Towards a Model-Based Evolutionary Chain of Evidence for Compliance with Safety Standards

Jose Luis de la Vara[1], Sunil Nair[1], Eric Verhulst[2], Janusz Studzizba[3], Piotr Pepek[3], Jerome Lambourg[4], and Mehrdad Sabetzadeh[1]

[1]Simula Research Laboratory
P.O. Box 134, 1325 Lysaker, Norway
{jdelavara, sunil, mehrdad}@simula.no

[2]Altreonic
Gemeentest. 61A, B3210 Linden, Belgium
eric.verhulst@altreonic.com

[3]Parasoft S.A.
Kielkowskiego 9, Krakow, 30-704, Poland
{januszst, piotr}@parasoft.com

[4]AdaCore
46 rue d'Amsterdam, 75009 Paris, France
lambourg@adacore.com

Abstract. Compliance with safety standards can greatly increase the development cost and time of critical systems. Major problems arise when evolutions to a system entail reconstruction of the body of safety evidence. When changes occur in the development or certification processes, identification of the new evidence to provide, the evidence that is no longer adequate, or the evidence that can be reused poses some challenges. Therefore, practitioners need support to identify how a chain of evidence evolves as a result of the changes. Otherwise, execution of the above activities can be very costly, and it can even result in abandonment of certification efforts. This paper outlines a solution to deal with these challenges. The solution is based on the use of model-driven engineering technology, which has already been applied for safety certification but not from an evolutionary chain of evidence-based perspective. The paper also sets the background for developing the solution, describes real situations in which the solution can help industry, and discusses possible challenges for developing it. The solution will be developed as part of OPENCOSS, a research project on cross-domain evolutionary certification.

Keywords: safety, safety certification, evidence, chain of evidence, evidence evolution, model-driven engineering, impact analysis, OPENCOSS.

1 Introduction

Most critical systems in domains such as avionics, railways, and automotive are subject to some form of safety assessment as a way to ensure that the systems do not pose undue risks to people, property, or the environment. The most common type of assessment is safety certification [18], whose goal is to provide a formal assurance that a system is deemed safe by a licensing or regulatory body. Certification is typically performed based on one or more standards that apply in a given domain. Examples of standards include IEC61508, DO-178C for avionics, the CENELEC standards for railways, and ISO26262 for the automotive sector [7, 13].

Demonstrating compliance with a safety standard involves the provision of evidence to show that the relevant criteria in the standard are met. This imposes unavoidable, high costs on companies [15]. Furthermore, system evolution often becomes costly because it entails regenerating the entire body of evidence. The evidence should be re-examined whenever the system is modified and, if the evidence is no longer adequate, new evidence should be generated. This is closely related to (change) impact analysis [4], which aims at identifying the potential consequences of a change, or at estimating what needs to be modified to accomplish it.

As a result, when a system is certified, subsequent modifications are usually avoided. This can also hinder innovation, as use of new technologies would require re-certification. Consequently, new approaches centred on evidence evolution, including chains of evidence (Section 2.1), are necessary.

This paper presents a solution sketch for managing evolutionary chains of evidence and thus how to deal with the above challenges for safety certification. The solution will be developed as part of the work in OPENCOSS [26], a large-scale European research project whose goal is to devise a common certification framework for the railway, avionics and automotive domains, addressing evidence evolution.

The solution is based on the use of model-driven engineering (MDE) [37], thus it supports a model-based evolutionary chain of evidence. As we discuss below, MDE is an enabler for performing several tasks related to evidence and chains of evidence management. For example, MDE can facilitate standard interpretation, electronic evidence management, and identification of chains of evidence.

In addition, the paper (1) sets the background on which the solution is based and that makes us believe that it is necessary and feasible, (2) describes realistic situations in which evidence and thus chains of evidence evolve, and (3) outlines the challenges that we might face. The set of challenges are related to both technology issues and business issues (e.g., industrial acceptance).

