerably.

OPENCOSS WP6 is concerned with defining the part of the OPENCOSS platform that will support an evolutionary chain of certification evidence. A chain of certification evidence is a set of pieces of evidence that are related (e.g., the agent that has created a requirements specification, the test derived from the requirements, the agent that executed the tests, the report where the tests results are documented, etc.). By evolutionary, we mean that a chain of evidence can suffer changes (e.g., a requirement is changed), thus evolve. As a result, the chain of evidence might not be adequate anymore for safety certification (e.g., the related test cases might have to be updated).

Therefore, WP6 needs to provide the necessary methods and supporting tools for the management of evidence used in the safety certification of critical systems and also to pay particular attention to situations in which the evidence changes or evolves. When evidence changes, it must be possible to determine whether the set of evidence for a system is still adequate or if new evidence and, thus, re-execution of certification-related activities is necessary. This can also apply to reuse of evidence between standards and domains. That is, how adequate a chain of evidence compliant with a given standard is to comply with another standard (in the same or in a different domain).

This document (D6.2) is the second deliverable of WP6. Its overall goal is to document the detailed requirements for evidence management of the OPENCOSS platform. As part of evidence management, WP6 will have to pay special attention to (1) evidence traceability and (2) evidence change impact analysis. Traceability between pieces of evidence has been acknowledged as a major concern for safety assurance and certification by most of OPENCOSS partners [13], as well as by other researchers and practitioners [4]. With regard to impact analysis, change management is recognised as one of the main demotivating factors for certification efforts [12]. Consequently, facilitating evidence traceability and supporting evidence change impact analysis can be regarded as two of the main features that the OPENCOSS platform should provide in relation to evidence management.

Nonetheless, other features are also necessary for evidence management in the OPENCOSS platform. More specifically, the main functional areas identified for evidence management are evidence storage, evidence traceability, evidence evaluation, evidence change impact analysis, and integration with external tools. As a

result of the background work performed for D6.2 and of the analysis of the needs of each functional area, a set of 111 detailed requirements has been specified.

In addition, some functionality for evidence management will depend on the constraints imposed by other WPs. Evidence management in the OPENCOSS platform will have to be addressed according to the conceptual frameworks developed in WP4 (Common Certification Language) and WP5 (Compositional Certification), and integrated with the work performed in WP7 (Transparent Certification and Compliance-aware Process).

1.2 Relationship with other Deliverables

D6.2 is related to other OPENCOSS deliverables, which have served as input, with which consistency must be kept, or that will use its results. These deliverables, and the relationship of D6.2 with them, are the following ones:

- D2.2 (High-level requirements) and D6.1 (Baseline for the evidence management needs of the OPENCOSS platform) have been used as main sources in order to discover requirements for evidence management of the OPENCOSS platform.
- In addition to the two deliverables indicated in the previous point, D1.1 (Constraints of the certification process), D1.2 (Use cases description and business impact), D2.1 (Business cases and user needs), D3.1 (Analysis of safety certification data of industrial use cases), D4.1 (Baseline for the common certification language), D5.1 (Baseline for the compositional certification approach), and D7.1 (Baseline for the process-specific needs of the OPENCOSS platform) have been analysed for requirements discovery.
- D2.3 (OPENCOSS platform architecture) includes the use cases related to evidence management of the OPENCOSS platform. These use cases has been as input for specification of detailed requirements too.
- D3.2 (Integration requirements and test plans) will be based on D6.2, as well as on the rest of deliverables targeted at documenting detailed requirements in OPENCOSS.
- D4.2 (Detailed requirements for the common certification language), D5.2 (Detailed requirements for the OPENCOSS compositional certification approach), and D7.2 (Detailed requirements for the process-specific needs of the OPENCOSS platform) contain the detailed requirements for other main areas of OPENCOSS.
- D4.3 (Intermediate common certification language: conceptual model) and D4.4 (Common certification language: conceptual model) will provide the evidence characterization metamodel used as basis for evidence management in the OPENCOSS platform. More details about the role of the CCL in evidence management are provided in Section 2.2.
- D5.3 (Compositional certification conceptual framework) will also define aspects that will have to be addressed in evidence management of an assurance project when dealing with compositional certification.
- D6.3 (Specification of the evidence management service infrastructure) will further analyse and develop the requirements for evidence management in order to specify the detailed architecture of the OPENCOSS service infrastructure for evidence management.
- D7.3 (Specification of the compliance-aware service infrastructure) and D7.4 (Specification of the transparent certification service infrastructure) will further analyse and develop the requirements for process-specific needs, and new requirements for evidence management might be discovered.

More details about the relationship of WP6 and other OPENCOSS WPs can be found in D6.1.

1.3 Structure

The rest of the deliverable is structured as follows. Section 2 presents background information for the deliverable and the detailed requirements for evidence management. Section 3 documents the requirements for evidence management. Section 4 presents our conclusions after finishing the deliverable. Appendix A includes a glossary of terms for the deliverable. Appendix B introduces the overall approach followed for requirements specification in D6.2. Appendix C lists the tools related to evidence management reviewed for requirements discovery. Finally, Appendix D presents a survey on the state of the practice concerning evidence management that has been conducted.

2 Background

This section presents some background information and work that might be necessary to fully understand the context, needs, and creation of D6.2. The information and work presented has allowed us to analyse, understand, and discover requirements for evidence management.

First, situations in which evidence and thus chains of evidence evolve are presented. Second, the role of the CCL in evidence management and the management of evidence and artefacts in the OPENCOSS platform are explained. Next, existing tools for evidence management and the results of a survey on the state of the practice concerning evidence management are discussed. Finally, some indications about the discovery of requirements from other deliverables are provided.

2.1 Situations in which Evidence Evolves

This section presents seven situations that practitioners can face during the development and certification processes, that might make a chain of evidence become inadequate for safety certification, and that can increase development time and cost. The situations have been discovered on the basis of previous experience on safety certification, and on input from and discussions with practitioners and researchers. It must be noted that all the situations might not addressed in WP6. Some are more related to other WPs.

2.1.1 Incomplete set of evidence

This is probably the most basic situation in which a chain of evidence might not be adequate. It corresponds to the development scenario in which evidence is gathered and structured for a new system. Therefore, evidence is collected, or at least structured, progressively. Until all the pieces of evidence that are part of a chain of evidence have not been gathered and structured, such a chain is inadequate.

This situation is also related to other scenarios addressed in OPENCOSS such as incremental certification and compositional/modular certification. Nonetheless, WP6 does not deal with adequate composition of evidence, beyond having all the necessary pieces of evidence of a chain. This aspect will be addressed in WP5.

2.1.2 System modification and recertification

This situation corresponds to a development scenario in which an already-assessed system is modified and thus a new assessment (e.g., recertification) is required. For example, a new system can be developed on the basis of an existing one. Such a new system can include, for instance, some new component.

In relation to development tools, their safety assessment is not referred to as certification, but as qualification. A tool is qualified in the sense that its results (e.g., source code) can be used as evidence for safety assurance and certification without needing, for instance, to review them. For these tools, the situation outlined would be referred to as requalification. For example, a tool aimed at verifying coding standards can require requalification as new versions are released, or clients request configurations that have not been qualified before. Qualification documentation consists of a tool qualification plan, the tool operation requirements and test cases, and the test results. Requalification would require identification of the necessary changes in these documents, based on the new evidence to provide.

2.1.3 Modifications during the development process of a system

Changes in a system and its associated artefacts (which can be used as evidence) can occur at any moment while a critical system is developed. For example, (a) a new hazard might be identified as a result of an accident in another system. Such a hazard should be analysed, and would impact other artefacts (safety requirements, design, test cases, etc.). Another scenario is, for instance, (b) a necessary change in the

architecture of system. This might impact other artefacts such as design specifications, test cases, or even source code, which might become inadequate.

In this situation, a chain of evidence might become inadequate because of (a) missing pieces of evidence or (b) the impact of the change on other piece of evidence.

2.1.4 Change in the confidence on evidence

Another situation in which evidence can evolve and thus a chain of evidence can become inadequate is the result of the change of the confidence in a piece of evidence. Confidence refers to how adequate the piece is on the basis of some criterion. For example, an expert can judge evidence adequacy, or evidence linked to an argument can be regarded as stronger (i.e., more adequate). A piece of evidence can be considered better or worse than another based on adequacy assessment.

The simplest way of adequacy assessment is probably to determine if a set of evidence can be regarded as complete (i.e., it allows justification of the fulfilment of all the criteria of a safety standard). Nonetheless, there are cases in which adequacy assessment can be more complex, based on specific pieces of evidence that are qualitative or quantitative assessed. In these cases, a change in the adequacy of a piece of evidence can affect the adequacy of the rest of pieces of a chain of evidence. For example, a change during the development of a system (e.g., related to requirements specification) that is made by an agent whose competence is not “high” (no “top confidence” on the agent) can negatively affect the confidence of the related pieces of evidence (e.g., a test case).

2.1.5 New context for a system

When an already-assessed system is to be used in a context other than what the system was certified for, then some pieces of evidence might become inadequate or new evidence might have to be provided. For example, a system for a type of train and a specific line (e.g., from Brussels to Paris) that is to be reused for the same type of train but in another line (e.g., from Rome to Milan) would not be certified per se, but new evidence would have to be provided. In the railway domain, this situation also matches the use of generic, certified applications in a specific train or line, in which impact analysis is necessary in order to determine what chains of evidence are not adequate and thus what new evidence must be generated.

Another situation related to context change is certification against another safety standard. That is, adequate evidence and chains of evidence for a standard might not be so for another (second standard). The second standard could correspond to a new standard, a new version of a standard, or a different interpretation of a standard (e.g., by a different certification authority).

2.1.6 Agreement with a certification authority

This situation corresponds to scenarios in which new or different evidence is requested by a certification authority. For example, an authority might request new evidence for some safety criteria at some moment, after having agreed previously upon how to show compliance with such criteria, in order to gain more confidence on the global safety of a system. As a result, a chain of evidence might be inadequate, for instance, because it is not complete.

2.1.7 Component reuse

The last situation presented and in which safety evidence can evolve is related to component reuse in a system. Although closely related to the first situation presented, they are not exactly the same. As a result of component reuse, new evidence might have to be provided in order to have an adequate set of chains of evidence. For example, reuse of an event recorder system for different trains might require provision of different evidence, or new evidence about the system might have to be provided. As mentioned above, this type of evidence evolution will be mainly addressed in WP5.

2.2 The Role of the Common Certification Language in Evidence Management

OPENCOSS WP4 aims at defining the CCL, a common conceptual and notational framework for specifying certification assets. The CCL would be used as a means to get mutual recognition agreement and to be employed to discuss abstract notions from different domains. This common conceptual framework for different safety standards aims to enable management of claims, evidences and arguments in a common format, sharing patterns of certification assessment and allowing cost-effective re-certification between different standards.

The current vision about the CCL is that it will consist of two main elements (Figure 1): a set of metamodels and a propositional language. The first one will provide the concepts from which models of safety standards and of the certification assets managed in an assurance project can be created. For evidence management, the concepts will be provided by means of an evidence characterization metamodel (Figure 2), from which evidence characterization models would be created. Therefore, the CCL will define what information can be and might have to be collected for the evidence items of an assurance project, as well as how to specify evidence traceability or evidence evaluations. The evidence characterization model of an assurance project might refer to elements of models of safety standards. With regard to the propositional language, it will provide and group the terminology used in safety standards and assurance projects.

In summary, evidence management in the OPENCOSS platform will be performed according to the CCL, and the CCL should also fulfil evidence management-related requirements. Consequently, new detailed requirements for evidence management might be discovered and thus specified as a result of the work for the definition of the CCL. At the same time, the work for the definition of the CCL will use the detailed requirements for evidence management presented in this deliverable as input.

Finally, the conceptual framework for compositional certification to be developed in WP5 might impose requirements for evidence management too. This will be further analysed and determined during the development of OPENCOSS task T5.2 (Compositional certification conceptual framework).

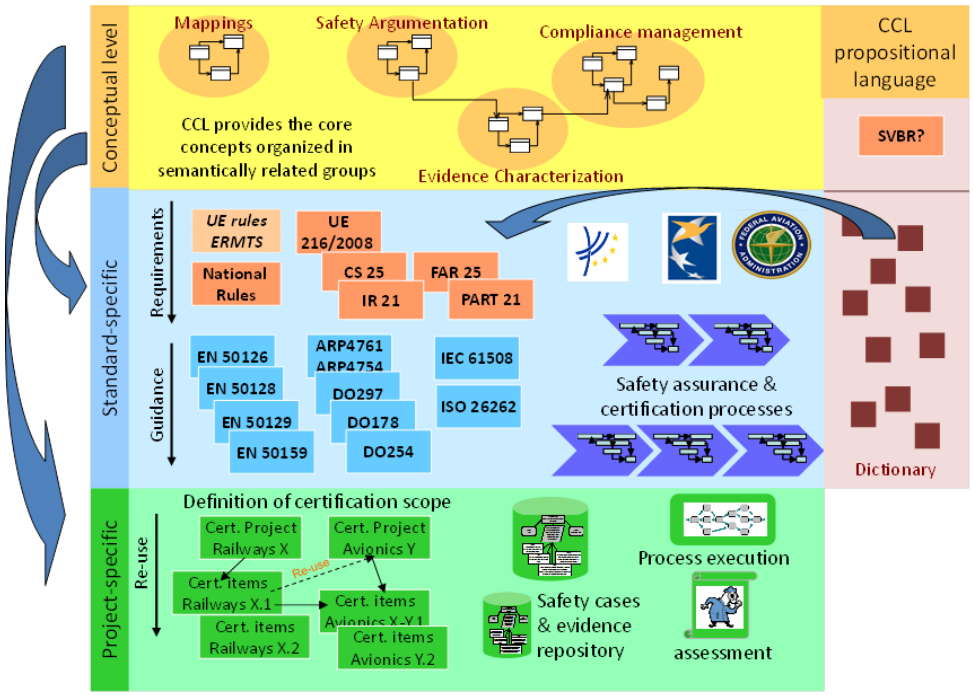


Figure 1. Overview of the CCL and its use

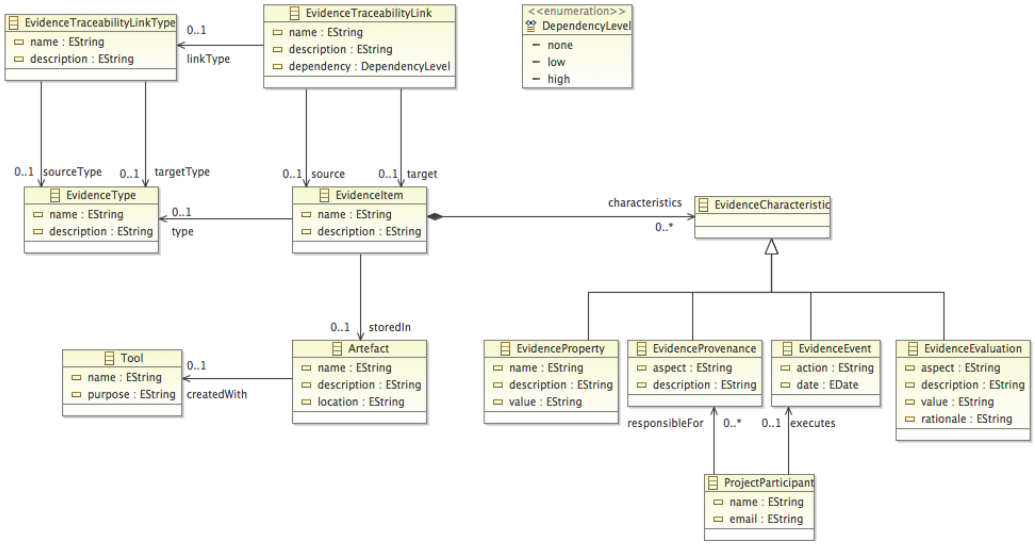


Figure 2. Fragment of proposal of CCL metamodel for evidence characterization in assurance projects

2.3 Evidence, Artefacts, and their Use in the OPENCOSS Platform

Evidence and artefact are described in deliverable D2.2 as follows:

- Evidence
  Evidence consists of a collection of documents that provide evidentiary support to a set of claims in an argument. In other words, evidence is information, based on established fact or expert judgment, which is presented to show that the claim to which it relates is valid (i.e., true) in the context of the argument. Anything that supports the claim can be presented as evidence. Often, this information is a record of some sort, demonstrating that a certain event or process took place. Evidence can be diverse as various things or artefacts may be produced as evidence, such as documents, expert testimony, test results, measurement results, records related to process, product, and people, etc.
- Artefact
  A versioned document or data item, or a collection of these; 'certification artefact' or 'assurance artefact' that indicates a document, data item, or collection required as part of the demonstration of assurance or compliance, either an evidence item, argument fragment or requirements document. Note that the term may be used at different levels of granularity - for example to refer to a single requirement, or an entire document.

In essence, information contained in an artefact can be used as evidence in an assurance project, and a relation exists between the two concepts. Nonetheless, an artefact or its content are not evidence per se, but can be used as and can become evidence if they are used to support some claim.

The OPENCOSS platform will store artefacts' information that can be used as evidence in assurance project, but not the (actual) artefacts themselves. This information is referred to as "evidence item" (Figure 2), and it can be located and stored in some artefact created with another tool. For example, a given requirement might be located in a requirements specification in a DOORS file. Therefore, *Evidence item* and *Artefact* are different concepts for evidence management in the OPENCOSS platform.

This distinction corresponds to the current perspective in OPENCOSS in relation to the use of evidence and artefacts in the OPENCOSS platform, and is in line with the results of the baseline survey on evidence management performed for D6.1. Most of the OPENCOSS partners recommended the use of a unified repository for storing and versioning safety certification documentation, and by means of a virtual repository, allowing artefacts to be physically stored in different tools and locations.

2.4 Tools for Evidence Management

Several tools for evidence management were reviewed for D6.1. Nonetheless, it was concluded that the review had to be extended, especially focusing on commercial tools that supported evidence management. For this reason, new effort has been spent for D6.2 towards finding and analysing more tools. As a result, a final set of 71 tools that might relevant for evidence management in OPENCOSS has been identified. The set of tools is presented in Appendix C.

The analysis of the tools has allowed us to determine:

- Functionality provided by existing tools that the OPENCOSS platform should provide (because it is essential for evidence management)
- Functionality provided by existing tools that the OPENCOSS platform should not provide (because it is widely provided by existing tools, or out of the scope of OPENCOSS)
- Functionality that is missing in existing, current tools, could differentiate the OPENCOSS platform from them, and could be very beneficial to safety assurance and certification practitioners.

This section summarises the main findings from the analysis performed.

Examples of functionality that the OPENCOSS platform should provide is:

- Integration with product engineering tools
- Evidence import/export
- Traceability between the information used as evidence
- Evidence dependency and change impact analysis
- Evidence and chains of evidence consistency checking
- Dashboards about the status of an assurance project and about its the body of evidence

The OPENCOSS platform should not be targeted at:

- Creation of development and V&V artefacts (e.g., modelling of fault trees)
- Supporting practices specific to only a given domain or standard (i.e., the platform must aim to be generic so that it can be used in different domains and for different safety standards)
- Requiring proprietary technologies for its use (e.g., DOORS)

Finally, the two main areas in which the tools reviewed could be improved are the following ones:

- Advanced traceability management, in line with some recent works that have studied, for instance, the granularity of evidence/artefacts specification for adequately tracing them, and traces suggestion [3].
- Advanced impact analysis, in which not only affected elements are identified but also different change effects and measures are taken into account (need for revoking evidence, re-evaluating it, providing new evidence…).

With regard to the user interface of the tools reviewed, those for which OPENCOSS partners has shown interest correspond to the following tools:

- Appraisal Assistant
- Atego Process Director
- Atego Workbench
- DOORS/Traceline
- Goedelworks
- Medini Analyze
- Parasoft Concerto
- Tracevis

The interest shown is mostly related to the way of (1) showing an overall picture the body of evidence of an assurance project, (2) indicating the status of the body of evidence and possible required actions on it, and (3) showing traceability and dependencies between elements.

It has also to be mentioned that the ideas behind SCMSs and software certificates had been taken initially taken as the main references for evidence management and evidence evolution management. Work on SCMSs was reviewed in D6.1. After further analysis, it is clear that there exists much functionality associated to SCMSs that the OPENCOSS platform should provide. For example, the platform should maintain links between artefacts (i.e., pieces of evidence), enable browsing of artefacts history and evolution, and perform change impact analysis.

However, the work performed so far in the context of SCMSs has almost exclusively dealt with small sets of artefacts, such as code and V&V results. Their application to the whole set of artefacts to produce and manage in an assurance project might not be feasible, or some aspects might have to be modified. For example, the degree of automation proposed for SCMSs might correspond to what OPENCOSS stakeholders consider adequate. They may prefer tool assistance instead of full automation, so that, for instance, humans make the ultimate decision about the validity or adequacy of evidence.

In summary, although many ideas and principles behind SCMSs have been taken into account in the RE process for evidence management of the OPENCOSS platform, and will be taken into account for its design, it seems that some aspects might require adaptation to the specific needs and vision of OPENCOSS.

Finally, an overall conclusion from the review of tools for evidence management is that most of those that seem to be more related to the evidence needs of the OPENCOSS platform are prototypes resulting from research projects. Mature, commercial tools for evidence management do not widely exist, and the most usable and interesting tools aim to support system/software development. Nonetheless, we consider that their functionality could be adapted and extended for specific purposes related to evidence management in safety assurance and certification.

2.5 State of the Practice concerning Evidence Management

One of the conclusions of D6.1 was that a deeper analysis of the state of the practice concerning evidence management was necessary. A systematic literature review and a baseline survey within OPENCOSS were conducted, but a broader, overall analysis was still regarded as necessary. As a solution, a survey with practitioners involved in evidence management for compliance of critical computer-based system with safety standards has been designed and run as part of the work for D6.2.

The questionnaire designed and the results obtained are presented in Appendix D. This section summarises and discusses those results regarded as most relevant for the requirements for evidence management of the OPENCOSS platform. Aspects that might be further studied in future tasks because of their potential relevance or usefulness are also analysed.

Answers to the questions about the types of information and artefacts used as evidence suggests that the main candidate tools to be integrated with the OPENCOSS platform are those used for requirements specification/management, design, and testing. Other tools whose integration could be important are those used for risk analysis/management. It should be studied to what extent specific tools or general-purpose ones (e.g., Word and Excel) are used for collection and management of evidence. It should then be determined if they should be integrated.

The results of the survey indicate that much manual work is still performed when having to check (1) completeness of the body of evidence and (2) evidence change impact analysis. Therefore, the OPENCOSS

platform could significantly contribute to the improvement of the state of the practice in these aspects. Information about change impact should be recorded in the platform, and traceability between pieces of evidence should be shown by means of matrices. The benefits from using other ways to show traceability (e.g., by means of models) could be studied in OPENCOSS.

Textual templates seem to be the most commonly used technique for structuring of evidence, thus support for this technique should be provided and it could be somehow embedded in the interaction of users with the system (e.g., indicating the overall categories of information to provide when storing evidence). The actual need and advantages of using graphical models for structuring evidence could be investigated, since the results suggests that practitioners use text more often than models.

When assessing (aka evaluating) evidence, it is necessary that the OPENCOSS platform allows the users to record the rationale behind the assessment. The results of the survey also seem to indicate that support for qualitative assessment has a higher priority than support for quantitative assessment. When specifying confidence in a piece of evidence, information about the confidence in related pieces should be provided. When changing the confidence in a piece of evidence, it should be determined if the confidence in related pieces is affected. For the latter two aspects, the degree of expected support and automation from the OPENCOSS platform should be determined.

The results suggest that new situations in safety assurance and certification (e.g., assurance and certification of new types of systems) are especially challenging and important for practitioners, thus knowledge about and insights into previous experiences could really help them. It might be important, for instance, to provide users with guidance when they have to determine confidence in evidence, and with information about evidence management and evidence-related issues in past projects.

Last but not least, it must be indicated that the results obtained from the survey conducted for D6.2 are, in general and for the aspects in common, in line with the results of the baseline survey conducted for D6.1. Therefore, it can be considered that the knowledge and practices of OPENCOSS partners adequately represent and correspond to the current state of the practice for evidence management.

2.6 Requirements Discovery from Previous OPENCOSS Deliverables

As stated in Section 1.2, several OPENCOSS deliverables have been used as input in order to discover requirements for evidence management of the OPENCOSS platform. This section explains some aspects that must be taken into account in order to understand how the deliverables have been used analysed.

When analysing previous deliverables, two types of information were identified. One type corresponds to general, abstract, domain-independent requirements for the OPENCOSS platform. That is, these requirements can relate to any assurance project. For example, a requirement discovered in D2.2 is “Propagate change information” (in relation to evidence change impact analysis). The other type of information corresponds to detailed, concrete, domain-specific requirements. These requirements would only apply to some assurance projects. For example, a requirement discovered in D1.2 is “The system should be able to store information about malfunctions, hazardous events, safety goals (id, description), associated safe states and associated external measures”, which is related to the automotive use case for evaluating OPENCOSS results.

It is important to note that, in general, the requirements specified for evidence management of the OPENCOSS platform correspond to the first type of information. Otherwise, a cross-domain platform might not be provided. When analysing requirements that correspond to the second type, their abstraction level has been raised in order to specify a requirement regarded as adequate. For the example presented above, it has been considered that the elements mentioned (malfunctions, hazardous events, etc.) corresponds to

evidence items and evidence traceability links between them, whose specification and storage for evidence management has been taken into account.

For some requirements that correspond to the first type of information, changes have been made in order to refine and provide more details about the requirements. For example, some high-level requirements in D2.2 were related to several WPs (e.g., WP5 and WP6), thus the needs specific to WP6 have had to be extracted.

In summary, some needs and requirements contained in other deliverables might not directly map to the detailed requirements for evidence management specified in this deliverable. Some modifications have been necessary for providing the adequate level of abstraction and detail.

3 Requirements for Evidence Management

Regarding the specific needs of WP6, the mission of the OPENCOSS platform is to manage the evidence required in an assurance project, focusing on its chains of evidence, and also aiming to increase the effectiveness and efficiency of the management. As shown below, evidence management involves different activities, such as evidence collection, combination, and evaluation.

The overall approach followed in WP6 for requirements discovery and specification is presented in Appendix B. The approach is mainly based on the modelling of business process for understanding and analysing the application domain, goal modelling for determination and analysis of stakeholders' needs (product level requirements) and system features (feature level requirements), use case specification (function level requirements), and the specification of detailed requirements (referred to as component level requirements) by means of textual statements and user interface mock-ups.

The following subsections present the high-level business process, product level and feature level requirements, and the component level requirements for evidence management. The use cases for evidence management have been included in D2.3.

3.1 High-Level Business Process

Figure 3 shows the overall, high-level business process that is executed in an assurance project for evidence management. It has been modelled with the BPMN notation [9], and complements the business process modelled in other deliverables by focusing on evidence management needs. For example, business process models for safety assessment were included in D2.1. It must also be noted that the business process model might not exactly correspond to reality (e.g., all the evidence to provide might not be determined before its collection), and some details have been omitted for the sake of clarity and simplicity. The purpose of the model is to provide an overview of the domain under analysis.

When managing evidence in an assurance project, the first step is usually to determine what evidence must be provided. Afterwards, the evidence items that conform the body of evidence of the project must be collected, and might also have to be evaluated and traced to other evidence items (creation of chains of evidence). During this process, it might necessary to make changes in the evidence items, and such changes might impact other items. As a result, issues and problems (e.g., inconsistency) might appear in the body of evidence and would have to be addressed. Otherwise, the body of evidence might not be adequate.

Once the body of evidence of the assurance project is regarded as adequate (i.e., it is regarded as complete and no issues exist), the process can be finished.

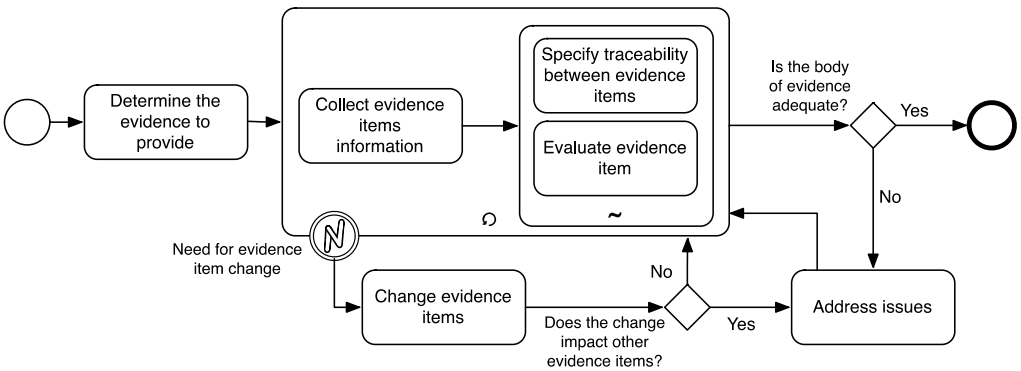


Figure 3. Business process model for evidence management in an assurance project

3.2 Product Level and Feature Level Requirements

This section presents the product level and feature level requirements that have been specified for the OPENCOSS platform in relation to evidence management for an assurance project. They are graphically shown in the Map diagram [15] of Figure 4.

Three product level requirements have been specified. They are goals that are expected to be achieved as result of the use of the OPENCOSS platform, and correspond to the main aspects of an assurance project that the stakeholders aim to change in current practice regarding evidence management. The product level requirements are:

- Facilitate evidence combination, so that users can more easily and more efficiently identify, specify, and maintain relationships and traceability links between the evidence items of an assurance project.
- Facilitate evidence change management, so that users receive help when having to determine and address how changes in the body of evidence of an assurance project impact individual evidence items of the project.
- Improve knowledge about evidence status, so that users find fewer difficulties when needing to determine evidence adequacy in an assurance project and the actions that have to be taken to reach it.

The feature level requirements specified, which contribute to the achievement of the above goals, are:

- Collection of evidence from external tools, so that users can import information used as evidence in an assurance project from product engineering tools used for the development of critical systems (e.g., V&V tools).
- Provision of a unified evidence repository, so that users can maintain the whole body of evidence of an assurance project in only one tool.
- Support for evidence traceability specification, so that users receive help and suggestions when dealing with determination of the relationships to maintain between the evidence items of an assurance project.
- Evidence change impact analysis, so that users get information about the effects of the changes in the body of evidence of an assurance project, and thus how the changes affect the evidence items of the project.
- Indication of evidence traceability needs, so that users get information about what evidence items of an assurance should be traced to others, or about what traces should be modified or updated.
- Synchronization of changes with external tools, so that users effectively maintain consistency in the evidence and information of an assurance project imported from external tools, and are aware of possible differences between the information stored in the OPENCOSS platform and the information created with external tools.
- Identification of evidence gaps, so that users get information about what evidence has still to be collected in an assurance project.
- Detection of evidence inconsistency, so that users can better know what evidence items of an assurance project require execution of some actions in order to obtain a adequate and consistent body of evidence.
- Detection of evidence evaluation needs, so that users can better know what evidence items of an assurance project require evaluation or re-evaluation.
- Report on evidence-related required actions, so that users are aware of and can easily check the actions that have to be taken in an assurance project in order to complete the necessary evidence-related efforts.

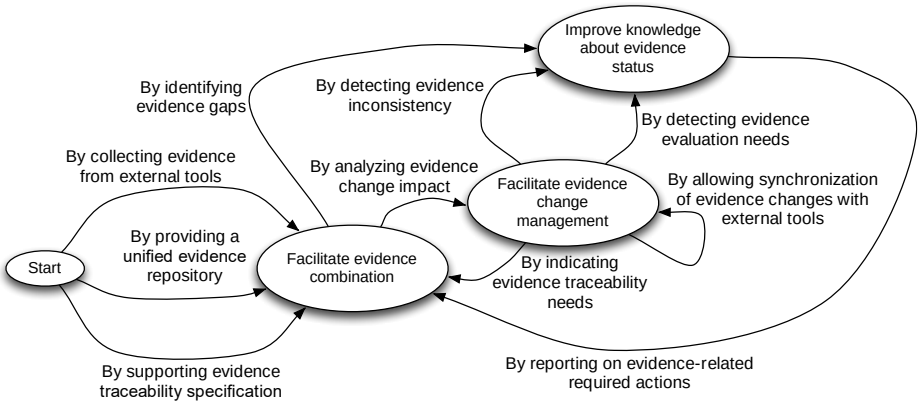


Figure 4. Map diagram for evidence management of the OPENCOSS platform

3.3 Component Level Requirements

Five main functional areas have been defined for specifying component level requirements related to evidence management of the OPENCOSS platform:

- Evidence storage, concerned with the determination, specification, and structuring of the evidence items of an assurance project.
- Evidence traceability, concerned with the specification and adequate maintenance of traceability between evidence items of an assurance project.
- Evidence evaluation, concerned with the assessment of the completeness and adequacy of the body of evidence of an assurance project, and of specific criteria defined for evaluation of individual evidence items.
- Evidence change impact analysis, concerned with the identification and analysis of possible effects resulting from changes in the body of evidence of an assurance project.
- Integration with external tools, concerned with the possibility of importing and exporting information from and to external tools, and information synchronization with them.

For each functional area, requirements have also been specified in order to provide mechanisms that help users perform these tasks. For example, requirements have been specified in relation to different ways of visualizing evidence traceability, in order to help users check and manage evidence traceability links.

Each component level requirements has been specified according to the template used in D4.2 for requirements specification. The table includes the ID, description, rationale, stakeholders, status, and priority. At this moment the status of all the requirements is “proposed” (see D4.2 for a justification). Priority specification is based on the MoSCoW approach (see D2.2 for details).

Some aspects specific to specification of detailed requirements in this deliverable are the following ones:

- The table for requirements specification has been extended with the “type” of requirement. The type can be functional or non-functional, existing several categories for non-functional requirements (see D2.2 for more details).
- The textual patterns presented in Appendix B have been used for specifying the description of the requirements.
- User interface mock-ups have been included; they are mainly based on the preferences indicated by OPENCOSS partners in relation to the user interface of existing tools for evidence management reviewed.

It must also be indicated that specification of component level requirements does not fulfil all the conditions presented in Appendix B for them, or those indicated in D2.2 and D4.2 for guaranteeing the quality of a requirements specification. For example, some requirements still have to be refined for being testable, or their implementation by different people might not be very close. The reason is that several

aspects related to evidence management in the OPENCOSS platform have to be determined and further analysed in other tasks, both in tasks running at this moment and in future tasks. Requirements related to the CCL, evidence traceability, evidence change impact analysis, and integration with external tools will have to be refined as work in these areas advances. It also still has to be studied in depth how the compositional certification framework defined will constrain evidence management in the OPENCOSS platform, thus new requirements, not considered yet, might be discovered.

In summary, the detailed requirements specified in this deliverable will evolve as work progresses in OPENCOSS areas such as CCL specification, design for evidence management, integration requirements for the tools created in WP4-7, and integration with external tools. Consequently, it is expected that new versions of the deliverables will be released in future, as modifications and refinements are made.

The following subsections contain the details of the component level requirements specified for each functional area, and the user interface mock-ups. The requirements of each functional area are also listed and summarized in Tables 1, 2, 3, 4, and 5.

Table 1. Component level requirements for evidence storage

ID	Name
01_01	Evidence item to provide
01_02	Evidence types to provide
01_03	Apply existing evidence characterization model
01_04	CCL-based evidence item characteristics
01_05	Association of evidence types to evidence items
01_06	Apply existing evidence types
01_07	Association of artefacts to evidence items
01_08	CCL-based artefact information
01_09	Modification of artefacts associated to evidence items
01_10	Association of already-used artefacts to evidence items
01_11	Evidence item drop
01_12	Confirmation of evidence item drop
01_13	Drop of evidence item characteristics after evidence item drop
01_14	Drop of artefacts after evidence drop
01_15	Evidence item information modification
01_16	Evidence item reuse
01_17	Record of evidence item history
01_18	Automatic evidence repository creation
01_19	Evidence item characteristics status specification
01_20	Use of colours to report on evidence item characteristics completeness
01_21	Use of colour to report on evidence item characteristics incompleteness
01_22	Tree-view for display of evidence items
01_23	Intermediate nodes of tree-view
01_24	Modification of intermediate nodes of tree-view
01_25	Drop of intermediate nodes of tree-view for display of evidence items
01_26	Association of evidence types to intermediate nodes of tree-view
01_27	Proposal of intermediate nodes for tree-view
01_28	Inclusion of evidence items in tree-view
01_29	Automatic inclusion of evidence items in tree-view
01_30	Generation of HTML-based evidence reports
01_31	Generation of document-based evidence reports
01_32	Customization of evidence report generation

Table 2. Component level requirements for evidence traceability

ID	Name
02_01	Evidence traceability link specification
02_02	CCL-based specification of evidence traceability links
02_03	Definition of evidence traceability links types
02_04	Reuse of evidence traceability links types
02_05	Display of source traced evidence items
02_06	Display of target traced evidence items
02_07	Notification of evidence traceability links correctness
02_08	Use of colours for evidence traceability links correctness
02_09	Evidence traceability link proposal
02_10	Evidence traceability link drop
02_11	Confirmation of Evidence traceability link drop
02_12	Evidence traceability link drop after evidence item drop
02_13	Evidence traceability links completeness assessment
02_14	Matrices for traceability visualization
02_15	Source and target evidence types in matrices for traceability visualization
02_16	Display of required actions related to evidence evaluation
02_17	Creation of evidence traceability links in matrices
02_18	Models for visualization of chains of evidence
02_19	Tables for traceability visualization
02_20	Evidence types of tables for traceability visualization
02_21	Columns of tables for evidence visualization
02_22	Creation of evidence traceability links in tables
02_23	Reuse of tables for traceability visualization

Table 3. Component level requirements for evidence evaluation

ID	Name
03_01	CCL-based evidence item evaluation
03_02	Evidence evaluation rationale
03_03	Possible relationships with the evaluation of related evidence items
03_04	Definition of evidence item evaluation criteria
03_05	Modification of evidence item evaluation criteria
03_06	Reuse of evidence item evaluation criteria
03_07	Definition of categories for evidence item evaluation criteria
03_08	Modification of categories for evidence item evaluation criteria
03_09	Reuse of categories for evidence item evaluation criteria
03_10	Modification of evidence item evaluation
03_11	Specification of required actions after evidence evaluation
03_12	Email for required actions after evidence evaluation
03_13	Evidence item associated to a required actions after evidence evaluation
03_14	Use of colours in evidence items associated to a required actions after evidence evaluation
03_15	Status of actions related to evidence evaluation
03_16	Display of required actions related to evidence evaluation
03_17	Completion of actions required as a result of evidence evaluation
03_18	Use of colours to report on the status of actions required after evidence evaluation
03_19	Report on evidence set completeness
03_20	Report on evidence set adequacy

Table 4. Component level requirements for evidence change impact analysis

ID	Name
04_01	Determination of evidence items affected by a change
04_02	Determination of evidence traceability links affected by a change
04_03	Confirmation of changes
04_04	Impact of a possible evidence change
04_05	Specification of actions required after evidence item changes
04_06	Evidence items associated to actions required after evidence item changes
04_07	Evidence traceability links associated to actions required after evidence item changes
04_08	Suggestion of actions after evidence item change
04_09	Email with required actions after evidence item changes
04_10	Status of actions related to evidence change
04_11	Display of required actions after evidence item changes
04_12	Completion of actions required after evidence item changes
04_13	Use of colours to report on the status of actions required after evidence item changes
04_14	Use of colours to report on the status of evidence items after evidence item changes
04_15	Use of colours to report on the status of evidence traceability links after evidence item changes

Table 5. Component level requirements for integration with external tools

ID	Name
05_01	Evidence item information import
05_02	Automatic artefact association to imported evidence item
05_03	Tool information about the artefact associated to imported evidence item information
05_04	Evidence traceability link import
05_05	Automatic artefact association to imported evidence traceability link
05_06	Tool information about the artefact associated to imported evidence traceability link
05_07	Evidence traceability link export
05_08	SAEM import
05_09	SAEM export
05_10	Automatic update of evidence items in the OPENCOSS platform
05_11	Automatic update of evidence items in external tools
05_12	Notification of change of imported evidence item in the OPENCOSS platform
05_13	Notification of change of imported evidence item in external tool
05_14	Use of colours to indicate change of imported evidence item in the OPENCOSS platform
05_15	Use of colours to indicate change of imported evidence item in external tool
05_16	Automatic update of evidence traceability links in the OPENCOSS platform
05_17	Automatic update of evidence traceability links in external tools
05_18	Notification of change of imported evidence traceability link in the OPENCOSS platform
05_19	Notification of change of imported evidence traceability link in external tool
05_20	Use of colours to indicate change of imported evidence traceability links in the OPENCOSS platform
05_21	Use of colours to indicate change of imported evidence traceability link in external tools

3.3.1 Functional Area 1: Evidence Storage

The following 32 component level requirements have been specified for evidence storage in the OPENCOSS platform.

3.3.1.1 Evidence item to provide

01_01	
Type	Functional
Description	The OPENCOSS platform shall provide the users with the ability to indicate the evidence items that will have to be provided for an assurance project
Rationale	To be able to evaluate evidence and assurance projects at some point, it must be possible to indicate the evidence items that have to be provided for an assurance project.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.2 Evidence types to provide

01_01	
Type	Functional
Description	The OPENCOSS platform shall provide the users with the ability to indicate the evidence types of the evidence items to be provided for an assurance project
Rationale	The Assurance Manager must be able to indicate the evidence type of an evidence item when such an item has been created. Similarly, Engineering Tools and Product Engineers must be able to indicate the evidence type of an item when, for instance, updating evidence item information. Indicating the required evidence types facilitates the traceability and evaluation of evidence at a later stage.
Stakeholders	Assurance Manager, Engineering Tool, and Product Engineer
Status	Proposed
Priority	Must-have

3.3.1.3 Apply existing evidence characterization model

01_03	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to select an existing evidence characterization model for an assurance project
Rationale	By selecting an existing evidence characterization model, the Assurance Manager indicates how the evidence items should be characterized. Evidence can, for example, be characterized according to SAEM in terms of directness/indirectness, primary or secondary status of the information, originality or derivation of the source. Evidence characterization models might also be created according to the metamodels provided by the CCL
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.4 CCL-based evidence item characteristics

01_04	
Type	Functional
Description	The OPENCOSS platform shall store the characteristics of the evidence items of an assurance project according to the CCL
Rationale	Besides the evidence itself, created by an external Engineering Tool or a Product Engineer, there should be characteristics of the evidence, which are properties describing the evidence. In SAEM, for instance, evidence is described in properties like the qualitative levels as relevance, confidence, significance, support, strength, reporting, and accuracy.
Stakeholders	Assurance Manager, Product Engineer, and Engineering Tool
Status	Proposed
Priority	Must-have

3.3.1.5 Association of evidence types to evidence items

01_05	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate the evidence type of the current evidence items of an assurance project
Rationale	Evidence type must be indicated in order to facilitate traceability and evaluation of evidence at a later stage.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.6 Apply existing evidence types

01_06	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to use in an assurance project evidence types that have been defined in another project
Rationale	When creating evidence items, the Assurance Manager must be able to assign predefined evidence types to evidence items.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.7 Association of artefacts to evidence items

01_07	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate the artefact to which evidence items of an assurance project is associated
Rationale	The Assurance Manager must be able to indicate the actual artefact in which an evidence items is located.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.8 CCL-based artefact information

01_08	
Type	Functional
Description	The OPENCOSS platform shall store the CCL-information of the artefact to which an evidence item of an assurance project is associated.
Rationale	The platform needs to be able to store the characteristics of an artefact defined by the CCL
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.9 Modification of artefacts associated to evidence items

01_09	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to modify the information about the artefact to which an evidence item of an assurance project is associated
Rationale	The information of an artefact might change during an assurance project, thus this possibility must be taken into account in the platform.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.10 Association of already-used artefacts to evidence items

01_10	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to associate evidence items of an assurance project with an artefact that has been used in another project
Rationale	To enable reuse of artefacts used in other projects, it must be possible to associate evidence items to such artefacts.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.11 Evidence item drop

01_11	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to drop evidence items of an assurance project
Rationale	The user must be able to remove evidence items, but rather than they will be deleted, they will be archived.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.12 Confirmation of evidence item drop

01_12	
Type	Functional
Description	When a user wants to drop an evidence item from an assurance project, the OPENCOSS platform shall request confirmation from the user for this action and subsequent related actions
Rationale	To avoid accidental removal of an evidence item from an assurance project, confirmation is requested when a user drops such an item.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.13 Drop of evidence item characteristics after evidence item drop

01_13	
Type	Functional
Description	When an evidence item is dropped from an assurance project, the OPENCOSS platform shall drop all the evidence information (types and characteristics) related to the evidence item from the project
Rationale	Evidence characteristics are associated with a single evidence item. If such an item is dropped, the evidence characteristics should be dropped too.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.14 Drop of artefacts after evidence drop

01_14	
Type	Functional
Description	When an evidence item is dropped from an assurance project and its associated artefact is not associated with other evidence items, the OPENCOSS platform shall drop the artefact from the project
Rationale	Artefacts that have no associations with evidence items have no purpose in the context of an assurance project. If an evidence item is dropped the artefact should not be referred to by (or associated with) the assurance project anymore.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.15 Evidence item information modification

01_15	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to modify the information of the evidence items of an assurance project
Rationale	This is a general requirement, where the type and characterisation are specific parts of the information.
Stakeholders	Assurance Manager, Product Engineer, and Engineering Tool
Status	Proposed
Priority	Must-have

3.3.1.16 Evidence item reuse

01_16	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to copy evidence items to an assurance project that previously have been stored in another assurance project.
Rationale	Evidence-related effort should be minimized (and reuse maximised) if possible.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.17 Record of evidence item history

01_17	
Type	Functional
Description	The OPENCOSS platform shall store the time and date of the actions performed to the evidence items of an assurance project
Rationale	Storing the time and date of actions performed to the evidence items improves the transparency of the assurance process.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.18 Automatic evidence repository creation

01_18	
Type	Functional
Description	The OPENCOSS platform shall create the evidence repository of an assurance project from its evidence characterization model
Rationale	After the Assurance Manager has selected an appropriate evidence characterization model for an assurance project, the OPENCOSS platform must create an evidence repository to store the evidence items related to the project in a way that matches the selected characterization model.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.19 Evidence item characteristics status specification

01_19	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate whether the specification of characteristics for an evidence item of an assurance project is regarded as complete
Rationale	By manually indicating that the specification of characteristics for an evidence item is complete, evaluation of evidence is simplified.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.1.20 Use of colours to report on evidence item characteristics completeness

01_20	
Type	Non-functional (Usability)
Description	When the specification of characteristics for an evidence item of an assurance project is regarded as complete, the OPENCOSS platform shall colour the evidence item in green
Rationale	By providing visual feedback regarding the completeness of the specification of characteristics of an evidence item, the process of evaluating evidence is simplified.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.21 Use of colour to report on evidence item characteristics incompleteness

01_21	
Type	Non-functional (Usability)
Description	When the specification of characteristics for an evidence item of an assurance project is not regarded as complete, the OPENCOSS platform shall colour the evidence item in yellow
Rationale	By providing visual feedback regarding the completeness of the specification of characteristics of an evidence item, the process of evaluating evidence is simplified.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.22 Tree-view for display of evidence items

01_22	
Type	Non-functional (Usability)
Description	The OPENCOSS platform shall show the set of evidence items of an assurance projects by means of a tree-view
Rationale	By showing the set of evidence items by means of a tree-view, the structure of the body of evidence of an assurance project is visualised.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.23 Intermediate nodes of tree-view

01_23	
Type	Non-functional (Usability)
Description	The OPENCOSS platform shall provide users with the ability to define intermediate nodes in the tree-view for display of the evidence items of an assurance project
Rationale	The user can define additional intermediate nodes facilitate evidence structuring and visualization.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.24 Modification of intermediate nodes of tree-view

01_24	
Type	Non-functional (Usability)
Description	The OPENCOSS platform shall provide users with the ability to modify the intermediate nodes in the tree-view for display of the evidence items of an assurance project
Rationale	The user can modify intermediate nodes to facilitate evidence structuring and visualization.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.25 Drop of intermediate nodes of tree-view for display of evidence items

01_25	
Type	Non-functional (Usability)
Description	The OPENCOSS platform shall provide users with the ability to drop intermediate nodes in the tree-view for display of the evidence items of an assurance project
Rationale	The user can remove intermediate nodes to facilitate evidence structuring and visualization.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.26 Association of evidence types to intermediate nodes of tree-view

01_26	
Type	Non-functional (Usability)
Description	The OPENCOSS platform shall provide users with the ability to associate evidence types to the intermediate nodes of the tree-view for display of the evidence items of an assurance project
Rationale	By associating evidence types to an intermediate node, evidence structuring and visualization is facilitated.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.27 Proposal of intermediate nodes for tree-view

01_27	
Type	Non-functional (Usability)
Description	When and evidence characterization model has been selected for an assurance project, the OPENCOSS platform shall propose intermediate nodes for the tree-view for display of the evidence items of the project according to the evidence characterization model
Rationale	An evidence characterization model can propose evidence types and a certain structure for evidence, which can be used as basis for the tree-view.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.28 Inclusion of evidence items in tree-view

01_28	
Type	Non-functional (Usability)
Description	The OPENCOSS platform shall provide users with the ability to include the evidence items of an assurance project in the structure of the tree-view for display of the evidence items of the project
Rationale	When specifying evidence items and using a tree-view to structure and visualize evidence, users can indicate where new evidence items must be included in the tree-view.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.29 Automatic inclusion of evidence items in tree-view

01_29	
Type	Non-functional (Usability)
Description	When an evidence characterization model has been selected for an assurance project, the evidence items of the project are associated to evidence types, and the evidence types have been used for definition of intermediate nodes of the tree-view for display of the evidence items of the project, the OPENCOSS platform shall automatically include the evidence items of the project in the tree-view nodes of their associated evidence types
Rationale	When specifying evidence items and using a tree-view to structure and visualize evidence, new evidence might be automatically categorized in the tree-view.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.30 Generation of HTML-based evidence reports

01_30	
Type	Functional
Description	The OPENCOSS platform shall be able to generate reports about the body of evidence of an assurance project in the form of HTML files
Rationale	To facilitate activities such as evaluation of evidence and analysis of evidence change impact, the platform must be able to produce human-readable reports about the body of evidence.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.31 Generation of document-based evidence reports

01_31	
Type	Functional
Description	The OPENCOSS platform shall be able to generate reports about the body of evidence of an assurance project by means of documents in PDF, RTF, and MS Word formats
Rationale	To facilitate activities such as evaluation of evidence and analysis of evidence change impact, the platform must be able to produce human-readable reports about the body of evidence.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.1.32 Customization of evidence report generation

01_32	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate the information to be included in the reports about the body of evidence of an assurance project
Rationale	By reducing the amount of information in a report, these reports can be tailored for different audiences.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2 Functional Area 2: Evidence Traceability

The following 23 component level requirements have been specified for evidence traceability in the OPENCOSS platform.