The rest of the paper is organized as follows. Section 2 presents background work. Section 3 describes situations in which evidence evolves. Section 4 outlines the envisioned solution, whereas Section 5 discusses the challenges that we foresee. Finally, Section 6 summarises our conclusions and future work.

2 Background

This section introduces: (a) safety certification; (b) OPENCOSS; (c) two surveys on certification and evidence management; (d) past work on evidence management and on model-based safety certification, and; (e) some related projects and initiatives. Overall, past work has not focused enough on evolution of chains of evidence.

2.1 Safety Certification

Safety-critical systems are typically subject to a rigorous safety certification process. The purpose of certification is to provide assurance that the system is safe to use in a specific environment under specific conditions [7].

Satisfaction of safety objectives according to a specific standard involves gathering convincing evidence during the lifecycle of the system. In general, evidence can be defined as "the available body of facts or information indicating whether a belief or proposition is true or valid" [28]. However, one can seldom argue that evidence for safety certification serves as a definitive proof of the truth or validity of safety claims, but only whether the evidence is sufficient for building (adequate) confidence in the claims. Hence, we define evidence for safety certification as "information or artefacts that contribute to developing confidence in the safe operation of a system". Such information or artefacts must also be linked to the requirements/objectives of the safety standard(s) that need to be met.

A chain of evidence is a set of pieces of evidence that are related (e.g., the agent that has created a requirements specification, the test cases derived from the requirements, etc.). Therefore, traceability between these pieces of evidence exists. By evolutionary, we mean that a chain of evidence can suffer changes (e.g., a requirement is modified), and thus it can evolve. As a result, the chain of evidence might not be adequate anymore (e.g., the related test cases might have to be updated).

Safety evidence can be supported by argumentation. Safety arguments are a set of inferences between claims and evidence that leads from the evidence forming the basis of the argument to a top-level safety claim. This claim is typically that the system is safe to operate in its intended environment [7].

2.2 OPENCOSS

OPENCOSS [26] is a FP7 European project that aims (1) to devise a common certification framework that spans different vertical markets for railway, avionics and automotive industries, and (2) to establish an open-source safety certification infrastructure. 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 are expected to boost innovation and system upgrades considerably. The project consortium consists of 17 partners from 9 different countries.

The problems that OPENCOSS addresses are: (1) lack of precision and large variety of certification requirements; (2) lack of composable/system view for certification; (3) high and non-measured costs for (re)certification, and; (4) lack of openness to innovation and new approaches. The project will deal with: (1) creation of a common certification language (metamodel); (2) compositional certification; (3) evolutionary chains of evidence (whose envisioned solution is outlined in this paper); (4) transparent certification process, and; (5) compliance-aware development process.

2.3 Earlier Surveys on Certification Issues and Evidence Management

This section summarises part of the results of two surveys that have been conducted at the beginning of OPENCOSS in order to gain an overall understanding of practices related to the project.

In the first survey [1], a total of 85 valid responses were obtained on certification issues. The main conclusions related to this paper are:

- Certification was considered as important for 68% of the respondents.
- The demotivating factors for certification are:
    - Effort, cost, complexity, inconsistency, bureaucratic (paperwork) (60.7%)
    - Change management (evolving standards, evolving products), differences national/ international (21.4%)
    - Rigidity, lagging market and technology (17.9%)

In the second survey [25], a total of 15 responses were obtained from OPENCOSS partners. It aimed to set a baseline concerning the state of the practice on safety certification within the consortium. The main conclusions related to this paper are:

- Traceability between evidence was acknowledged as a major concern for safety certification by most of the partners.
- 11 partners selected MDE as a suitable way to manage traceability, and only matrices were selected more times.
- Most of the partners recommended using models to structure certification documentation.
- 27 types of traceability between types of evidence were identified.

The results of the surveys suggest (1) the need of mitigating the demotivating factors for certification, (2) the importance of chains of evidence (traceability), and (3) the suitability of using MDE technology for evidence management.

2.4 Safety Evidence Management and Evolution

This section reviews some existing research and tools that have dealt with safety evidence, its management, and its evolution.