3.3.2.1 Evidence traceability link specification

02_01	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to specify evidence traceability links between evidence items of an assurance project
Rationale	The ability to create evidence traceability links is a prerequisite for the creation of evidence chains.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.2 CCL-based specification of evidence traceability links

02_02	
Type	Functional
Description	The OPENCOSS platform shall store the information of the evidence traceability links of an assurance project according to the CCL
Rationale	Storing the information of the evidence traceability links according to the CCL is a prerequisite for supporting automated processes such as gap analysis. Additionally, this improves the transparency of the assurance process. The information itself can be regarded as evidence too.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.3 Definition of evidence traceability links types

02_03	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to define evidence traceability link types for an assurance project
Rationale	In assurance projects, it is possible an Assurance Manager wants to maintain specific types of traceability between evidence types. For example, between requirements and test cases.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.4 Reuse of evidence traceability links types

02_04	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to reuse in an assurance project evidence traceability link types defined in another project
Rationale	It is important that Assurance Managers can reuse evidence traceability links types in order to reduce the effort to spend in assurance projects.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.5 Display of source traced evidence items

02_05	
Type	Functional
Description	The OPENCOSS platform shall show the evidence items of an assurance project that are the source of evidence traceability links for which a given evidence item is the target
Rationale	Traceability links can be navigated to retrieve information about related evidence items.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.6 Display of target traced evidence items

02_06	
Type	Functional
Description	The OPENCOSS platform shall show the evidence items of an assurance project that are the target of evidence traceability links for which a given evidence item is the source
Rationale	Traceability links can be navigated to retrieve information about related evidence items.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.7 Notification of evidence traceability links correctness

02_07	
Type	Functional
Description	When an evidence characterization model has been selected for an assurance project and specifying an evidence traceability link for an evidence item of the project, the OPENCOSS platform shall indicate if the evidence traceability link is correct according to the model
Rationale	To evaluate evidence and to evaluate the consistency of an evidence chain, the Assurance Manager must know whether all evidence traceability links are correct according to the selected evidence characterization model.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.8 Use of colours for evidence traceability links correctness

02_08	
Type	Non-functional (Usability)
Description	When an evidence traceability link of an assurance project is not correct according to the evidence characterization model selected for the project, the OPENCOSS platform shall colour such a link in red
Rationale	To evaluate evidence and to evaluate the consistency of an evidence claim, the Assurance Manager must know whether all evidence traceability links are correct according to the selected evidence characterization model, and assistance in the form colouring links can be very helpful.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.9 Evidence traceability link proposal

02_09	
Type	Functional
Description	When an evidence characterization model has been selected for an assurance project and specifying evidence traceability links for an evidence item of the project, the OPENCOSS platform shall show possible evidence items with which evidence traceability links might be specified according to the model
Rationale	To facilitate the process of adding traceability links between evidence items, the platform suggests possible links according to the selected evidence characterization model.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.10 Evidence traceability link drop

02_10	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to drop evidence traceability links of an assurance project
Rationale	The Assurance Manager must be able to drop evidence traceability links, for instance because the evidence items connected by the link may become unrelated at some point.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.11 Confirmation of Evidence traceability link drop

02_11	
Type	Functional
Description	When a user aims to drop an evidence traceability link of an assurance project, the OPENCOSS platform shall require confirmation of the action from the user
Rationale	To prevent accidental removal or evidence traceability links, confirmation is requested.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.12 Evidence traceability link drop after evidence item drop

02_12	
Type	Functional
Description	When an evidence item of an assurance project is the source or the target of an evidence traceability link, and the item is dropped from the project, the OPENCOSS platform shall drop the evidence traceability link from the project too
Rationale	To retain the consistency of traceability links, links related to dropped items should be removed.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.13 Evidence traceability links completeness assessment

02_13	
Type	Functional
Description	When an evidence characterization model has been selected for an assurance project, the OPENCOSS platform shall indicate the required missing links according to the model
Rationale	If gathered evidence and the corresponding links do not adhere to the selected evidence characterization model because of missing links, this inconsistency must be indicated to be able to resolve it.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.14 Matrices for traceability visualization

02_14	
Type	Functional
Description	The OPENCOSS platform shall be able to show evidence traceability links between the evidence items of an assurance project by means of traceability matrices
Rationale	The Assurance Manager must be able to evaluate the consistency of evidence chains. Traceability matrices provide a way to do so.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.15 Source and target evidence types in matrices for traceability visualization

02_15	
Type	Functional
Description	When using matrices to show evidence traceability links between the evidence items of an assurance project, the OPENCOSS platform shall require that the users indicate the source and target evidence types of the evidence traceability links to show in a matrix
Rationale	By selecting the source and target evidence types, the user specifies the desired content of the traceability matrix.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.16 Reuse of matrices for traceability visualization

02_16	
Type	Functional
Description	When using matrices to show evidence traceability links between the evidence items of an assurance project, the OPENCOSS platform shall provide users with the ability to select the source and target evidence types of the evidence traceability links used for another matrix
Rationale	Specifying the desired content of a traceability matrix is simplified by allowing the user to select a previously defined matrix. The source and target evidence types of this matrix are also used to populate the new matrix.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.17 Creation of evidence traceability links in matrices

02_17	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to specify evidence traceability links in traceability matrices
Rationale	The Assurance Manager must be able to indicate in a traceability matrix if an evidence traceability link exists between two evidence items.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.18 Models for visualization of chains of evidence

02_18	
Type	Functional
Description	The OPENCOSS platform shall be able to show the chains of evidence to which an evidence item of an assurance project belongs by means of models
Rationale	The Assurance Manager must be able to evaluate the consistency of evidence chains. Models provide a way to do so.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.19 Tables for traceability visualization

02_19	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to define tables to show traceability between the evidence items of an assurance project
Rationale	The Assurance Manager must be able to evaluate the consistency of evidence chains. Tables provide a way to do so.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.20 Evidence types of tables for traceability visualization

02_20	
Type	Functional
Description	When showing evidence traceability by means of tables, the OPENCOSS platform shall require that the users indicate the evidence types to include in the table
Rationale	The desired content of a traceability table is specified by indicating the evidence types that should be included in the table.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.21 Columns of tables for evidence visualization

02_21	
Type	Functional
Description	The OPENCOSS platform will include a column in a table for traceability visualization for each evidence type to include in the table indicated by users
Rationale	When creating tables for showing traceability, each evidence type selected for a table will correspond to a column in the table.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.2.22 Creation of evidence traceability links in tables

02_22	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to specify evidence traceability links in traceability tables
Rationale	The Assurance Manager must be able to indicate in a traceability table if an evidence traceability link exists between two evidence items.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.2.23 Reuse of tables for traceability visualization

02_23	
Type	Functional
Description	When using tables to show evidence traceability in an assurance project, the OPENCOSS platform shall provide users with the ability to select the evidence types used for another table
Rationale	Specifying the desired content of a traceability table is simplified by allowing the user to select a previously defined table. The evidence types of this table are also used to populate the new table.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.3 Functional Area 3: Evidence Evaluation

The following 20 component level requirements have been specified for evidence evaluation in the OPENCOSS platform.

3.3.3.1 CCL-based evidence item evaluation

03_01	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to evaluate the evidence items of an assurance project according to evaluation criteria defined in the CCL
Rationale	The use of any evidence item is strictly dependent on the context in which the evidence was obtained (i.e., the assumptions and constraints driving the process for creation of the evidence item.) As a consequence, it is important to have a set of evaluation criteria in the CCL to be used to evaluate an evidence item.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.2 Evidence evaluation rationale

03_02	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to specify the rationale behind the evaluation of an evidence item of an assurance project
Rationale	The use of any evidence item is strictly dependent on the context in which the evidence was obtained (I.e., the assumptions and constraints driving the process for creation of the evidence item). As a consequence, it is important to have the possibility to evaluate any evidence item against a set of evaluation criteria and to have the possibility to motivate and comment the result.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.3 Possible relationships with the evaluation of related evidence items

03_03	
Type	Functional
Description	When an evidence item of an assurance project is related to other evidence items of the project by means of evidence traceability links, the OPENCOSS platform shall inform the user that the evaluation of the evidence item might depend on the evaluation of the others
Rationale	From the moment that it is very common for an evidence item to be directly and/or indirectly dependent/linked to other evidence items, it is important to underline the fact that the evaluation of a specific evidence item can be considered exhaustive as long as it is taken into account together with the other evidence items to which it is traced.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.3.4 Definition of evidence item evaluation criteria

03_04	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to define criteria for evaluation of evidence items of an assurance project
Rationale	The use of any evidence item is strictly dependent on the context in which the evidence was obtained. As a consequence, it is important to have a set of evaluation criteria to be used to evaluate an evidence item. Moreover, from the moment that the evaluation criteria might not always be the same and shared among different projects, but can be task-specific, company specific etc., it becomes important for the user to have the possibility to create its own set of evaluation criteria to be used to guide the evaluation process of the evidence items of a specific project.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.5 Modification of evidence item evaluation criteria

03_05	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to modify criteria for evaluation of evidence items of an assurance project
Rationale	The evidence items of an assurance project shall be evaluated considering the most suitable set of criteria for “measuring” if and to what extent an evidence item can be eligible for (re)use both within the same domain (in a new project) and across different domains. From the moment that this activity is intrinsically dynamic, it is not so rare that the evaluation criteria to be used for the evaluation of the available evidence items shall be modified, deleted, or upgraded to match the needs.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.3.6 Reuse of evidence item evaluation criteria

03_06	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to evaluate evidence items of an assurance project according to evaluation criteria defined in other projects
Rationale	The criteria to be used for the evaluation of the evidence items of a project shall not be created from scratch any time, because of the level of similarity of some projects. Some criteria are applicable to the majority of the evidence items, independently from the peculiarities of a project. As a consequence, it is important to implement an efficient way to manage the evaluation criteria, so as to re-use those of general applicability.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.7 Definition of categories for evidence item evaluation criteria

03_07	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to define evaluation categories for evaluation criteria of evidence items of an assurance project
Rationale	From the moment that the evaluation criteria are linked among them, it is possible to define relationships between them and to group them into categories, so as that whenever a specific criteria is chosen for the evaluation of the evidence items of a project, all the related criteria are automatically identified, so as to make it easier for the user to select a complete set of evidence criteria without omitting necessary evaluation criteria.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.8 Modification of categories for evidence item evaluation criteria

03_08	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to modify the evaluation categories defined for evaluation criteria of evidence items of an assurance project
Rationale	The selection of the criteria for evaluating the evidence items of a project is an intrinsically dynamic activity that implies real-time updates. In the light of this fact, the categories into which these criteria can be grouped is consequently prone to modifications, updates etc.; consequently, it becomes important for the user to have at disposal a means for a dynamic management of all the categories of evaluation criteria, in order to adopt them to the needs.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.3.9 Reuse of categories for evidence item evaluation criteria

03_09	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to evaluate evidence items of an assurance project with evaluation categories for evaluation criteria defined in other projects
Rationale	The criteria to be used for the evaluation of the evidence items of a project shall not be created from scratch any time, but can be re-used from a project an applied to another. As a consequence, it is important to implement an efficient way to manage the evaluation criteria and the corresponding categories into which they can be grouped, so as to re-use those of general applicability and take advantage of well-established categories, applicable to several projects.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.10 Modification of evidence item evaluation

03_10	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to modify the evaluation of the evidence items of an assurance project, tracking the history of the changes
Rationale	The evaluation of any evidence item is intrinsically prone to updates and changes; as a consequence, it is necessary to provide the user the possibility to modify the evaluation of any evidence item, tracking the history of changes
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.11 Specification of required actions after evidence evaluation

03_11	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to specify actions required in the body of evidence of an assurance project as a result of evidence evaluation
Rationale	The evaluation of any evidence item is aimed at specifying actions impacting the evidence item. Additional tests can be recommended in order to verify the reaction of the system taken into account to specific test conditions and/or to provide a complete and exhaustive set of evidences in support of a safety claim. In addition to this, in several cases the evaluation of an evidence item is followed by a list of requested actions to be performed in order to increase the confidence in the evidence (test results etc.). As a consequence, it is necessary to have the possibility to associate to any evidence item the requested actions as a result of the evaluation process.
Stakeholders	Process Manager, Assurance Manager, Product Engineer
Status	Proposed
Priority	Must-have

3.3.3.12 Email for required actions after evidence evaluation

03_12	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to send emails containing information about actions to perform in an assurance project as a result of evidence evaluation
Rationale	In order to guarantee an efficient strategy for the management of the required actions associated to any evidence item contained in an assurance project, the platform could offer the user the possibility to specify the name(s) of the persons in charge of performing the requested actions for every evidence item and to directly inform them of the actions of which they are responsible by e-mail.
Stakeholders	Process Manager
Status	Proposed
Priority	Could-have

3.3.3.13 Evidence item associated to a required actions after evidence evaluation

03_13	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate the evidence items associated to actions required in the body of evidence of an assurance project as a result of an evidence evaluation
Rationale	The evaluation of any evidence item is aimed at specifying actions impacting the evidence item. Additional tests can be recommended in order to verify the reaction of the system taken into account to specific test conditions and/or to provide a complete and exhaustive set of evidences in support of a safety claim. In addition to this, in several cases the evaluation of an evidence item is followed by a list of requested actions to be performed in order to increase the confidence in the evidence (test results etc.). As a consequence, it is necessary to have the possibility to associate to any evidence item the requested actions as a result of the evaluation process.
Stakeholders	Process Manager, Assurance Manager, Product Engineer
Status	Proposed
Priority	Must-have

3.3.3.14 Use of colours in evidence items associated to a required actions after evidence evaluation

03_14	
Type	Non-functional (Usability)
Description	When an evidence item of an assurance project has been associated to actions to perform as a result of evidence evaluation, the OPENCOSS platform shall colour such an evidence item in red
Rationale	In order to make it easier for the user to see at glance the evidence items for which some actions have been defined, the platform should automatically highlight them through the use of colours.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Should-have

3.3.3.15 Status of actions related to evidence evaluation

03_16	
Type	Functional
Description	The OPENCOSS platform shall record if the status of an action required in an assurance project as a result of evidence evaluation is “pending” or “addressed”
Rationale	From the moment that the actions required for any evidence item are assigned to a responsible person, the platform shall offer the possibility to assign a status to every specific action, so as to have the possibility to monitor their progress.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Must-have

3.3.3.16 Display of required actions related to evidence evaluation

03_15	
Type	Functional
Description	The OPENCOSS platform shall show the actions required in an assurance project as a result of evidence evaluation
Rationale	It is necessary to specify for every evidence item contained in an assurance project the associated actions, with a clear indication for each of them of the responsible person. To this regard, the platform shall offer the possibility to display a summary report of all the actions required for the evidence items contained in an assurance project, plus the indications of a pre-defined set of attributes for any of these actions (name(s) of the responsible persons, status of the action, etc.)
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Must-have

3.3.3.17 Completion of actions required as a result of evidence evaluation

03_17	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate if an action required in an assurance project as a result of evidence evaluation has been addressed
Rationale	From the moment that the actions required for any evidence item are assigned to a responsible person, the platform shall offer the possibility to assign a status to every specific action, so as to have the possibility to monitor their progress.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Must-have

3.3.3.18 Use of colours to report on the status of actions required after evidence evaluation

03_18	
Type	Non-functional (Usability)
Description	When the status of an action required in an assurance project as a result of evidence evaluation is “pending”, the OPENCOSS platform shall colour such an action in red
Rationale	The platform shall offer the possibility to assign a status (pending or complete) to every action associated to an evidence item. In order to make it easier for the user to see at glance the status of the defined actions, the platform should automatically highlight them through the use of different colours.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Should-have

3.3.3.19 Report on evidence set completeness

03_19	
Type	Functional
Description	When all the evidence items to provide in an assurance project has been provided according to the evidence characterization model of the project, the OPENCOSS platform shall indicate that the body of evidence of an assurance project can be regarded as complete
Rationale	From the moment that the reference norms usually prescribe the evidence items to be produced as well as the structure of the safety case, the platform shall be capable of automatically detect whether all the required set of evidences have been inserted by the user (quantitative check for completeness)
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.3.20 Report on evidence set adequacy

03_20	
Type	Functional
Description	When the body of evidence of an assurance project is regarded as complete, all the evidence traceability links of the project are correct, there are no pending actions in the project, and the information imported from external tools in the project is synchronized, the OPENCOSS platform shall indicate that the body of evidence of an assurance project can be regarded as adequate
Rationale	The platform shall be capable of performing an automatic check that the evidence set matches the types of evidence prescribed/recommended by the standard through an analysis of the attributes of every evidence item (evidence characterisation).
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.4 Functional Area 4: Evidence Change Impact Analysis

The following 15 component level requirements have been specified for evidence change impact analysis in the OPENCOSS platform.

3.3.4.1 Determination of evidence items affected by a change

04_01	
Type	Functional
Description	When an evidence item is changed in an assurance project, the OPENCOSS platform shall indicate the evidence items of the assurance project affected by the change
Rationale	From the moment that it is very common for an evidence item to be directly and/or indirectly dependent on other evidences, the platform shall be capable of automatically perform what-if analyses aimed at providing the user with a clear picture of all the evidence items that are affected by any change to a specific evidence item.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.4.2 Determination of evidence traceability links affected by a change

04_02	
Type	Functional
Description	When an evidence item is modified in an assurance project, the OPENCOSS platform shall indicate the evidence traceability links of the assurance project affected by the modification
Rationale	From the moment that it is very common for an evidence item to be directly and/or indirectly traced to other evidences, the platform shall be capable of automatically identify those evidence traceability links that are affected by a change to a specific evidence
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.4.3 Confirmation of changes

04_03	
Type	Functional
Description	When an evidence item is changed in an assurance project and other evidence items are affected, the OPENCOSS platform shall require confirmation of the changes from the user
Rationale	Any time a change to an evidence item is done, the platform shall automatically detect all the other evidence items impacted by this change. Anyway, it is not possible for the platform to automatically accept all the changes, but a confirmation from the user is requested by the platform. As a consequence, the platform is a supporting tool for the identification of the evidence items affected by a change in a specific evidence item, but the acceptance/refusal of the consequences of this change is assigned to the user.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.4.4 Impact of a possible evidence change

04_05	
Type	Functional
Description	When a user has indicated a possible change in an evidence item of an assurance project, the OPENCOSS platform shall indicate the evidence items of the assurance project that would be affected by the change
Rationale	From the moment that any change to an evidence item can have a relevant impact on the whole chain of evidences of a project, the platform shall provide the user an automatic analysis of all the evidence items that would be affected by a change to a specific evidence item. This feature is important to perform what-if analyses on the impact of a change to an evidence item and to provide information for its efficient management.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Must-have

3.3.4.5 Specification of actions required after evidence item changes

04_07	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to specify actions required in the body of evidence of an assurance project as a result of an evidence item change
Rationale	From the moment that any change to an evidence item can have a relevant impact on the other evidences and on the rest of the safety case in general, the platform shall offer the user the possibility to specify the list of requested actions to be performed, providing for each of them a justification and linking them to the specific evidence item to which they apply (1:n relationship)
Stakeholders	Assurance Manager, Process Manager, Product Engineer
Status	Proposed
Priority	Must-have

3.3.4.6 Evidence items associated to actions required after evidence item changes

04_08	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate the evidence items associated to actions required in the body of evidence of an assurance project as a result of an evidence item change
Rationale	From the moment that any change to an evidence item can have a relevant impact on the other evidences and on the rest of the safety case in general, the platform shall offer the user the possibility to specify the list of requested actions to be performed, providing for each of them a justification and linking them to the specific evidence item to which they apply (1:n relationship)
Stakeholders	Assurance Manager, Process Manager, Product Engineer
Status	Proposed
Priority	Must-have

3.3.4.7 Evidence traceability links associated to actions required after evidence item changes

04_09	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate the evidence traceability links associated to actions required in the body of evidence of an assurance project as a result of an evidence item change
Rationale	From the moment that any change to an evidence item can have a relevant impact on the other evidences and on the rest of the safety case in general, the platform shall offer the user the possibility to specify the list of requested actions to be performed, providing for each of them a justification and linking them to the specific evidence traceability links to which they apply (1:n relationship)
Stakeholders	Assurance Manager, Process Manager, Product Engineer
Status	Proposed
Priority	Must-have

3.3.4.8 Suggestion of actions after evidence item change

04_10	
Type	Functional
Description	The OPENCOSS platform shall suggests actions to perform after modification of an evidence item of an assurance project is approved
Rationale	Any time an evidence item is changed, a set of actions shall be performed. Some of them are strictly dependent on the specific evidence item that has been modified, while others are more general and are recommended/prescribed by the reference safety norms applying. As a consequence, the platform shall inform the user of the actions to be performed in compliance to what stated in the reference safety documentation applicable to the project.
Stakeholders	Assurance Manager, Process Manager, Product Engineer
Status	Proposed
Priority	Must-have

3.3.4.9 Email with required actions after evidence item changes

04_11	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to send emails containing information about actions to perform in an assurance project as a result of evidence item changes
Rationale	In order to guarantee an efficient strategy for the management of the required actions associated to a change in any evidence item contained in an assurance project, the platform could offer the user the possibility to specify the name(s) of the persons in charge of performing the requested actions for every evidence item and to directly inform them of the actions of which they are responsible by e-mail.
Stakeholders	Process Manager
Status	Proposed
Priority	Could-have

3.3.4.10 Status of actions related to evidence change

04_13	
Type	Functional
Description	The OPENCOSS platform shall record if the status of an action required in an assurance project as a result of the change of an evidence item is “pending” or “addressed”
Rationale	From the moment that the actions required for any evidence item are assigned to a responsible person, the platform shall offer the possibility to assign a status to every specific action, so as to have the possibility to monitor their progress.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Must-have

3.3.4.11 Display of required actions after evidence item changes

04_12	
Type	Functional
Description	The OPENCOSS platform shall show the actions required in an assurance project as a result of evidence item changes
Rationale	It is necessary to specify for every evidence item contained in an assurance project the associated actions, with a clear indication for each of them of the responsible person. To this regard, the platform shall offer the possibility to display a summary report of all the actions required for the evidence items contained in an assurance project, plus the indications of a pre-defined set of attributes for any of these actions (name(s) of the responsible persons, status of the action etc.)
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Must-have

3.3.4.12 Completion of actions required after evidence item changes

04_14	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to indicate if an action required in an assurance project as a result of the change of an evidence item has been addressed
Rationale	From the moment that the actions required for any evidence item are assigned to a responsible person, the platform shall offer the possibility to assign a status to every specific action, so as to have the possibility to monitor their progress.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Must-have

3.3.4.13 Use of colours to report on the status of actions required after evidence item changes

04_15	
Type	Non-functional (Usability)
Description	When the status of an action required in an assurance project as a result of evidence item changes is “pending”, the OPENCOSS platform shall colour such an action in red
Rationale	The platform shall offer the possibility to assign a status (pending or complete) to every action associated to an evidence item. In order to make it easier for the user to see at glance the status of the defined actions, the platform should automatically highlight them through the use of different colours.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Should-have

3.3.4.14 Use of colours to report on the status of evidence items after evidence item changes

04_16	
Type	Non-functional (Usability)
Description	When an evidence item of an assurance project has been affected by evidence item changes, the OPENCOSS platform shall colour such an evidence item in red
Rationale	In order to make it easier for the user to see at glance the status of the evidence items after a change, the platform should automatically highlight them through the use of colours.
Stakeholders	Assurance Manager, Product Engineer, Process Manager
Status	Proposed
Priority	Should-have

3.3.4.15 Use of colours to report on the status of evidence traceability links after evidence item changes

04_17	
Type	Non-functional (Usability)
Description	When an evidence traceability link of an assurance project has been affected by evidence item changes, the OPENCOSS platform shall colour such a link in red
Rationale	In order to make it easier for the user to see at glance the status of evidence traceability links after a change, the platform should automatically highlight them through the use of colours.
Stakeholders	Assurance Manager
Status	Proposed
Priority	Should-have

3.3.5 Functional Area 5: Integration with External Tools

The following 21 component level requirements have been specified for integration with external tools in the OPENCOSS platform.

3.3.5.1 Evidence item information import

05_01	
Type	Functional
Description	The OPENCOSS platform shall provide users with the ability to import evidence items for an assurance project from artefacts in an external tool
Rationale	There are various engineering tools used in safety-critical project development, storing information that can be considered as safety evidence. The OPENCOSS platform's aim is not to replace those tools, but integrate with them and import the tools artefacts data that can be used as safety evidence.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Must-have

3.3.5.2 Automatic artefact association to imported evidence item

05_02	
Type	Functional
Description	The OPENCOSS platform shall be able to automatically associate an artefact in an external tool to an evidence item of an assurance project imported from the external tool
Rationale	The OPENCOSS platform manages evidence items information, but not the actual artefacts from which this information has been derived. The artefacts are in external tool. Users need to have backward traceability pointing from evidence instance in OPENCOSS platform to the external tool artefact from which the evidence information has been extracted and imported.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.3 Tool information about the artefact associated to imported evidence item information

05_03	
Type	Functional
Description	The OPENCOSS platform shall be able to automatically indicate the tool that stores the artefact associated to an evidence item of an assurance project imported from external tool
Rationale	The OPENCOSS platform manages evidence items information, but not the actual artefacts from which this information has been derived. The artefacts are in external tool. Users need to have backward traceability pointing from evidence instance in OPENCOSS platform to the external tool.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.4 Evidence traceability link import

05_04	
Type	
Description	The OPENCOSS platform shall provide users with the ability to import evidence traceability links for an assurance project from an external tool
Rationale	The OPENCOSS platform aim is not to completely replace engineering tools that facilitate artefact traceability, but integrate with them and import the traceability information from them.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.5 Automatic artefact association to imported evidence traceability link

05_05	
Type	Functional
Description	The OPENCOSS platform shall be able to automatically associate an artefact to an evidence traceability link of an assurance project imported from an external tool
Rationale	The OPENCOSS platform could import traceability links from external tool. Users need to have a backward link from OPENCOSS to the external artefact in order to easily browse back to the external tool artefact data.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.6 Tool information about the artefact associated to imported evidence traceability link

05_06	
Type	Functional
Description	The OPENCOSS platform shall be able to automatically indicate the tool that stores the artefact associated to an evidence traceability link of an assurance project imported from an external tool
Rationale	The OPENCOSS platform could import traceability links from external tool. Users need to have a backward link from OPENCOSS to the external tool.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.7 Evidence traceability link export

05_07	
Type	Functional
Description	The OPENCOSS platform shall be able to export evidence traceability links of an assurance project to external tools
Rationale	The OPENCOSS platform will be able to import evidence data from various engineering tools and store them in a common model. It would be good that this common data can be exported into a common format so that it can it can be further processed or visualized by external tools.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.8 SAEM import

05_08	
Type	Non Functional (Standard Compliance)
Description	The OPENCOSS shall be able to import evidence for an assurance project from SAEM models
Rationale	SAEM defines a fine-grained model for evidence item description. The OPENCOSS platform could import evidence data from any tool that supports SAEM.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.9 SAEM export

05_09	
Type	Non Functional (Standard Compliance)
Description	The OPENCOSS platform shall be able to export the evidence of an assurance project in the form of a SAEM models
Rationale	SAEM defines a fine-grained model for evidence item description. The OPENCOSS platform could export evidence data in SAEM so that it can be fed into any tool that supports SAEM.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.10 Automatic update of evidence items in the OPENCOSS platform

05_10	
Type	Functional
Description	When an evidence item of an assurance project contains information imported from an external tool artefact, and the information in the external tool is modified, the OPENCOSS platform shall detect such a modification and update the evidence item accordingly
Rationale	Evidence item data in the OPENCOSS platform is derived from external tool artefacts data. In typical projects there will be a lot of evidence items, thus it is crucial to have the data updated automatically in OPENCOSS when they change in the external tool.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.11 Automatic update of evidence items in external tools

05_11	
Type	Functional
Description	When an evidence item of an assurance project contains information imported from an external tool, and the information in the OPENCOSS platform is modified, the OPENCOSS platform shall detect such a modification and update the information in the external tool accordingly
Rationale	In typical projects there will be a lot of evidence items, thus it would be a nice feature to have the data updated automatically in external tools.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.12 Notification of change of imported evidence item in the OPENCOSS platform

05_12	
Type	Functional
Description	When an evidence item of an assurance project contains information imported from an external tool, and the information in the OPENCOSS platform is modified, the OPENCOSS platform shall indicate such a modification
Rationale	In typical projects there will be a lot of evidence items and transparency in safety-critical projects is crucial for assessment, thus having a change-notification mechanism and keeping change log history is a very important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Must-have

3.3.5.13 Notification of change of imported evidence item in external tool

05_13	
Type	Functional
Description	When an evidence item of an assurance project contains information imported from an external tool, and the information in the external tool is modified, the OPENCOSS platform shall indicate such a modification
Rationale	In typical projects there will be a lot of evidence items, and transparency in a safety-critical projects is crucial for assessment, thus having a change-notification mechanism and keeping change log history is a very important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.14 Use of colours to indicate change of imported evidence item in the OPENCOSS platform

05_14	
Type	Non-functional (Usability)
Description	When an evidence item of an assurance project contains information imported from an external tool, and the information in the OPENCOSS platform is modified, the OPENCOSS platform shall indicate such a modification by colouring the evidence item in red
Rationale	In typical projects there will be a lot of evidence items, thus having a good visualization of the changed data is an important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Must-have

3.3.5.15 Use of colours to indicate change of imported evidence item in external tool

05_15	
Type	Non-functional (Usability)
Description	When an evidence item of an assurance project contains information imported from an external tool, and the information in the external tool is modified, the OPENCOSS platform shall indicate such a modification by colouring the evidence item in red
Rationale	In typical projects there will be a lot of evidence items, thus having a good visualization of the changed data is an important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.16 Automatic update of evidence traceability links in the OPENCOSS platform

05_16	
Type	Functional
Description	When an evidence traceability link of an assurance project contains information imported from an external tool, and the information in the external tool is modified, the OPENCOSS platform shall detect such a modification and update the evidence traceability link accordingly
Rationale	In typical projects there will be a lot of evidence traceability links, thus it is crucial to have the data updated automatically in OPENCOSS when they change in the external tool.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.17 Automatic update of evidence traceability links in external tools

05_17	
Type	Functional
Description	When an evidence traceability link of an assurance project contains information imported from an external tool, and the information in the OPENCOSS platform is modified, the OPENCOSS platform shall detect such a modification and update the information in the external tool accordingly
Rationale	In typical projects there will be a lot of evidence traceability links, thus it is would be a nice feature to have the data updated automatically in external tool when they are changed in OPENCOSS platform.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Could-have

3.3.5.18 Notification of change of imported evidence traceability link in the OPENCOSS platform

05_18	
Type	Functional
Description	When an evidence traceability link of an assurance project contains information imported from an external tool, and the information in the OPENCOSS platform is modified, the OPENCOSS platform shall indicate such a modification
Rationale	In typical projects there will be a lot of evidence items, and transparency in a safety-critical projects is crucial for assessment, thus having a change-notification mechanism and keeping change log history is a very important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Must-have

3.3.5.19 Notification of change of imported evidence traceability link in external tool

05_19	
Type	Functional
Description	When an evidence traceability link of an assurance project contains information imported from an external tool, and the information in the external tool is modified, the OPENCOSS platform shall indicate such a modification
Rationale	In typical projects there will be a lot of evidence traceability links, and transparency in a safety-critical projects is crucial for assessment, thus having a change-notification mechanism and keeping change log history is a very important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.20 Use of colours to indicate change of imported evidence traceability links in the OPENCOSS platform

05_20	
Type	Non-functional (Usability)
Description	When an evidence traceability link of an assurance project contains information imported from an external tool, and the information in the OPENCOSS platform is modified, the OPENCOSS platform shall indicate such a modification by colouring the evidence traceability link in red
Rationale	In typical projects there will be a lot of evidence traceability links, thus having a good visualization of the changed data is an important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.5.21 Use of colours to indicate change of imported evidence traceability link in external tools

05_21	
Type	Non-functional (Usability)
Description	When an evidence traceability link of an assurance project contains information imported from an external tool, and the information in the external tool is modified, the OPENCOSS platform shall indicate such a modification by colouring the evidence traceability link in red
Rationale	In typical projects there will be a lot of evidence traceability links, thus having a good visualization of the changed data is an important functionality.
Stakeholders	Engineering Tool, Assurance Manager, Safety Engineer
Status	Proposed
Priority	Should-have

3.3.6 User Interface Mock-Ups

In addition to the requirements textually specified above for the five functional areas of the OPENCOSS platform related to evidence management, several user interface mock-ups have been created as part of the specification of component level requirements for evidence management. The figures below show mock-ups for: (1) a dashboard for evidence-related actions to perform (Figure 5); (2) a dashboard for evidence item information (Figure 6); (3) matrix-based traceability display (Figure 7); (4) model-based traceability display (Figure 8), and; (5) table-based traceability display (Figure 9)

The mock-ups are initial proposals about how the OPENCOSS platform could provide users with evidence-related information, and will evolve when requirements are refined and integration with the requirements specified in other WPs is addressed.

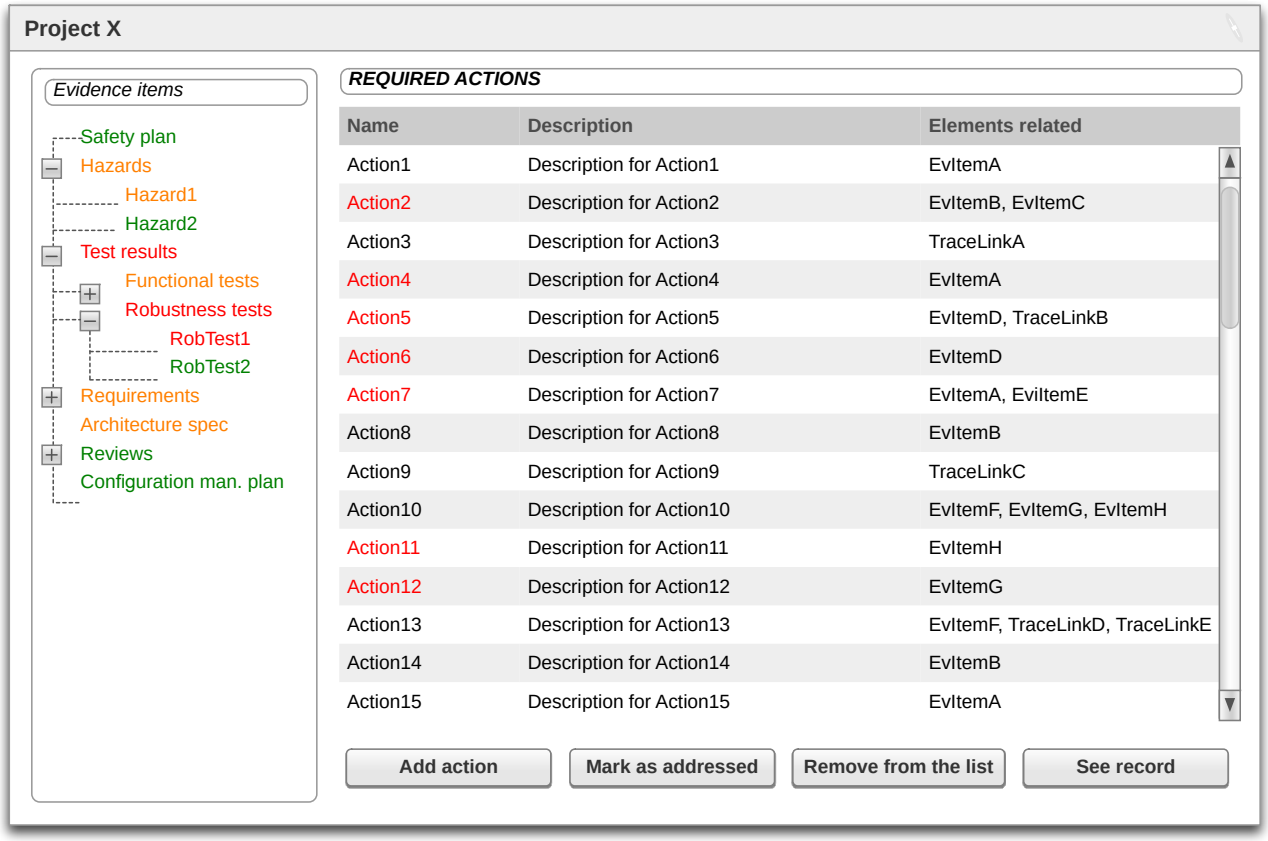


Figure 5. Main dashboard: required actions view

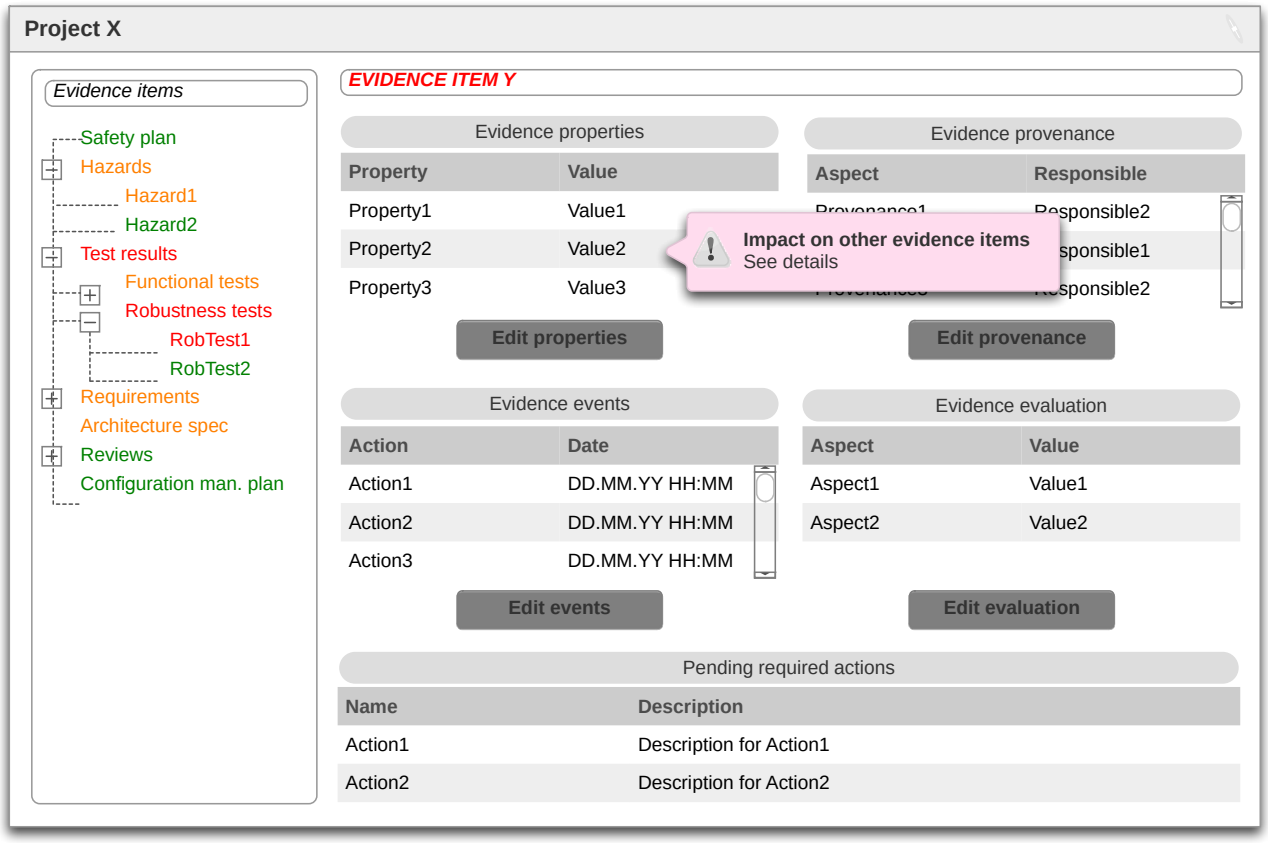


Figure 6. Main dashboard: evidence item view

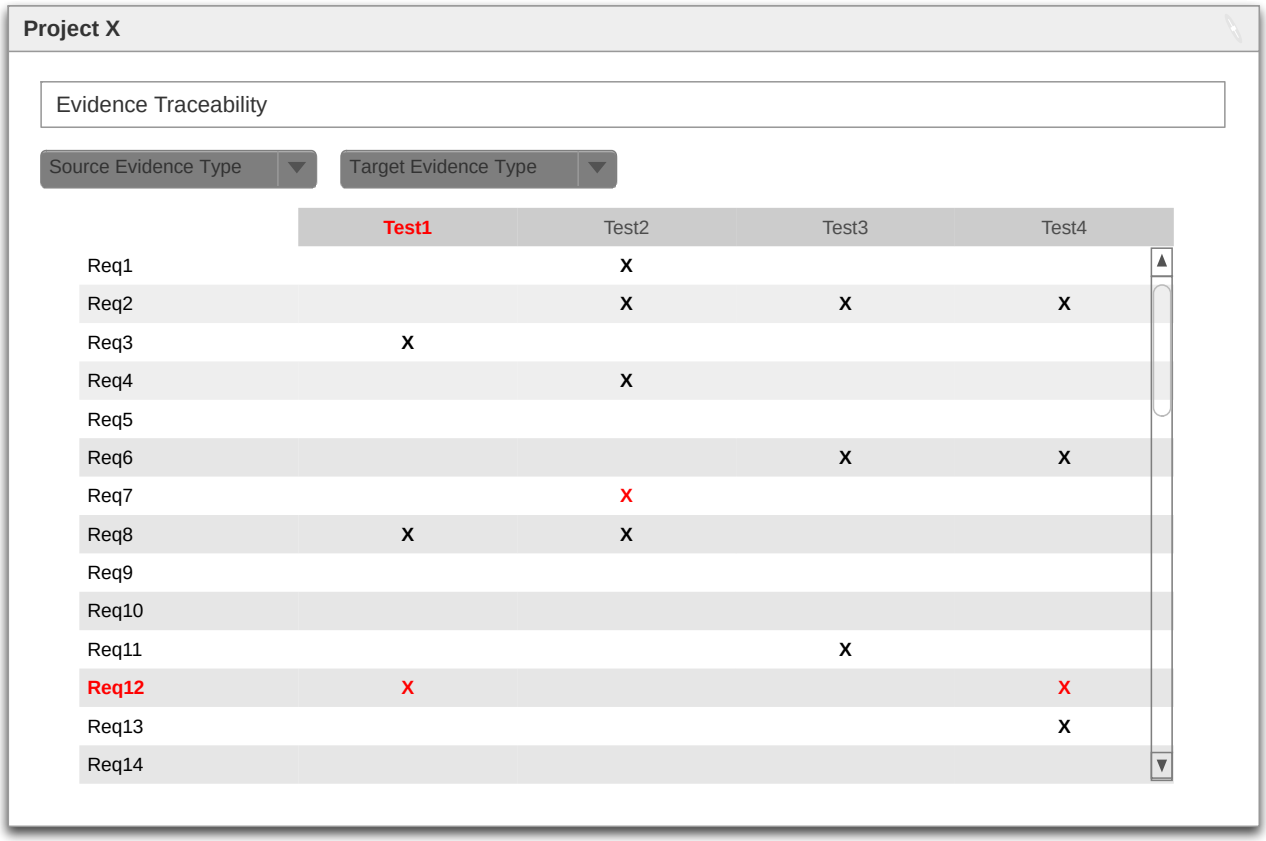


Figure 7. Matrix-based traceability

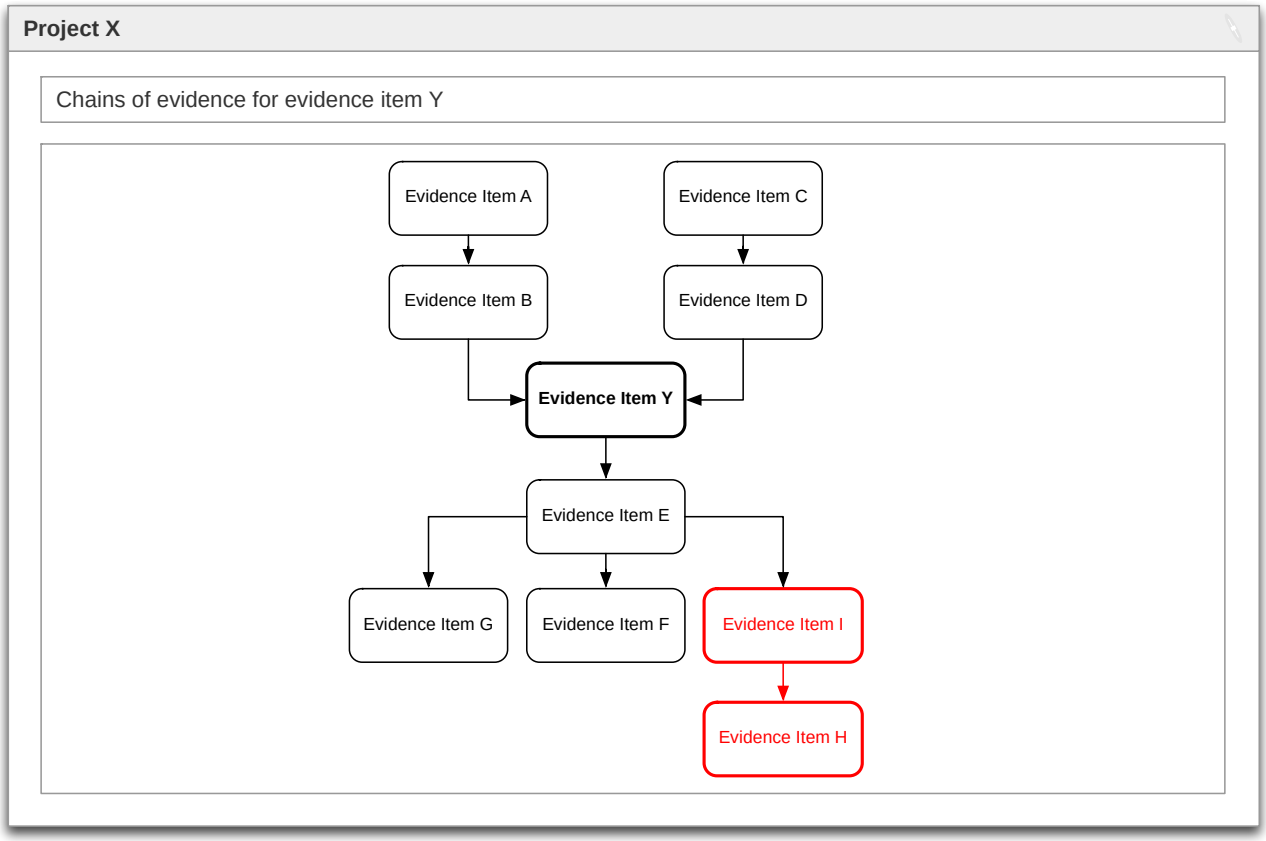


Figure 8. Model-based traceability

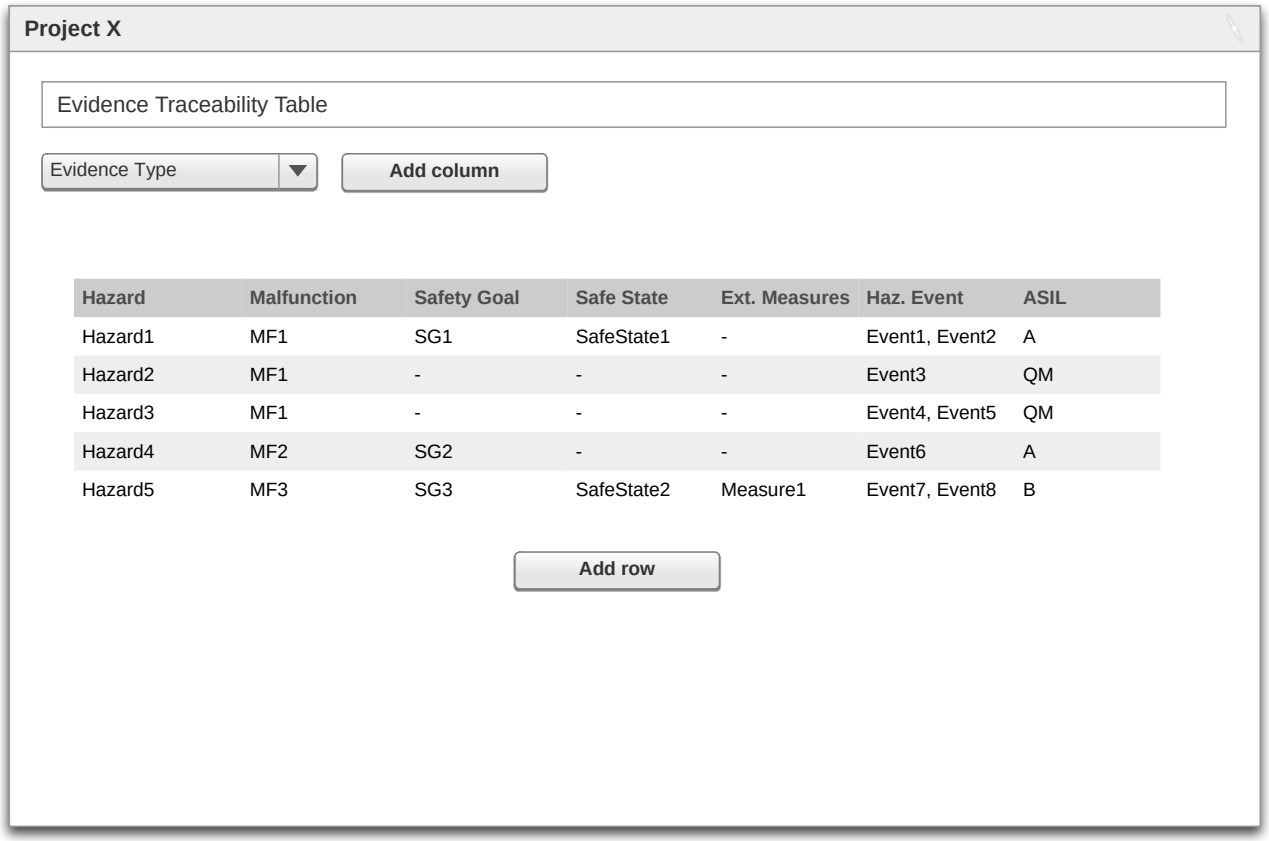


Figure 9. Table-based traceability

4 Conclusions

D6.2 has provided the current set of detailed requirements for evidence management of the OPENCOSS platform. These requirements will be the basis for the design of the evidence management service infrastructure in T6.2, and will also be used as input for specification of integration requirements and test plans in T3.2 and for the certification language conceptual model in T4.2.

Specification of detailed requirements for evidence management has been based on the analysis of:

- Situations in which evidence evolves, which indicate issues related to evidence management that practitioners face and must address.
- The role of the CCL in evidence management, which constrains how some evidence-related aspects of an assurance project must be addressed in the OPENCOSS platform.
- The management of evidence and artefacts in the OPENCOSS platform, which defines how evidence-related information will be managed.
- The review of existing tools for evidence management, which has allowed us to determine functionality that the OPENCOSS platform should or should not provide, as well as possible improvements in existing tools.
- A survey on the state of the practice concerning evidence management, which indicates details that must be considered in OPENCOSS for adequately supporting practitioners.
- Other OPENCOSS deliverables, which explicitly or implicitly impose requirements for evidence management.
- The high-level business process for evidence management in an assurance project, which outlines the related activities to support by the OPENCOSS platform.
- The product level and feature level requirements for evidence management, which correspond to stakeholders' goals and system features that can allow their achievement.
- The use cases of the OPENCOSS platform, which specify interactions between the OPENCOSS platform and its users for evidence management.

From this analysis and the input from OPENCOSS partners, 111 detailed requirements have been specified for evidence management of the OPENCOSS platform, and they have assigned to one of the main functional areas identified: evidence storage, evidence traceability, evidence evaluation, evidence change impact analysis, and integration with external tools. In addition, several initial user interface mock-ups have been created.

In relation to the current and future work to perform in WP6 and other WPs, the aspects related to evidence management that must be further studied and carefully addressed are:

- Advanced support for evidence traceability, aiming to investigate functionality that can make evidence traceability specification more efficient; examples of functionality can be trace discovery, trace suggestions, and provision of specialized traceability information models for assurance projects.
- Advanced evidence change impact analysis, aiming to investigate in depth the possible effects that evidence changes can have in the body of evidence of an assurance project, how the effect should be propagated, what evidence characteristics are relevant for impact analysis, and when user's feedback would be required for decision making regarding impact analysis.
- Integration with external tools, aiming to study how external tools can and should be integrated with the OPENCOSS platform, and possible issues that may arise from the integration (e.g., security aspects in external tools).
- Relation with CCL and with compositional certification conceptual framework, aiming to determine how they constrain and relate with evidence management, and thus to discover new requirements that should be included in future updates of this deliverable.

- Further elaboration of user interface mock-ups, aiming to both refine the mock-ups provided in this deliverable and create new ones.
- Further automation of system behaviour, aiming to study to what extent the OPENCOSS platform should automatically execute certain actions related to evidence management (e.g., change impact propagation), ask users for approval, or suggests actions that could guide platform usage.
- Integration with the work and results of WP7, aiming to determine how the information managed for transparent certification and compliance-aware process can be used as evidence in an assurance project, and how the evidence can be used as input for transparent certification and compliance-aware process.

References

[1] Alexander, I.F., Maiden, N. (eds.): Scenarios, Stories, Use Cases. John Wiley and Sons (2004)

[2] Bohner, S.A., Arnold, R.S. (eds.): Software Change Impact Analysis. IEEE Computer Society Press (1996)

[3] Cleland-Huang, J., Gotel, O., Zisman, A. (eds.): Software and Systems Traceability. Springer (2012)

[4] Cleland-Huang, J., Heimdahl, M., Hayes, J.H., Lutz, R., Maeder, P.: Trace Queries for Safety Requirements in High Assurance Systems. In: REFSQ2012, LNCS 7195, pp 179-193. Springer (2012)

[5] de la Vara, J.L.: Business process-based requirements specification and object-oriented conceptual modelling of information systems. PhD thesis, Universidad Politécnica de Valencia (2011)

[6] Gorscheck, T., Wohlin, C.: Requirements Abstraction Model. Requirements Engineering 11(1): 79-101 (2006)

[7] Lauesen, S., Harning, M.B.: Virtual Windows: Linking User Tasks, Data Models, and Interface Design. IEEE Software 20(2): 58-65 (2003)

[8] Mavin, A.: Listen, Then Use EARS. IEEE Software 29(2): 17-18 (2012)

[9] OMG: Business Process Model and Notation (BPMN) Specification v.1.2 (online) http://www.bpmn.org (2009)

[10] OMG: Software Assurance Evidence Metamodel (SAEM) 1.0 – Beta 1 (online) http://www.omg.org/spec/SAEM/ (2010)

[11] OMG: Unified Modeling Language: Superstructure Version 2.0 (online) http://www.uml.org (2005)

[12] OPENCOSS: Deliverable D3.1 - Analysis of safety certification data of industrial use cases (2012)

[13] OPENCOSS: Deliverable D6.1 - Baseline for the evidence management needs of the OPENCOSS platform (2012)

[14] Pohl, K.: Requirements Engineering. Springer (2010)

[15] Rolland, C.: Capturing System Intentionality with Maps. In: Conceptual Modelling in Information Systems Engineering, pp 141-158. Springer (2007)

[16] Sommerville, I.: Integrated Requirements Engineering: A Tutorial. IEEE Software 22(1): 16-23 (2005)

Appendix A. Glossary

This appendix defines a set of terms that have been used in the D6.2, have not been defined in the glossaries of previous OPENCOSS deliverables (e.g., D2.2 and D4.2), and whose semantics might be ambiguous or difficult to understand.

Chain of evidence
Definition: series of pieces of evidence that are related, so traceability between the pieces exists.
Synonyms: evidence chain, evidential chain

Evidence item
Definition: information used as evidence in assurance project and stored in the OPENCOSS platform.
Synonyms: piece of evidence

Evidence set
Definition: Set of evidence items and their associated information.
Synonyms: body of evidence, set of evidence

Evidence traceability link type
Definition: Category of traceability that can exist between evidence items of an assurance project and that is based on traceability that can exist between evidence items of certain evidence types; for example, an evidence traceability link type can refer to traceability between requirements and test cases.

Evidence type
Definition: Category of information or artefacts that can be used as evidence in an assurance project; for example, safety requirement.

Impact Analysis
Definition: identification of the potential consequences of a change, or estimation of what needs to be modified to accomplish a change [2]
Synonyms: change impact analysis

Traceability
Definition: the ability to define and trace relationships among entities (e.g., software work products and their components) [2]

Tree-view
Definition: graphical user interface element that presents a hierarchical view of information; each item (aka branch or node) can have a number of subitems; this is often visualized by indentation in a list; an item can be expanded to reveal subitems, if any exist, and collapsed to hide subitems.

Appendix B. Overall Approach for Requirements Specification

This appendix presents an overview of the approach followed for specification of detailed requirements for evidence management of the OPENCOSS platform.

First of all, we must provide a definition of requirement [5]:

> *Requirements are activities, capabilities or conditions, external to a system, that the system must support, possess or meet, respectively, to fulfil stakeholders' needs.*

As shown in Figure 10, the overall approach consists of 6 steps: (1) analyse the current situation; (2) determine product level requirements; (3) determine feature level requirements; (4) determine function level requirements; (5) determine component level requirements, and; (6) determine the impact of the system on the current situation. This process will very likely be iterative, as new needs are discovered and new insights are gained regarding, for instance, the application domain and its current situation. It must be noted that, among all the activities of the RE process [16], these steps mainly focus on requirements elicitation and requirements specification. The activities that are not explicitly addressed are requirements analysis, requirements negotiation, requirements management, and requirements management.

The approach proposed is mainly based on two existing research works that have also been applied in industry: the business process-based RE approach proposed in [5], and RAM [6], an approach for dealing with requirements at different abstraction levels. The approach proposed for specification of detailed requirements for evidence management of the OPENCOSS platform adapts and extends both works according to the specific needs of OPENCOSS.

The business process-based RE approach aims to specify system requirements of an IS so that the system supports and fits the business processes of the organization in which the IS will be introduced. Two of the problems identified in practice that led to the design of the approach were (1) the lack of knowledge about the application by system analysts, and consequently (2) the difficulties to meet the actual needs of an organization when developing an IS.

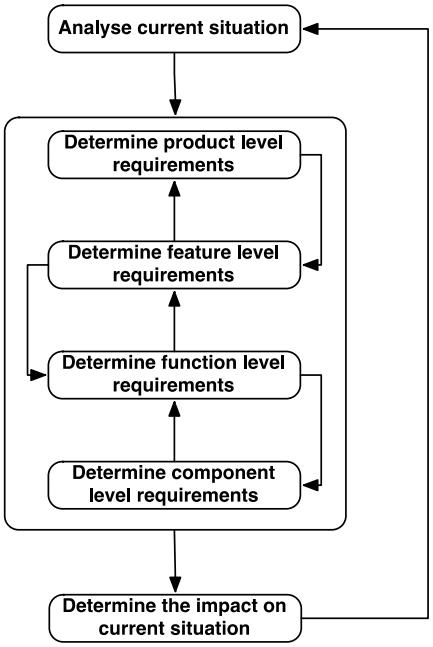


Figure 10. Overview of the approach for requirements specification

A business process is defined as follows [5]:

> *A business process is a set of structured and ordered activities that are performed in an organization to achieve some business goal. A business process takes inputs from the business environment and creates outputs, and is executed coordinately and dynamically by people and/or technical components that exchange information*

The approach proposes modelling of the current situation of an organization (As-Is BPDs) in order to understand the application domain and thus the problem(s) to solve or need(s) to meet. The business processes of the organization might have to be redesigned (To-Be BPDs) as a result of the development and introduction of the IS. System purpose (i.e., the business goals whose achievement will be possible thanks to the IS) drives business process redesign. That is, business process redesign might be (1) necessary in order to fulfil system purpose, and (2) enabled by the IS. Business processes are modelled with BPMN [9]. For evidence management of the OPENCOSS platform, To-Be BPDs have not included in D6.2.

Figure 11 shows an overview of the business process-based RE approach as a whole. It must be noted that it is not necessary to apply the approach as originally defined. First, the approach was designed in the context of a model-driven software process, thus requirements specification was linked to object-oriented modelling. Second, all the artefacts proposed do not need to be created always. For example, we can directly model (As-Is) business process without creating specific artefacts to specify information such as business rules and business events. Therefore, what is proposed is an adaptation of the overall business process-based RE approach to the specific needs of WP6 (and OPENCOSS).

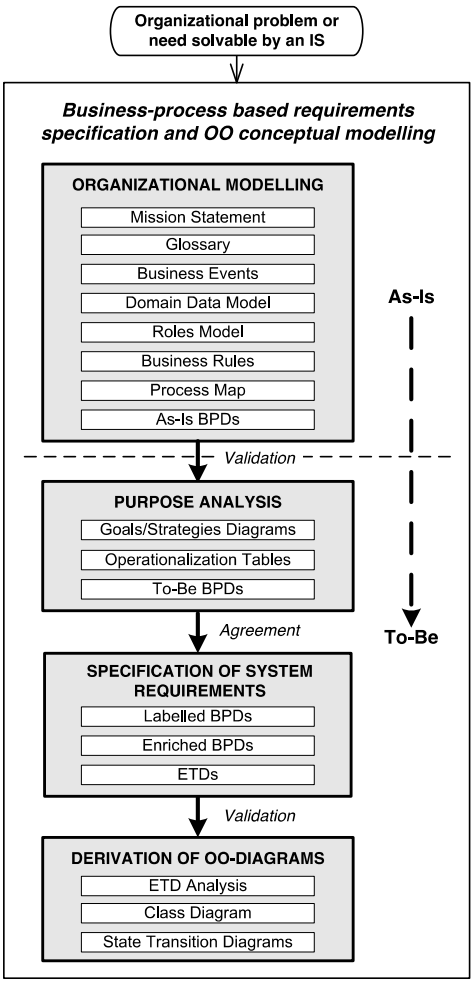


Figure 11. Stages and artefacts of the business process-based RE approach

RAM is a model targeted at placement of requirements on different abstraction levels and at supporting abstraction or break down of requirements in order to make them comparable to each other. The main problem identified in practice that led to the design of the model was the fact that requirements usually arrive at different abstraction levels. This poses problems in, for instance, requirements analysis and requirements management, influencing tasks such as requirements periodization and implementation planning.

RAM proposes the following 4 abstraction levels for requirements:

- Product level
  This level corresponds to the goals whose achievement will be possible thanks to the development of a system. In the context of OPENCOSS, goals correspond to the business weaknesses or problems to solve on the business needs to meet by means of the OPENCOSS platform (e.g., "Facilitate evidence combination"). Such problems and needs exist independently from the existence of the platform.
- Feature level
  This level corresponds to features that a system must support in order to meet the goals of the product level. In general, a system feature [5] is characterised by (1) representing an abstraction of the functionality of a system, (2) corresponding to a system characteristic that is valuable for customer stakeholders, and (3) not being testable (i.e., a feature must be refined or broken down in order to verify that a system supports it).
- Function level
  This level corresponds to the functions and actions that a user should be able to do. That is, requirements at this abstraction level should describe what users should be able to perform/do when using the system. As a rule of thumb, function level requirements are detailed and complete enough to kick-start system design. However, they are not detailed and complete enough to, for instance, allow two separate development teams to implement a same system (specification) and that the systems for both teams provide the same functionality and/or services.
- Component level
  This level corresponds to requirements specified in such a detailed and precise way that would allow the two developments used as an example above to implement two systems with (almost) the same functionality and/or services. Requirements at this level represent the boundary between requirements and design. It should be possible to assign the requirements specified at this level to the system components (architecture) that will provide such functionality and/or service.

Figure 12 shows the propose actions to apply RAM. Overall, it fits how requirements are being discovered and provided in OPENCOSS. We have several, different sources that provide requirements at different abstraction levels. It is important that we can determine why a requirement at a given level is necessary (abstraction), and how a requirement can be broken down if necessary.

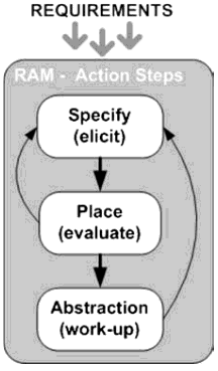


Figure 12. RAM action steps

The requirements specification styles proposed for each level are:

- Product level and Feature level: textual specification (list), partially supported by the Maps approach (for goal modelling) [15] to facilitate goal and feature discovery and analysis (Figure 13).
- Function level: uses cases [11].
- Component level: use case scenarios [1], textual specification of functional and non-functional requirements, and user interface mock-ups [7].

It must be indicated that function level requirements and use case scenarios for evidence management can be found in D2.3.

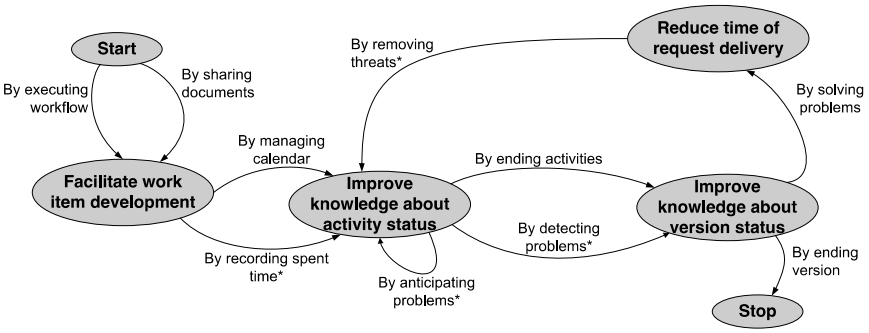


Figure 13. Example of Map diagram

For description of component level requirements as a part of their textual specification, we have used the text structure shown in Figure 14 [14]. It has also been complemented with the patterns proposed by the EARS approach [8]:

- Ubiquitous requirements: *The <system name> shall <system response>*
- Event-driven requirements: *When <optional preconditions> <trigger>, the <system> shall <system response>*
- State-driven requirements: *While <in a state>, the <system> shall <system response>*
- Unwanted behaviour: *If <optional preconditions> <trigger>, then the <system> shall <system response>*
- Optional requirements: *Where <feature>, the <system> shall <system response>*

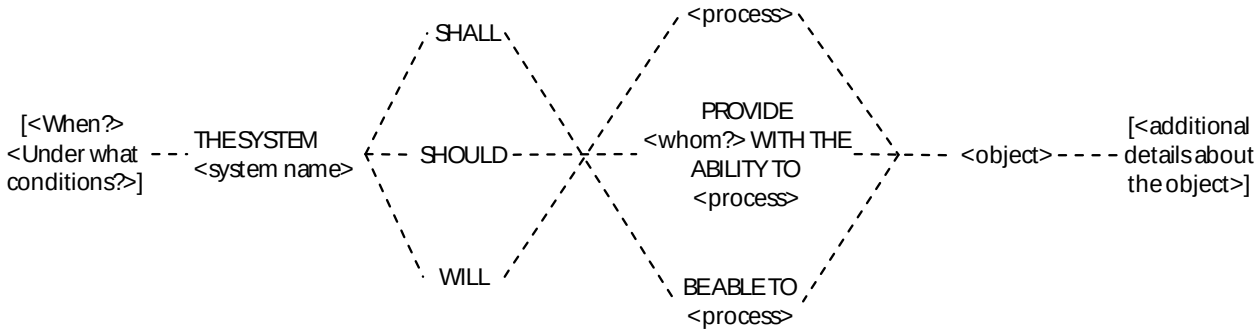


Figure 14. Text structure for requirements specification

Appendix C. Tools for Evidence Management

This appendix shows the existing tools for evidence management that have been reviewed in the scope of WP6 for requirements discovery. For each tool, the link(s) from which information was obtained and brief description are provided.

ACCESS
Information www.cs.virginia.edu/~pvs5x/publications/2011.ACCESS.pdf
Overall Description Toolset designed to support creation, inspection, validation, maintenance, and other activities related to GSN

Appraisal Assistant
Information http://www.sqi.gu.edu.au/AppraisalAssistant/about.html
Overall Description Tool for CMMI compliance management

Appraisal Wizard
Information http://isd-inc.com/tools.appraisalWizard/
Overall Description Tool for preparing and performing appraisals. Although its fundamentally made for Scampi appraisals, its functionality goes above and beyond CMMI-based appraisal

ASCE
Information http://www.adelard.com/asce/index.html
Overall Description Tool for development and management of assurance cases and safety cases.

ASCET tools
Information http://www.etas.com/en/products/applications_iso26262.php http://www.etas.com/en/products/ascet_md_modeling_design.php http://www.etas.com/en/products/applications_iso26262.php http://www.etas.com/en/products/ascet_se_software_engineering.php http://www.etas.com/en/products/ascet_scm.php
Overall Description Set of tools for development of electrical and electronic systems in the automotive domain, targeted at modelling and design, prototyping, measurement and calibration, code generation, and testing of microcontrollers.

Atego Artisan Studio
Information http://www.atego.com/products/artisan-studio/
Overall Description Tool for system/software design and modelling.

Atego GSN Modeler
Information http://www.atego.com/products/atego-gsn-modeler/
Overall Description Visual modelling tool for designing and documenting Safety Arguments using GSN.

Atego Process Director
Information http://www.atego.com/products/do-178-with-atego-process-director/
Overall Description Tool for supporting the DO-178 compliance process

Atego Workbench
Information http://www.atego.com/products/atego-workbench/
Overall Description This tool provides a common repository, viewers and traceability for project artefacts developed in different tools (DOORS, Artisan Studio, Excel, Word). It is an integrated engineering environment for team collaboration, information sharing and traceability between tools artefacts.

Cantata++
Information http://www.nohau.se/$2/file/ipl-iso26262.pdf http://www.ipl.com/include/download/Download.php?FileID=p0855 http://www.ipl.com/pdf/Cantata++%20DO-178B%20ED-12B%20Tool%20Qualification%20Presentation.pdf
Overall Description Tool for Unit and Integration Testing for C/C++

CertWare
Information https://c3.nasa.gov/dashlink/projects/59/ http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5747648 http://nasa.github.com/CertWare/ https://c3.nasa.gov/dashlink/static/media/project/Flyer.pdf https://c3.nasa.gov/dashlink/static/media/project/13-0413_Barry_CertWare.pdf
Overall Description Open-source tool for safety case development and management

CESAR Platform
Information http://www.cesarproject.eu http://www.erts2012.org/Site/0P2RUC89/2A-1.pdf http://www.springerlink.com/content/gj4h48w73153l213/
Overall Description Integrated tool-chain for the development of safety-relevant embedded systems

CRESCO
Information http://home.simula.no/~rpanesar/cresco/ http://simula.no/publications/Simula.simula.1120/simula_pdf_file
Overall Description Tool based on model-driven technologies for automated generation of (physical) safety evidence databases

D-Case
Information http://www.dependable-os.net/tech/D-CaseEditor/ https://lwn.net/images/conf/rtlws-2011/proc/Matsuno.pdf
Overall Description Tool for creation or assurance cases.

DECOS Test Bench
Information http://home.mit.bme.hu/~gonczy/publications/safecomp2006.pdf http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4618105 http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5350012 http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4384905
Overall Description Framework to support V&V of dependable systems.

DevCOP
Information http://www.cs.virginia.edu/~sherriff/papers/ISSRE06-Sherriff.pdf
Overall Description Tool for storage and management of certificates, with basic reporting functionality regarding V&V coverage on source code and for estimation of the defect density in a certified program.

DOORS/TraceLine
Information http://www.integrate.biz/traceline/ http://www.integrate.biz/downloads/TraceLine_for_DOORS_081031.pdf
Overall Description DOORS/TraceLine is a DOORS add-on extension for managing and visualizing information and its traceability in DOORS

EDONA Platform
Information http://www.edona.fr
Overall Description Platform targeted at the inter-operation of existing commercial tools and advanced academic technologies necessary to develop automotive software-based systems

eSafetyCase Toolkit
Information http://www.altranpraxis.com/downloads/whitepapers/eSC_challenges.pdf http://www.springerlink.com/index/Q2133234HK512157.pdf
Overall Description Tool for creation and management of safety cases.

EvidenceAgreement
Information http://modelme.simula.no/assets/ieeesoft.pdf
Overall Description Tool for assisting suppliers and certifiers in developing an agreement about the evidence necessary to demonstrate compliance to a safety standard

exSILentia
Information http://www.exida.com/index.php/Software/exSILentia/ http://www.exida.com/index.php/Certification/about/processes_systems http://www.xsindu.cn/beta/pdf/2.2010.BETA-SIL2%20Assessment%20Report.pdf http://files.pepperl-fuchs.com/selector_files/navi/productInfo/cert/cert1416.pdf
Overall Description Tool for Design, operation and maintenance of Safety Instrumented Systems

FormalSafe tools
Information http://www.dfki.de/cps/projects/formalsafe/flyer.pdf, http://www.dfki.de/cps/projects/formalsafe/proposalEN.pdf
Overall Description Platform that integrates formal methods-based verification, change management, and documentation generation for the development of safe and reliable technical systems.

GEMDE
Information http://www.artemis-ediana.eu/documents/D63D_SoftEngineeringProcessCM.pdf http://ebookbrowse.com/03-a-generic-executable-framework-for-model-driven-engineering-gemde-pdf-d199598807
Overall Description Support tool for model-driven engineering

GoedelWorks (FLAME)
Information http://www.altreonic.com/sites/default/files/Systems%20Engineering%20with%20GoedelWorks.pdf
Overall Description Web-based, safety engineering process tool which supports multiple standards (IEC 61508, IEC 62061, ISO DIS 26262, ISO 13849, ISO DIS 25119 and ISO 15998)

GSN CaseMaker
Information http://www.cetadvantage.com/GSNCaseMaker.aspx http://213.144.23.75/wEnglish/download/Deliverables/ D3.2_Part1_Guidelines_Dependability_Hazard_Analysis.pdf
Overall Description Tool for creating GSN models.

HighRely tools
Information http://www.atego.com/products/highrely-check/ http://www.atego.com/products/highrely-trace/
Overall Description It is a simple tool for performing reviews of your project artefacts - files in Configuration Management (CM)

Hive Writer
Information http://www.eventcorp.info/files/ISSEC/Full%20ISSEC%202009%20Paper%20Proceedings%20-%20FINAL.pdf#page=123
Overall Description Tool that supports structured technical documentation via a centrally- managed datastore so that any documents created within the tool are constrained to be consistent with this datastore and therefore with each other. It can be used for safety case development.

Hugin Explorer
Information http://www.hugin.com/productsservices/products/commercial/explorer
Overall Description Tool for the construction, maintenance and usage of knowledge bases using BBNs.

IBM Rational
Information http://www-01.ibm.com/software/rational/solutions/compliance/ http://www-01.ibm.com/software/rational/solutions/electronics/devices.html ftp://ftp.software.ibm.com/software/rational/web/datasheets/G507-0963-00_LoRes.pdf http://www.youtube.com/watch?v=FPi3vq0EcTk&feature=related http://www.youtube.com/watch?v=QVOaVi4wuEw&NR=1&feature=endscreen http://www.youtube.com/watch?v=CY_ylIHLk_Y&feature=relmfu
Overall Description IBM offers several tools which together help company to comply with safety standards, i.e. they provide traceability, auditability of the development process and the resulting product.

IEC Certification Kit
Information http://www.mathworks.se/products/iec-61508/
Overall Description Tool for providing qualification artefacts, certificates, and test suites, and generates traceability matrices. For ISO 26262 and IEC 61508

INESS tools
Information http://www.iness.eu/IMG/pdf/INESS_WSG_MuellerBuxhoevedenSchnieder_TTZ2011_angenommen.pdf www.iness.eu/IMG/pdf/INESS_Deliverable_G4.2_WS_Final.pdf
Overall Description Set of tools for safety case development

interCENELEC
Information http://www.sqs.es/es/solutions/intercenelec/index.php
Overall Description Tool to manage that the development complies with CENELEC 50126, 50128 and 50129 (railway). This tool provides the user with support in the evaluation of compliance with these standards and provides the necessary guidelines for the implementation process.

ISCADE
Information http://www.iscade.co.uk/ http://213.144.23.75/wEnglish/download/Deliverables/ D3.2_Part1_Guidelines_Dependability_Hazard_Analysis.pdf
Overall Description Tool for safety case development.

ISIS
Information http://www.origin-consulting.com/gsnclub/members/2008_q2_pres/Integrated%20support%20presentation.pdf http://213.144.23.75/wEnglish/download/Deliverables/ D3.2_Part1_Guidelines_Dependability_Hazard_Analysis.pdf
Overall Description Tool for creation of GSN models.

jUCMNav
Information http://istar.rwth-aachen.de/tiki-index.php?page=jUCMNav http://jucmnav.softwareengineering.ca/ucm/pub/UCM/VirLibCaise07/CAISE07.pdf http://jucmnav.softwareengineering.ca/ucm/pub/UCM/VirLibRE09Compliance/RE09-Compliance.pdf
Overall Description Tool for creation of goal models

LDRA Tools
Information http://www.ldra.com/iec61508.asp http://www.ldra.com/do178btqp.asp http://www.ldra.com/iso26262.asp
Overall Description LDRA tool suite contains several tools facilitating different aspects of delivering software quality, testing, analysis, and requirements traceability.

LSRD
Information http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5636545
Overall Description Legacy Systems Risk Database

Markup-Based Tool
Information http://www.sciencedirect.com/science/article/pii/S0029549311008752
Overall Description Tool for generating reviews.

McCabe tools
Information http://www.mccabe.com/pdf/DO-178BandMcCabeIQ.pdf http://www.mccabe.com/iq.htm http://www.mccabe.com/cm.htm
Overall Description Tool to analyse the security, quality, and testing of mission, life, and business critical software.

Medini Analyze
Information http://www.ikv.de/index.php/en/products/functional-safety
Overall Description Functional safety analysis and engineering tool. It supports the ISO26262 standard.

Modus
Information http://modelme.simula.no/assets/modus.pdf http://simula.no/publications/Simula.simula.920/simula_pdf_file
Overall Description Tool for quantitative assessment of technical systems. The main driver for Modus has been a particular kind of assessment, called Technology Qualification, aimed at verifying that the overall goals of a new technology will be satisfied within specific margins.

Papyrus/Eclipse Plug-In for GSN Modelling
Information http://subs.emis.de/LNI/Proceedings/Proceedings199/163.pdf
Overall Description Tool for creating GSN models with the Eclipse environment

Parasoft Concerto
Information http://www.parasoft.com/jsp/products/concerto/home.jsp
Overall Description Requirements management tool

PROCE
Information http://modelme.simula.no/assets/er.pdf
Overall Description UML profile for the IEC 61508 standard implemented in IBM's Rational Software Architect tool

Programatica
Information http://programatica.cs.pdx.edu/
Overall Description Tool for Haskell programming language that provides a command line interface (which parses the Haskell project, builds module dependency graphs and calculates code metrics), a html renderer for Haskell program source code (which Functionality syntax highlighting and hyperlinks), and certification evidence management for the parsed code.

RAM Commander
Information http://www.aldservice.com/en/reliability-products/rams-software.html http://www.aldservice.com/en/reliability-products/safety.html
Overall Description Tool for Reliability and Maintainability Analysis and Prediction, Spares Optimization, FMEA/FMECA, Testability, Fault Tree Analysis, Event Tree Analysis and Safety Assessment. Safety Assessment Software is a comprehensive safety tool implementing the requirements and tasks of SAE ARP4761, MIL-STD-882 and other standards

Regulatory Compliance
Information http://www.3ds.com/es/solutions/cross-industry-solutions/overview/management/regulatory-compliance/ http://www.3ds.com/es/solutions/automotive/solutions/enterprise-governance/regulatory-compliance-management/ http://www.3ds.com/products/enovia/solutions/governance-user/regulatory-compliance/
Overall Description Tool for Managing Compliance with Regulations

Reliability Workbench
Information http://www.isograph-software.com/2011/software/reliability-workbench/ http://www.isograph-software.com/2011/software/reliability-workbench/iec-61508-safety-instrumented-systems/ http://www.isograph-software.com/2011/software/reliability-workbench/iso-26262/
Overall Description Suite of reliability, safety and maintainability software

reStructuredText
Information http://docutils.sourceforge.net/rst.html
Overall Description Tool for generation of structured text documents.

RiskCATS
Information http://www.nohau.se/products/safety-security/cats-iec61508-iso26262 http://www.phaedsys.com/principals/riskcats/riskdata/Using_RiskCATS_61508.pdf
Overall Description Tool for Safety Critical work. The RiskCAT requirements tools permit rapid and intuitive navigation around the standard(s).

SafeSlice
Information http://modelme.simula.no/assets/ist-shiva.pdf
Overall Description SysML-based environment for establishing traceability links from safety requirements to design models and for extracting fragments of design that are relevant to safety requirements. SafeSlice has been implemented as a plugin for Enterprise Architect.

Safety Analysis Profile for UML
Information ftp://ftp.software.ibm.com/software/in/rational/innovate/industry_tracks/Douglass-Safety_Analysis_Profile.pdf
Overall Description The Safety Metamodel identifies and characterizes the important concepts (and their metadata) and their relations.

Safety Argument Manager
Information http://www.sciencedirect.com/science/article/pii/S1383762197000325 http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=494529 http://www-users.cs.york.ac.uk/tpk/maintress.pdf http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=145352
Overall Description Tool for management of safety arguments

Safety Case DB
Information http://shop.exida-dev.com/tools/safety-case-db http://www.exida.com/brochures/SafetyCaseDB.pdf
Overall Description Tool for safety case creation in the scope of IEC61508 and ISO26262

SCADE LifeCycle
Information http://www.esterel-technologies.com/products/scade-lifecycle/
Overall Description Application lifecycle manager

Siemens PLM
Information http://www.plm.automation.siemens.com/en_gb/about_us/index.shtml http://www.plm.automation.siemens.com/en_gb/plm/definition/
Overall Description Product lifecycle management software that allows managing the entire lifecycle of a product: to design, produce, support and retire products. It is a big family of products. Except from TeamCenter product, the rest of the family is very specific industry oriented e.g. fiber or aerospace industry.

SpecTRM
Information http://www.safeware-eng.com/software%20safety%20products/software%20safety%20products.htm
Overall Description SpecTRM focuses on system requirements and specification. It is designed to assist in the development of software-intensive safety-critical systems.

System Assessment Management Tools
Information http://www.aldservice.com/en/reliability-products/safety.html
Overall Description Tool for performing the task prescribed in SAE ARP4761, MIL-STD-882 and other standards

The Qualifying Machine
Information http://www.open-do.org/projects/qualifying-machine/ https://forge.open-do.org/plugins/moinmoin/qmachine/1_Presentation https://forge.open-do.org/plugins/moinmoin/qmachine/2_Design?action=AttachFile&do=get&target=QM_GUI_mockups.pdf
Overall Description Intelligent repository specialized in managing the lifecycle and evolution of certification artefacts

ToolNet
Information http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.97.5659&rep=rep1&type=pdf
Overall Description Tool for tracing artefacts that can be used to provide evidence from different tools

TraceVis
Information http://alexandria.tue.nl/extra1/afstversl/wsk-i/ravensteijn2011.pdf
Overall Description Tool for traceability management and visualisation

Trust Case Toolbox
Information http://iag.pg.gda.pl/iag/download/2008_cyra_gorski_expert_assessment_of_arguments_a_method_and_its_experimental_evaluation.pdf http://kio.eti.pg.gda.pl/trust_case/download/TCTEditor_Users_Manual.pdf
Overall Description Tool for development of trust cases

TTE Systems Tools
Information http://www.tte-systems.com/downloads/tte_meeting_iso26262_reqs_2012.pdf
Overall Description Set of tools for development of high-integrity embedded systems, targeted at design, code compilation, static analysis, testing, and traceability.

UOY's GSN Visio plugin
Information http://www.cs.york.ac.uk/~tpk/gsn/gsnaddoninstaller.zip http://213.144.23.75/wEnglish/download/Deliverables/ D3.2_Part1_Guidelines_Dependability_Hazard_Analysis.pdf
Overall Description Vision plugin for creating GSN models.

Vds
Information http://www.origin-consulting.com/gsnclub/members/2008_q2_pres/Integrated%20support%20presentation.pdf
Overall Description Centralised Collaborative Integrated Systems and Software Engineering Environment

VectorCast tools
Information http://www.vectorcast.com/industries/do178b-software-testing.php
Overall Description Set of tools for Dynamic, Static analysis of the source code and Test Coverage monitoring.

Verocel tools
Information http://www.verocel.com/home
Overall Description Set of tools for ensuring safety of mission critical software, targeted at requirements traceability, coverage analysis, and control-coupling analysis

Windriver tools
Information http://www.windriver.com/products/workbench/ http://www.windriver.com/announces/spain/industrial-seminar/presentations/4-Wind-River-Safety-and-Security.pdf http://www.windriver.com/products/product-overviews/PO_VE_61508_0109.pdf
Overall Description Set of tools for debugging, code analysis, advanced visualization, root-cause analysis, and automated tests, monitoring, and debugging at each phase of development, providing early detection of potential defects.

Appendix D. Survey on the State of the Practice concerning Safety Evidence Management

This appendix presents a survey conducted for D6.2 and aimed at gaining insights into how practitioners manage evidence for demonstrating compliance of critical computer-based systems with safety standards. The research questions addressed have been the following ones:

RQ1. What information is provided as evidence for demonstrating compliance?
RQ2. How is evidence evolution managed?
RQ3. What techniques are used for structuring evidence and showing how it contributes to comply with a safety standard?
RQ4. How is evidence adequacy assessed?
RQ5. What challenges do practitioners face regarding provision of evidence?

52 respondents completed the survey. The next subsections show the questionnaire designed for the survey and its results. The questionnaire was based on the findings of a baseline survey on evidence management and a systematic literature review conducted for D6.1, and SurveyMonkey (www.surveymonkey.com) was used to manage it. As mentioned in Section 2, the results have helped to discover practitioners' needs that have had to be considered for requirements specification for evidence management of the OPENCOSS platform.

D.1 Questionnaire

EVIDENCE MANAGEMENT FOR COMPLIANCE OF CRITICAL COMPUTER-BASED SYSTEMS WITH SAFETY STANDARDS

Introduction

Most critical computer-based systems in domains such as avionics, railways, and automotive are subject to some form of safety assessment as a way to ensure that these systems do not pose undue risks to people, property, or the environment. The most common type of assessment is compliance with a safety standard. Examples of safety standards include IEC61508 for various types of systems, DO-178C for avionics, the CENELEC standards for railways, and ISO26262 for the automotive sector.

Demonstration of compliance with a specific standard involves gathering and providing convincing evidence of system safety. BY EVIDENCE, WE REFER TO THE INFORMATION THAT CONTRIBUTES TO DEVELOPING CONFIDENCE IN THE SAFE OPERATION OF A SYSTEM AND THAT IS USED TO MEET THE REQUIREMENTS/OBJECTIVES OF A SAFETY STANDARD. Examples of types of evidence are hazard analysis results, testing results, and reviews.

The aim of this survey is to gain insights into how practitioners manage evidence for demonstrating compliance of critical computer-based systems with safety standards. The survey has been designed as part of the work in OPENCOSS (http://www.opencoss-project.eu/), a European research project on safety assurance and certification of critical systems. Among the aspects to research in OPENCOSS, the survey focuses on the information that is provided as evidence, how evidence change is managed, how evidence is structured, how its adequacy is assessed, and the challenges that can be faced to provide evidence.

The survey is targeted at PRACTITIONERS THAT DIRECTLY PARTICIPATE OR HAVE PARTICIPATED IN EVIDENCE MANAGEMENT FOR DEMONSTRATING COMPLIANCE OF CRITICAL COMPUTER-BASED SYSTEMS WITH SAFETY STANDARDS. The practitioners can correspond to people who have to provide evidence (e.g., an employee of a company that supplies components, such as a safety engineer or a tester), check others'

evidence (e.g., an independent safety assessor), or request evidence (e.g., a person that represents a certification authority).

A questionnaire has been designed for completing the survey. Filling it is expected to take around 15 minutes. All the responses will be held confidential and anonymous.

Finally, if you are interested in the results of the survey, please contact Sunil Nair (sunil@simula.no) or Jose Luis de la Vara (jdelavara@simula.no).

Thank you very much for your participation in the survey.

Background Information

IMPORTANT: Background information must be completed in relation to your participation in the demonstration of compliance of critical computer-based system with safety standards.

1. What is the main application domain in which you are working regarding demonstration of compliance with safety standards? (IMPORTANT: ALL remaining questions must be answered in relation to the domain selected)

- Aerospace
- Automotive
- Avionics
- Defence
- Machinery
- Maritime
- Medical
- Nuclear
- Off-highway equipment
- Oil and gas
- Railways
- Robotics
- Telecommunications
- Trucks
- Other - please specify:

2. What are the safety standards for which you currently provide, check, or request evidence of compliance?

3. What country do you mainly work in regarding demonstration of compliance with safety standards?

- Australia
- Austria
- Belgium
- Brazil
- Canada
- China
- Finland
- France
- Germany
- India
- Italy
- Japan

- Netherlands
- Norway
- Poland
- Portugal
- Russia
- Spain
- Sweden
- UK
- USA
- Other - please specify:

4. What is the main role of the organization for which you work in the development of critical computer-based systems?

- Certification authority
- Component/system supplier
- Developer/manufacturer of final systems
- Independent safety assessor
- Regulation authority
- Development tool vendor
- Other - please specify:

5. How long have you been involved in activities related to demonstration of compliance with safety standards?

- Less than 1 year
- Between 1 and 2 years
- Between 2 and 5 years
- Between 5 and 10 years
- More than 10 years

6. How many projects targeted at demonstrating compliance with safety standards have you participated in?

- Less than 5 projects
- Between 5 and 10 projects
- More than 10 projects

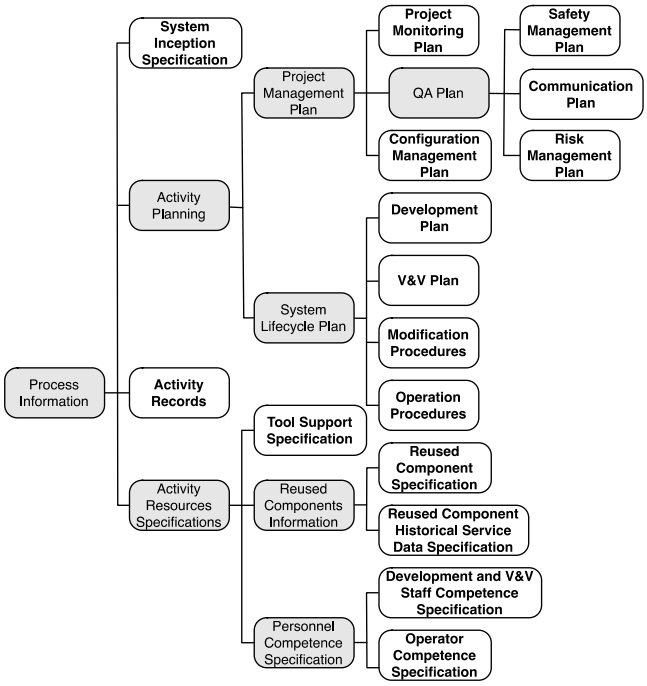
Information Provided as Evidence

REMINDER: please answer the questions in relation to the application domain selected previously.

Safety evidence can be divided into process information (i.e., about the process followed to develop a critical system) and product information (i.e., about the characteristics of the system). Below, two figures show and classify different types of information (and artefacts) that might be used as process-based evidence and product-based evidence, respectively, for demonstrating compliance with safety standards.

On this page you will be asked about the information provided, checked, or requested as evidence. More specifically, you will be asked about the leaf nodes of the classifications. Please note that SOME TYPES OF INFORMATION CAN BE REFERRED TO DIFFERENTLY in the application domain that you selected. You are kindly asked to read the definitions provided for each item carefully before deciding whether it applies to your domain or not.

PROCESS-BASED EVIDENCE

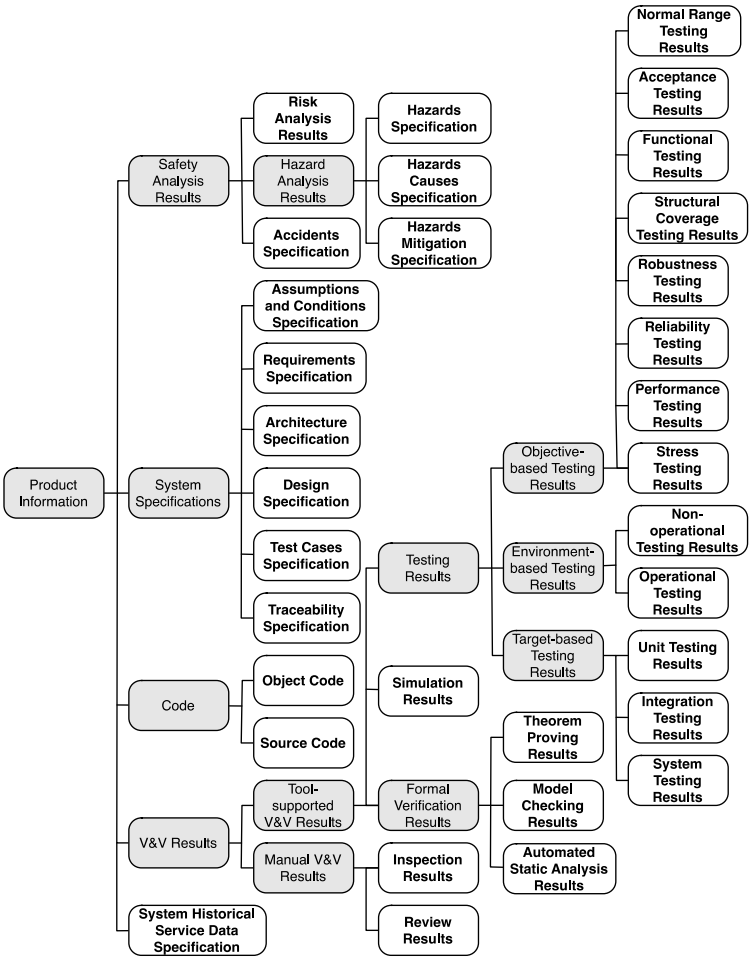


7. What process information (i.e., about the process followed to develop a critical system) do you provide, check, or request as evidence?

- SYSTEM INCEPTION: specification of initial details about the characteristics of a system and how it will be created.
- PROJECT MONITORING PLAN: description of how data about the actual progress of the activity planning of a system will be collected and compared with the baseline plans; e.g., meetings schedule and an organization chart.
- SAFETY MANAGEMENT PLAN: description of the coordinated, comprehensive set of processes designed to direct and control resources to optimally manage the safety of an operational aspect of an organization; e.g., safety culture and safety management processes.
- COMMUNICATION PLAN: description of the activities targeted at creating project-wide awareness and involvement in the development of a system; e.g., specification of the communication channels between service provider, device manufacturer, and regulation authorities.
- PROJECT RISK MANAGEMENT PLAN: description of the activity regarding the development and documentation of an organized and comprehensive strategy for identifying project risks; it includes establishing methods for mitigating risk and tracking them; e.g., risk reduction methodology.
- CONFIGURATION MANAGEMENT PLAN: description of how identification, change control, status accounting, audit, and interface of a system will be governed; e.g., version management and change control procedures.
- DEVELOPMENT PLAN: description of how a system will be built, which includes information about the requirements, design and implementation during coding and/or integration phases; e.g., development methodology and coding standards.
- VERIFICATION AND VALIDATION PLAN: description of how and by whom the verification and validation activities for a system will be executed; e.g., verification environment specification and tests plan.
- MODIFICATION PROCEDURES PLAN: description of the instructions about what to do when performing a modification in a system in order to make corrections, enhancements or adaptations to the validated system, ensuring that the required safety is sustained; e.g., change propagation and maintenance plan.

- OPERATION PROCEDURES PLAN: description of the instructions and manuals necessary to ensure that safety is maintained during system use; e.g., user manual and installation procedure.
- ACTIVITY RECORDS: artefacts collected during the execution of an activity planned for developing a system; e.g., maintenance log and review checklists.
- TOOL SUPPORT SPECIFICATION: specification of the different tools that will be used in the system lifecycle plan; e.g., tool qualification report.
- REUSED COMPONENTS SPECIFICATION: specification of the characteristics of an existing system that is (re)used to make up a system; e.g., reused component reliability specification and qualification documentation of a real-time operating system.
- REUSED COMPONENTS HISTORICAL SERVICE DATA SPECIFICATION: specification of the dependability of a component reused in a system based on past observation of the behaviour; e.g., mean time between failures.
- DEVELOPMENT AND V&V STAFF COMPETENCE: specification of the skills or knowledge that the parties involved in the development and V&V plans of a system need in order to perform the activities assigned to them; e.g., staff experience and tool training.
- OPERATOR COMPETENCE: specification of the skills or knowledge that the parties involved in the operation procedures need in order to perform the activities assigned to them; e.g., operational staff training needs specification.
- I do not provide, check, or request process information as evidence
- Other(s) – please specify:

PRODUCT-BASED EVIDENCE



8. What product information (i.e., about the characteristics of the system) do you provide, check, or request as evidence?

- RISK ANALYSIS/ASSESSMENT RESULTS: specification of the expected amount of danger when an identified hazard will be activated and thus become an accident in a system.
- HAZARDS SPECIFICATION: specification of the conditions in a system that can become a unique, potential accident.
- HAZARDS CAUSES SPECIFICATION: specification of the factors that create the hazards of a system.
- HAZARDS MITIGATION SPECIFICATION: specification of how to reduce hazard likelihood and hazard consequences when a hazard cannot be eliminated in a system.
- ACCIDENTS SPECIFICATION: specification of the conditions in a system that can become a unique, potential accident.
- ASSUMPTION AND CONDITIONS SPECIFICATION: description of the constraints on the working environment of a system for which it was designed.
- REQUIREMENTS SPECIFICATION: specification of the external conditions and capabilities that a system must meet and possess, respectively, in order to allow a user to solve a problem or achieve an objective, or to satisfy a contract, standard, or other formally imposed documents.
- ARCHITECTURE SPECIFICATION: description of the fundamental organization of a system, embodied in its components, their relationships to each other and to the environment, and the principles guiding its design and evolution.
- DESIGN SPECIFICATION: specification of the components, interfaces, and other internal characteristics of a system or component.
- TEST CASE SPECIFICATION: specification of the tests inputs, execution conditions, and predicted results for a system to be tested.
- TEST RESULTS: results from the execution of test cases; they also indicate if the objectives and criteria of the tests have been met.
- TRACEABILITY SPECIFICATION: specification of the relationship between two or more pieces of information related to the development - process or product information - of a system.
- OBJECT CODE: computer instructions and data definitions in a form output by an assembler or compiler.
- SOURCE CODE: computer instructions and data definitions expressed in a form suitable for input to an assembler, compiler, or other translator.
- THEOREM PROVING RESULTS: results from the verification of a system by formally expressing its properties in a common language based on mathematical logic and using a theorem prover; a property can be shown to be a logical consequence of a set of axioms if it can be formally derived from the axioms with a set of deduction steps, which are instances of the set of inference rules that are allowed in the common language.
- MODEL CHECKING RESULTS: results from the verification of the conformance of a system to a given specification by providing a formal guarantee; the system under verification is modelled as a state transition system, and the specifications are expressed as temporal logic formulae that express constraints over the system dynamics.
- AUTOMATED STATIC ANALYSIS RESULTS: results from an automatic process for evaluating a critical system or component based on its form, structure, content, or documentation; e.g., static code analysis and cyclomatic complexity analysis.
- INSPECTIONS/AUDITS: results from the visual examination of system lifecycle products of a system in order to detect errors, violations of development standards, and other problems; e.g., code inspection.
- REVIEWS/WALKTHROUGHS: description of a process or meeting during which a work product or set of works products is presented to some interested party for comment or approval; e.g., design review.

- SIMULATION RESULTS: Results from the verification of a critical system by creating a model that behaves or operates like the system when provided with a set of controlled inputs; e.g., emulation and results from Matlab/Simulink.
- SYSTEM HISTORICAL SERVICE DATA SPECIFICATION: specification of the dependability of a system based on past observation of its behaviour; e.g., prior field reliability in similar applications.
- I do not provide, check, or request process information as evidence.
- Other(s) - please specify:

9. What types of testing results are included in the product information (i.e., about the characteristics of the system) that you provide, check, or request as evidence?

- NORMAL RANGE TESTING: results from the verification of the behaviour of a system under normal operational conditions; e.g., equivalence classes and input partitioning testing.
- ACCEPTANCE TESTING: results from the validation of the behaviour of a system against the customers' requirements.
- FUNCTIONAL TESTING: results from the validation of whether or not the observed behaviour of a system conforms to its specification; e.g., hazard directed testing.
- STRUCTURAL COVERAGE TESTING: results from the verification of the behaviour of a system by executing all or a percentage of the statements or blocks of statements in a program, or specified combinations of them, according to some criteria; e.g., MC/DC and branch coverage testing.
- ROBUSTNESS TESTING: results from the verification of the behaviour of a system in the presence of faulty situations in its environment; e.g., fault injection testing.
- RELIABILITY TESTING: results from the verification of fault-free behaviour in a system; e.g., statistical and probabilistic testing.
- PERFORMANCE TESTING: results from the verification of the performance requirements of a system such as capacity and response time; e.g., timing and memory partitioning analysis.
- STRESS TESTING: results from the verification of the behaviour of a system at the maximum design load, as well as beyond it; e.g., boundary value and exhaustive input testing.
- NON-OPERATIONAL TESTING: results from evaluation of a system in an environment that does not correspond to but replicates its actual operational environment.
- OPERATIONAL TESTING: results from the evaluation of a system in its actual operating environment.
- UNIT/MODULE TESTING: results from the evaluation of the functioning in isolation of software pieces, which are separately testable; depending on the context, these could be the individual subprograms or a larger component made of tightly related units.
- INTEGRATION TESTING: results from the evaluation of the interaction between system components.
- SYSTEM TESTING: results from the evaluation of the behaviour of a whole system; external interfaces to other applications, utilities, devices, or the operating environment are also evaluated at this level.
- I do not provide, check, or request testing information as evidence
- Other(s) - please specify:

Evidence Change Management

REMINDER: please answer the questions in relation to the application domain selected before.

A characteristic of evidence for demonstrating compliance with safety standards is that it can evolve. That is, a set of evidence can change because of, for instance, some modification in a system or the need to provide new evidence in order to guarantee system safety in a new context. This can affect single, isolated pieces of evidence as well as several pieces of evidence that are interrelated. For example, the modification

of a requirement might affect the test cases specified to validate it. Consequently, the change of a piece of evidence can affect other pieces, which might become inadequate and/or might have to be (re)validated.

10. For the evidence that you provide, check, or request for demonstrating compliance with safety standards, how is the degree of completeness of evidence checked?

- Manually (e.g., with a paper-based checklist)
- With tools that store and provide information about the degree of completeness for some types of evidence
- With tools that store and provide information about the degree of completeness for all types of evidence
- I do not know it

11. When a piece of evidence has changed, how is its effect on other pieces of evidence checked?

- Manually, without following a predefined process
- Manually, according to a predefined process
- Automatically, using change analysis tools that provide information for the change effect of some types of evidence
- Automatically, using change analysis tools that provide information for the change effect of all types of evidence
- I do not know it
- Other(s) please specify:

12. Do you provide, check, or request details about how the change of a piece of evidence has affected others?

- Yes
- No

13. In the documentation that you provide, check, or request for demonstrating compliance with safety standards, how is traceability between different pieces of evidence recorded?

- Traceability matrices
- Models
- Metadata
- Hyperlinks
- Naming conventions
- Traceability between pieces of evidence is not recorded
- I do not know it
- Other(s) - please specify:

<u>Structuring of Evidence</u>

REMINDER: please answer the questions in relation to the application domain selected previously.

14. This question lists a set of techniques that can be used for structuring evidence in order to show how it contributes to the fulfilment of the requirements/objectives of a safety standard. Please indicate how often you use, check, or request each technique (Never; Rarely; Sometimes; Very often; Always)

- Unstructured text
- Structured text (providing patterns for the text to write)
- Textual templates (indicating the information to provide/the sections to fill)
- Argumentation-based graphical notations (e.g., GSN)
- Conceptual/information models (e.g., with UML)
- Process models (e.g., with SPEM)

15. If you would like to add any further techniques for structuring of evidence, please do so in the box below, and also indicate how often you use, check, or request them (for example, Technique X: very often; Technique Y: rarely, and so on)

Evidence Adequacy Assessment

REMINDER: please answer the questions in relation to the application domain selected before

When managing evidence for demonstrating compliance with safety standards, it is also common to assess its adequacy. Adequacy is usually assessed based on the confidence in the information collected to support a particular claim about system safety. Adequacy can be estimated, for instance, by means of a qualitative approach (e.g., a level confidence) or a quantitative approach (e.g., a numerical estimation of the adequacy).

16. How often do you use, check, or request the following techniques for determining evidence adequacy? (Never; Rarely; Sometimes; Very often; Always)

- Expert judgement, without documenting the rationale behind the assessment
- Expert judgement, documenting the rationale behind the assessment
- Argumentation
- A quantitative approach (e.g., based on the use of Bayesian Belief Networks)
- A qualitative approach (e.g., based on the assignation of confidence levels to evidence)
- Checklists

17. If you would like to add any further techniques for evidence adequacy assessment, please do so in the box below, and also indicate how often you use, check, or request them (for example, Technique X: very often; Technique Y: rarely, and so on)

18. For the evidence that you provide, check, or request, do you check if the confidence in a piece of evidence is related to the confidence of other pieces?

- Yes
- No

19. When a change occurs in the confidence in a piece of evidence that you provide, check, or request, do you check how the change might affect the confidence in other pieces of evidence?

- Yes
- No

Challenges in Evidence Provision

REMINDER: please answer the questions in relation to the application domain selected previously.

Practitioners might face different challenges when having to provide evidence for demonstrating compliance with safety standards. For example, safety standards can be difficult to understand, thus practitioners might have problems in determining what evidence has to be provided to comply with a safety standard

20. This question lists a set of possible challenges regarding provision of evidence for demonstrating compliance with safety standards. For those challenges that you have faced or observed, please indicate how important you consider them to be (Unimportant; Of little importance; Moderately important; Important; Very important)

- Compliance demonstration for new technologies (for example, model-driven technologies/development)
- Suitability and application of safety standards
- Determination and decision upon the information that can be provided as evidence
- Provision of adequate process information (i.e., about the process followed to develop a critical system) as evidence for the whole development and V&V process
- How to effectively create and structure safety cases
- Compliance demonstration for systems whose compliance has not been previously demonstrated (for example, a legacy system)
- Existence of problems which, based on your experience, are exclusive to the application domain selected and do not arise in others (for example, due to special regulations or processes)
- Determination of confidence in evidence to support a particular claim about system safety
- Need for providing arguments to show how evidence meets the requirements of a safety standard
- Provision of evidence for systems that reuse existing components/subsystems

21. If you would like to add any further challenges, please do so in the box below, and also indicate its importance (for example Challenge X: very important; Challenge Y: moderately important, and so on)

Follow-Up Studies

22. Finally, please fill the following information if you are interested in participating in follow-up studies (OPTIONAL)

- Name
- Organization
- Role
- Email

D.2 Results

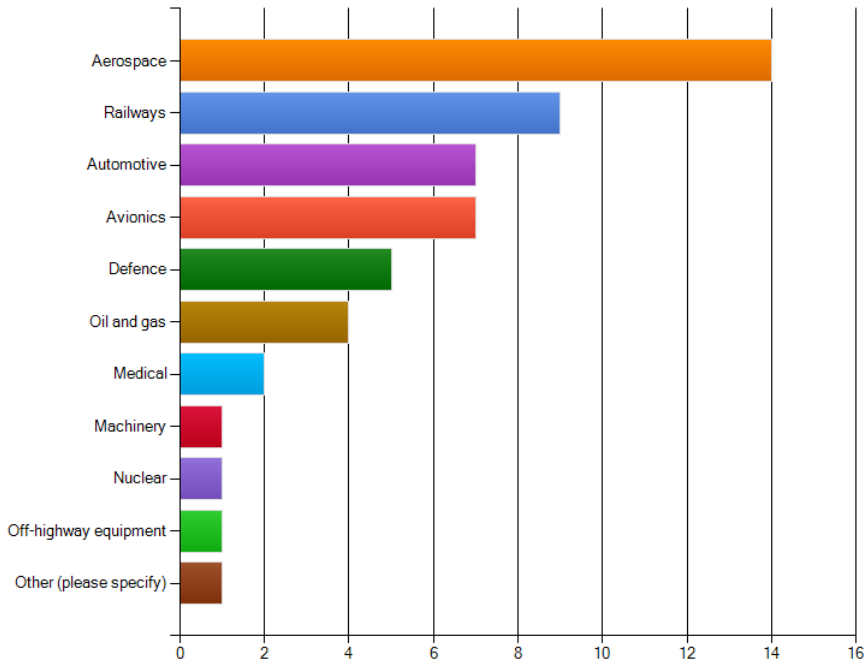


Figure 15. Application domain

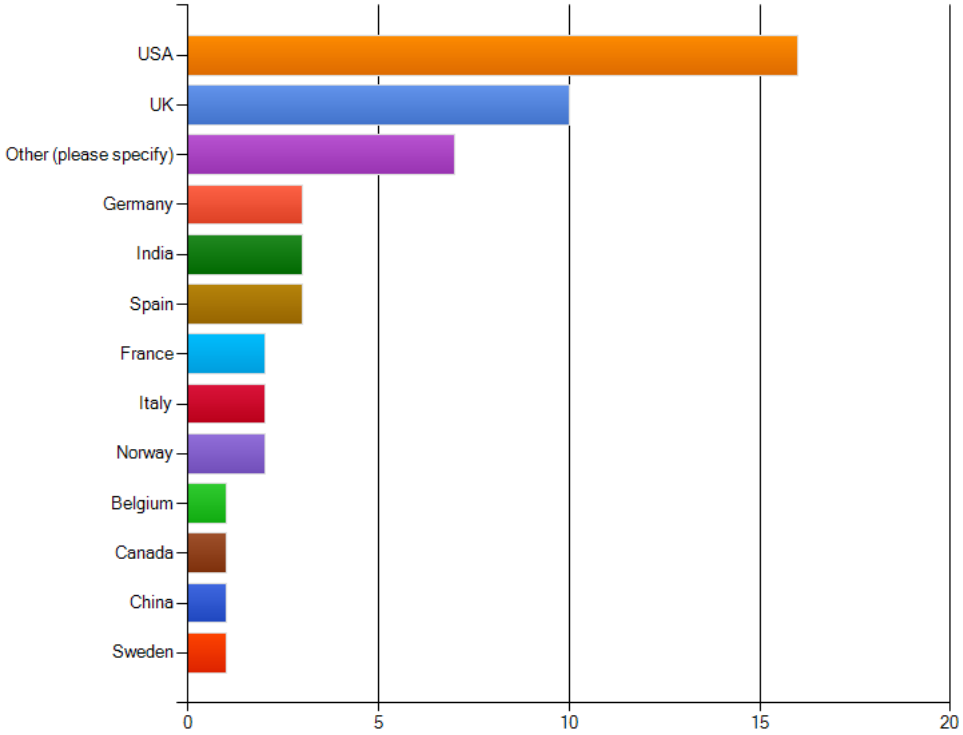


Figure 16. Country

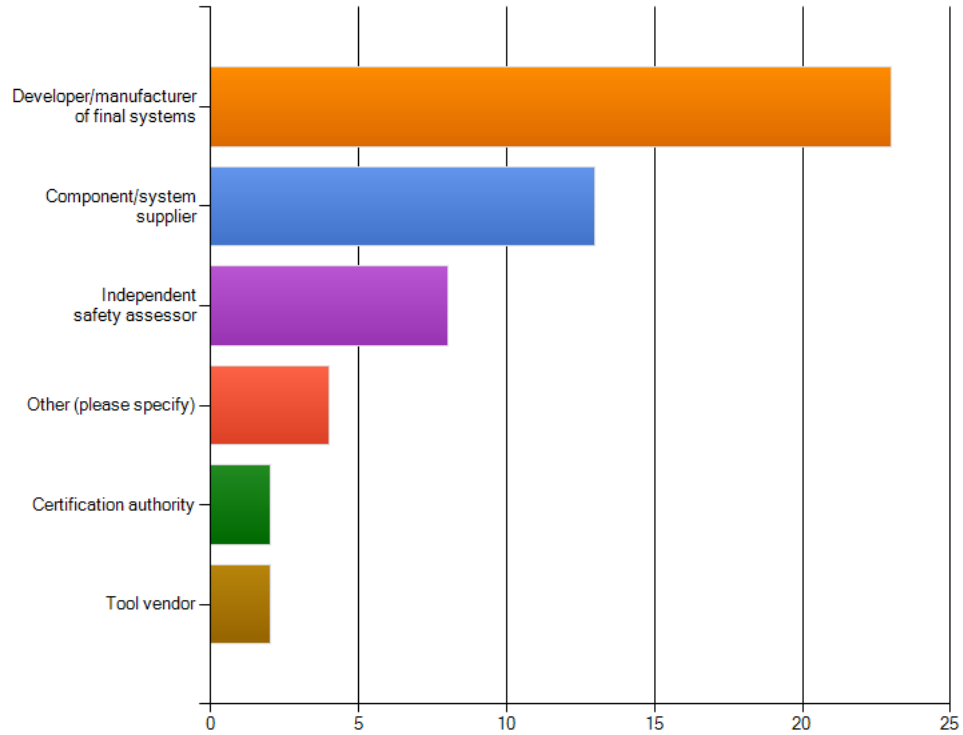


Figure 17. Role

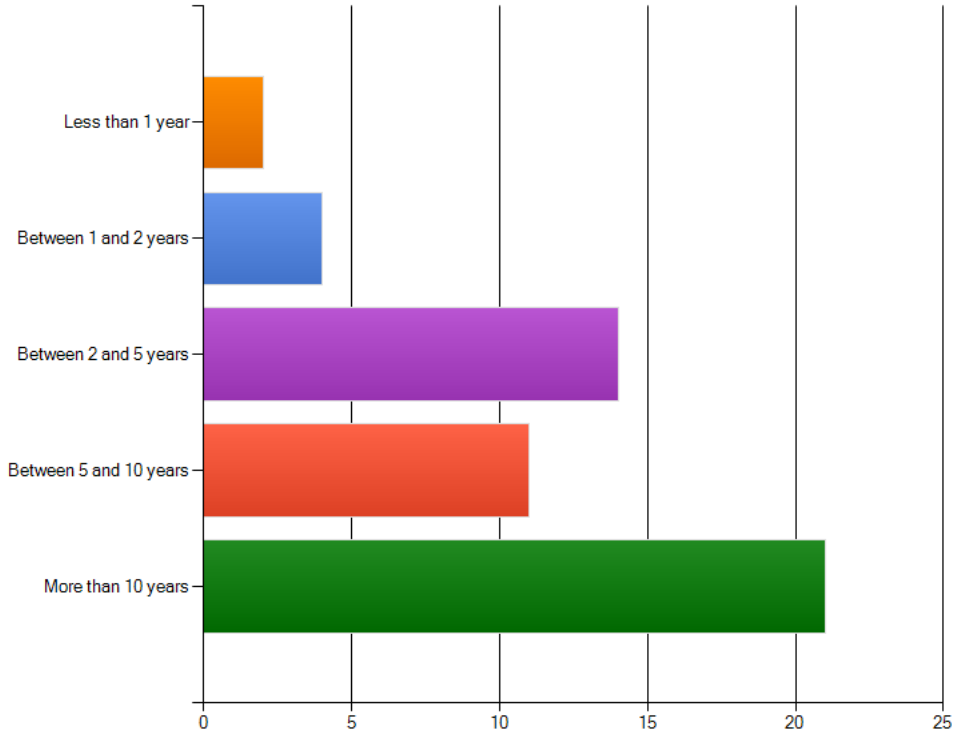


Figure 18. Years of experience

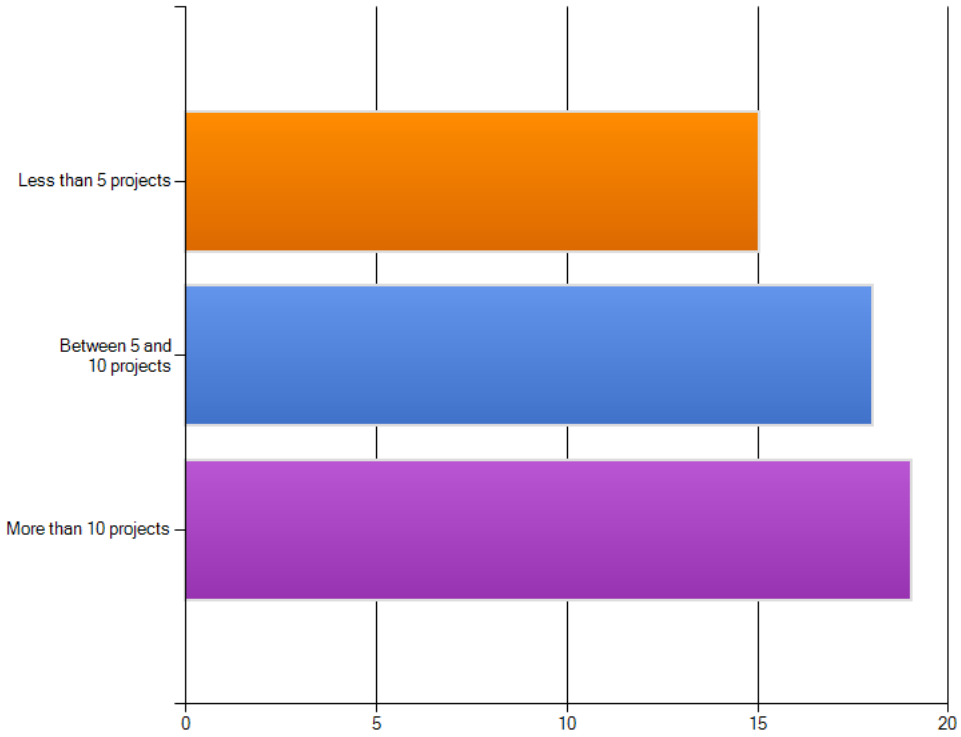


Figure 19. Number of projects of experience

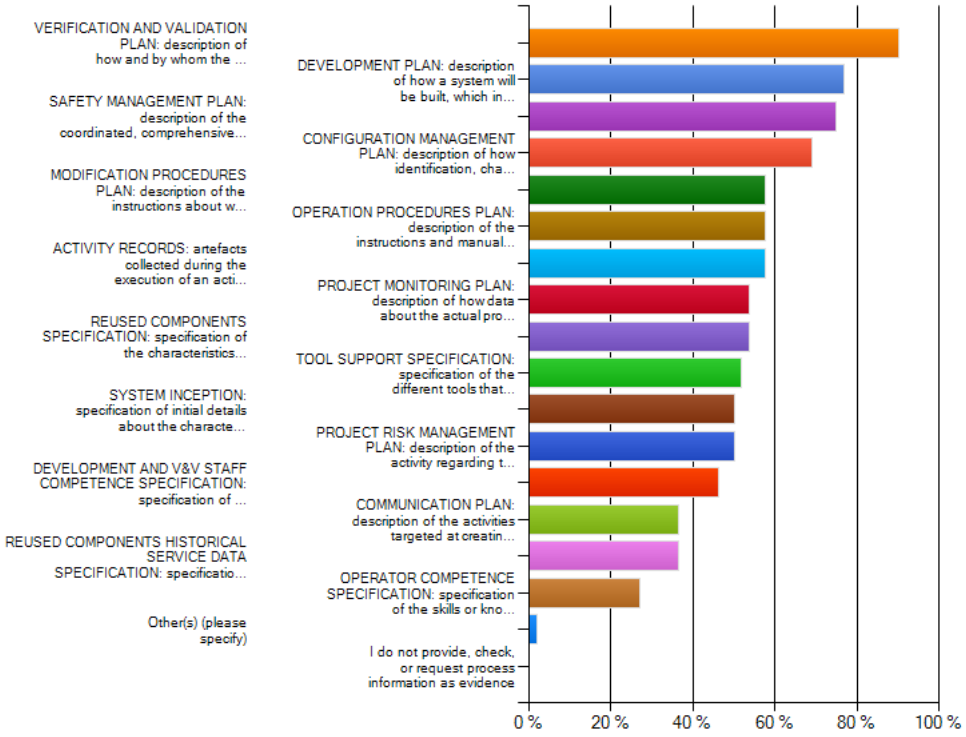


Figure 20. Process-based evidence used

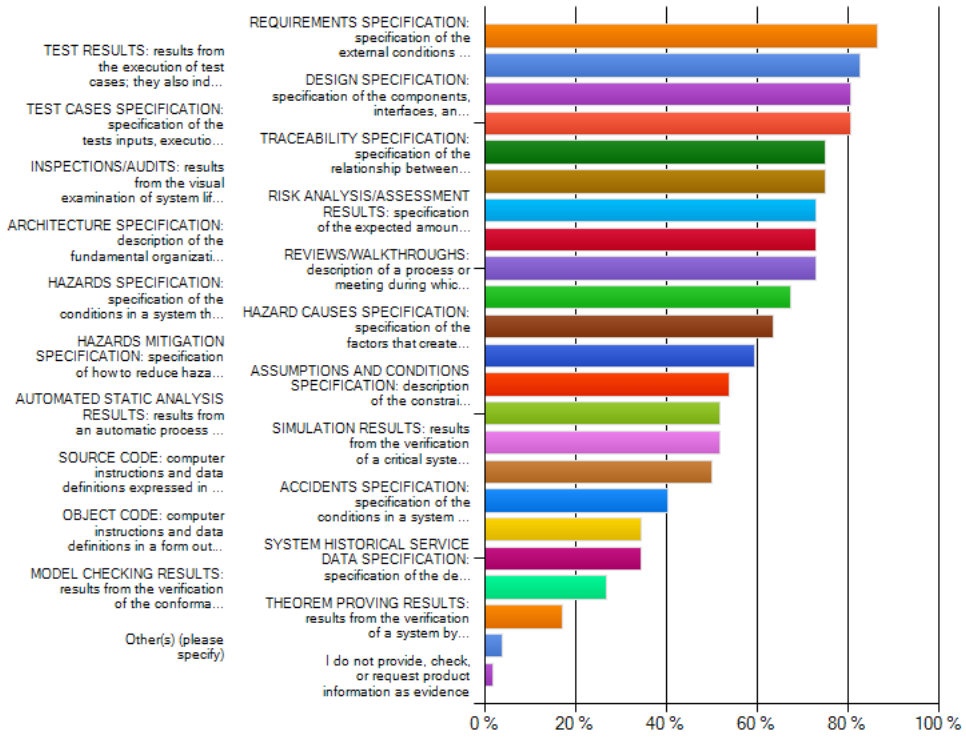


Figure 21. Product-based evidence used

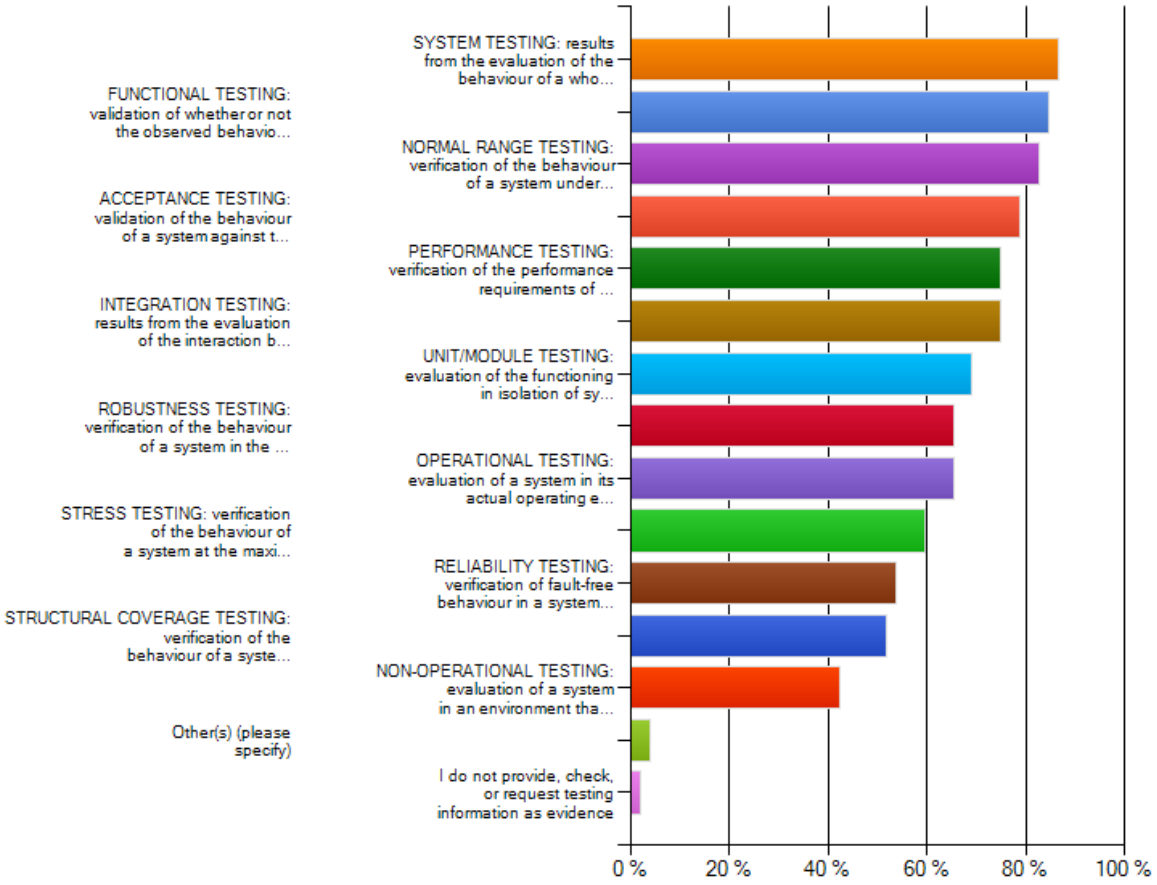


Figure 22. Types of testing used

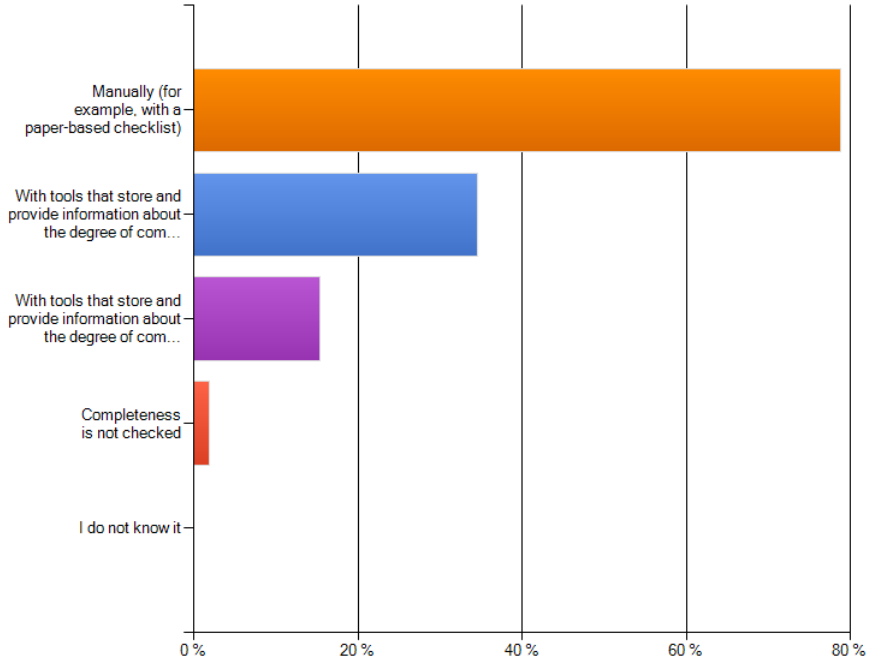


Figure 23. Ways to check evidence completeness

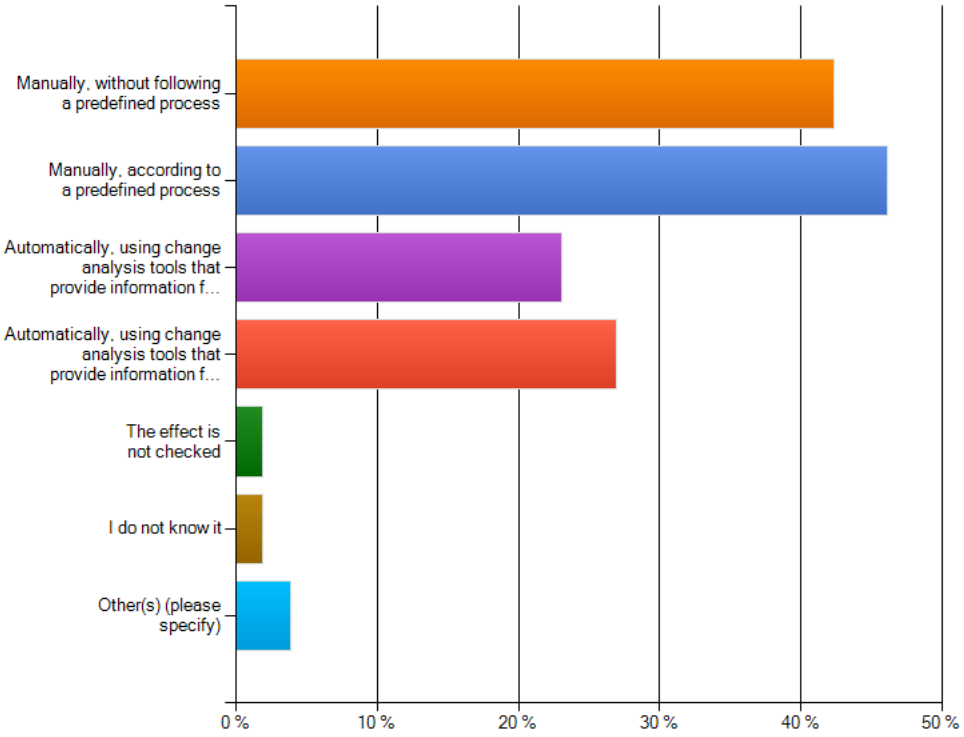


Figure 24. Ways to perform evidence change impact analysis

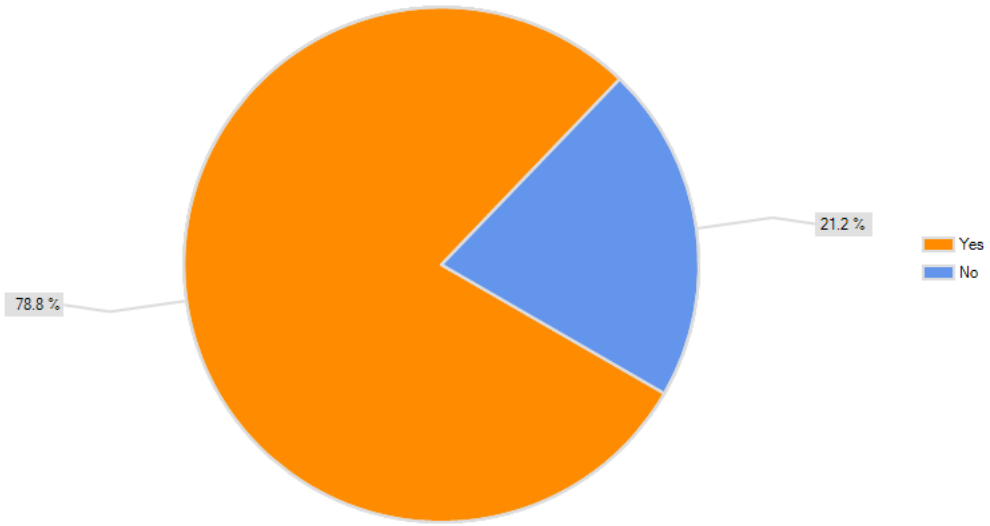


Figure 25. Record of details about evidence change impact

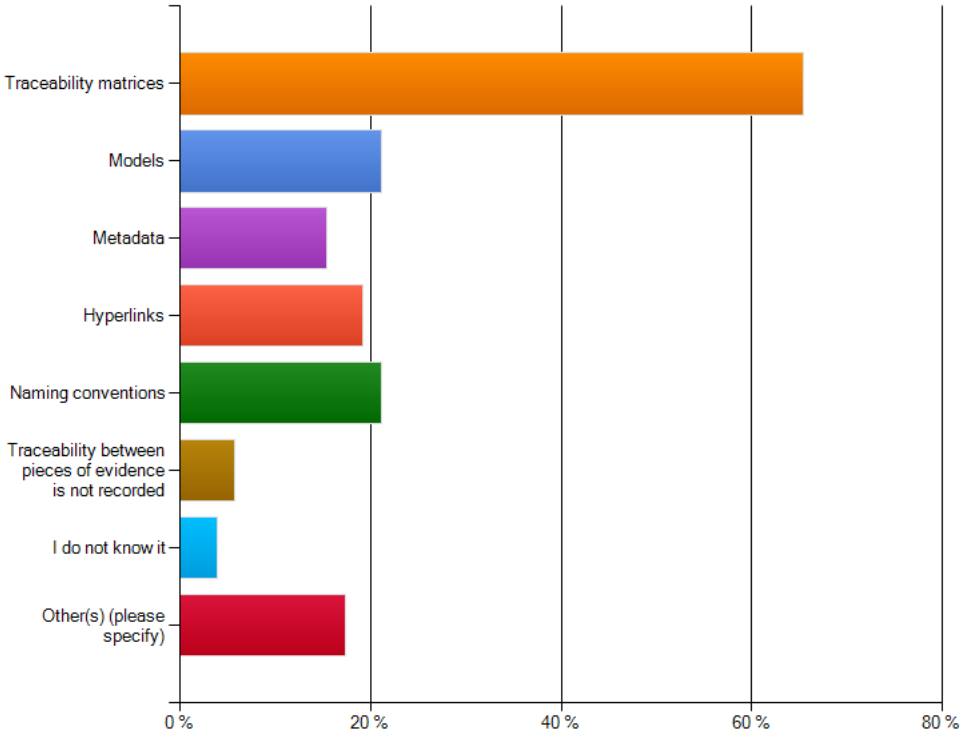


Figure 26. Ways to record evidence traceability

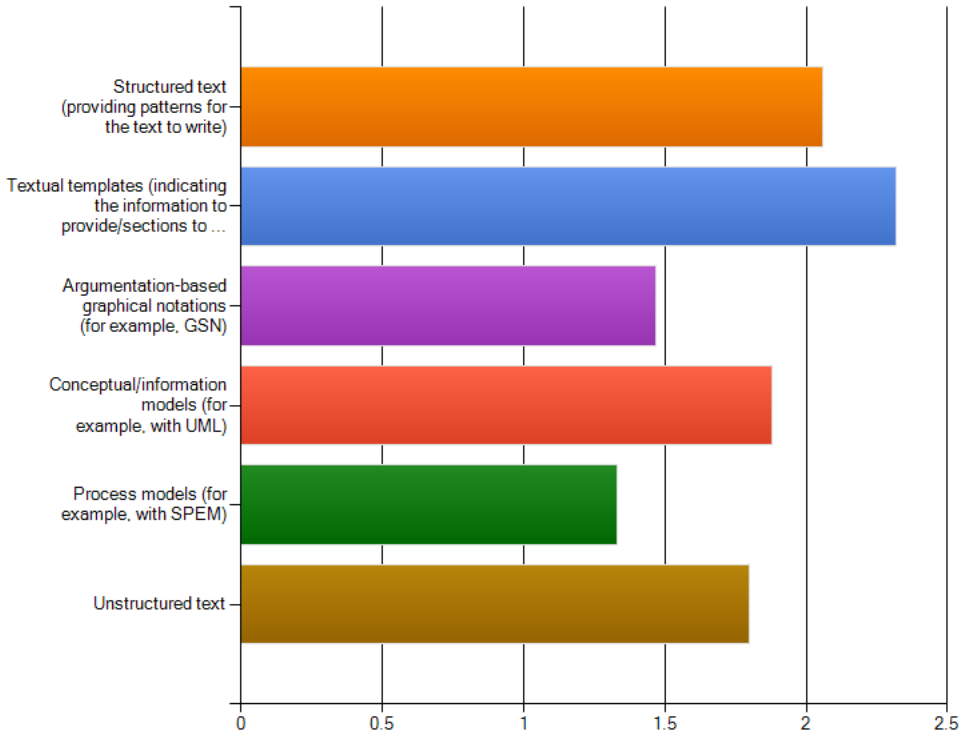


Figure 27. Techniques for structuring of evidence

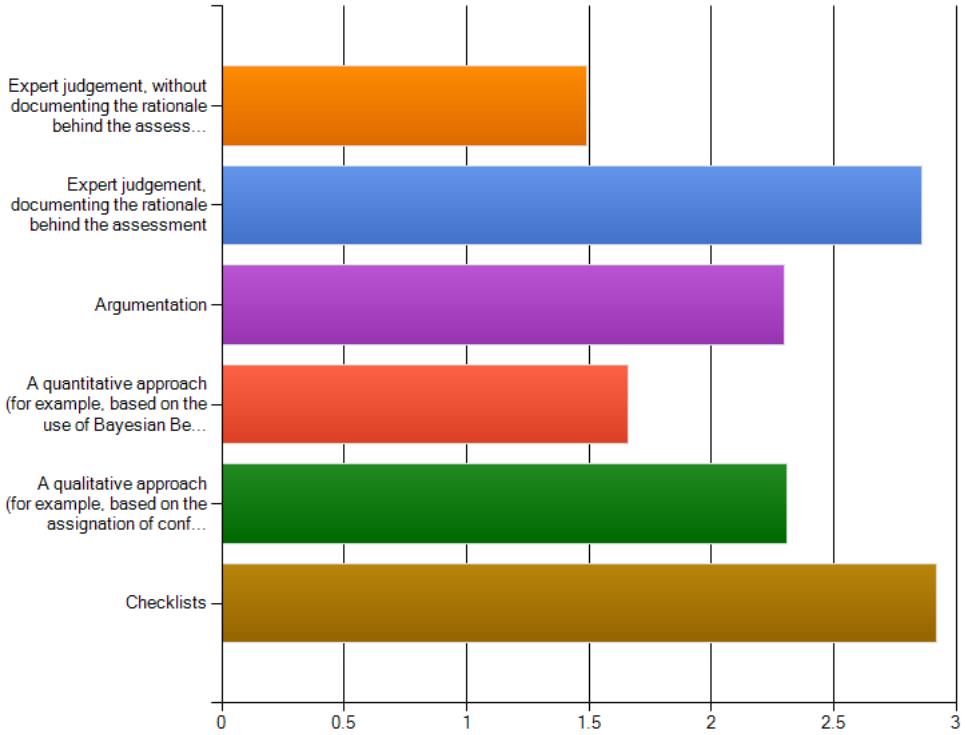


Figure 28. Techniques for evidence adequacy assessment

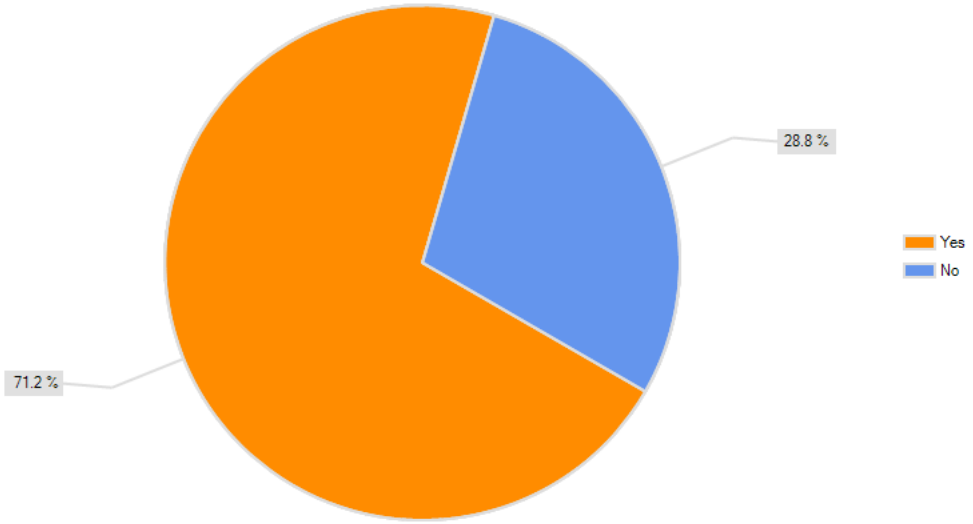


Figure 29. Check of how the confidence in a piece of evidence depends on the confidence in others

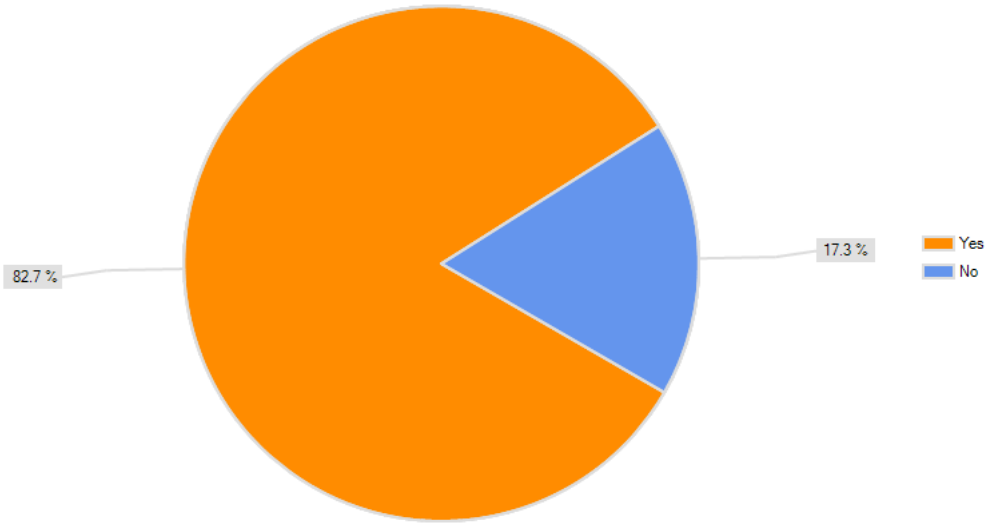


Figure 30. Check of confidence change effect in other pieces of evidence

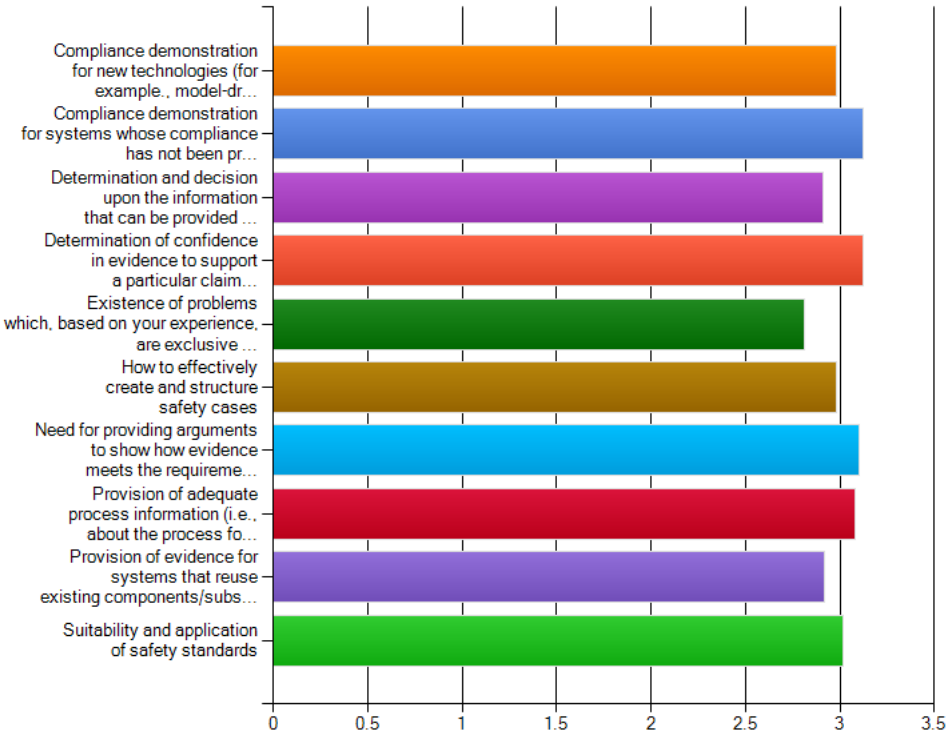


Figure 31. Importance of challenges for provision of evidence