Some works on the nature of safety evidence (e.g., [17]) have discussed process-based evidence (i.e., about the process followed) and product-based evidence (i.e., about system characteristics), and what type of evidence can be regarded as better suited for demonstrating safety. In general, the conclusion is that both types of evidence are necessary and are related.

Other works have defined evidence items for IEC61508 [35] and for the nuclear domain [16], have provided classifications of artefacts that can be used as evidence (e.g., [12]), or have proposed ways to structure evidence in certification documentation (e.g., [39]). Within OMG, there are two initiatives aimed at standardizing the notion of and the concepts related to assurance evidence [22] and arguments [24]. In relation to this paper, the main weakness of these works is that they have not dealt with chains of evidence. Other works have modelled standards such as IEC61508 [29] and DO-178B [42], identifying their main concepts and relations. However, they have not dealt with evolutionary chains of evidence.

Research-based prototypes have been developed for (1) specification of certificates associated to source code [34], V&V activities [38], and the activities of the development process [41], and (2) expert judgement-based quantification of confidence on evidence [35]. MDE-based prototypes for evidence management are presented in the following subsection.

Some existing commercial tools that directly or indirectly deal with evidence management are:

- Atego Workbench [3], which supports traceability, impact analysis, and versioning of software development work products.
- GoedelWorks [2], which supports IEC61508, IEC62061, ISO26262, ISO13849, ISO-DIS25119 and ISO15998, supports the specification of dependencies between (evidence) entities, and provides an entity lifecycle (Defined, InWork, FrozenForApproval, and Approved).
- Medini Analyze [19], which supports ISO 26262 and allows specification of traceability to express dependencies between (evidence) elements.
- Parasoft Concerto [33], which supports management (i.e., lifecycles) of requirements, test and defects, as well as traceability between them and impact analysis.

In summary, we consider that new research efforts that address and study in detail chains of evidence (of more types) and their evolution are necessary.

2.5 Model-Driven Compliance with Safety Standards

MDE can be a suitable and very useful technology for safety evidence management [32]. It is based on the use of models as main artefacts for concept representation and for communication, and of supporting tools for model verification and transformation.

MDE supports: (1) creation of interpretations of standards; (2) specialization of standards to industrial contexts (Fig. 1); (3) alignment of standards to organizational practices; (5) planning for certification; (6) electronic evidence management, and; (7) evidence reuse. Future, open issues to be addressed are: (1) facilitation of analysis and determination of the correspondence between different standards; (2) link of MDE-based safety certification with MDE-based development; (3) link of MDE-based evidence with argumentation, and; (4) use of MDE for management of evolutionary chains of evidence. The latter point would be the main contribution of the envisioned solution presented in this paper.

MDE has been used as basis for the development of prototypes aimed at: (1) facilitating the agreement upon the evidence to provide [9]; determining traceability between requirements and design [21]; (3) creating evidence repositories [30], and; (4) tailoring standards to specific companies, systems, and projects [31].

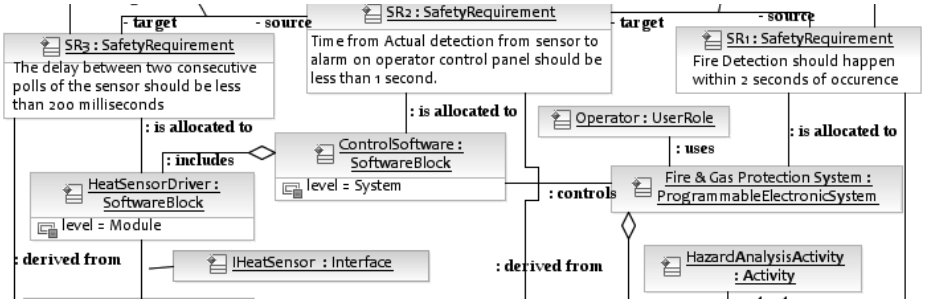

Fig. 1. Example of IEC61508-baed evidence information [29]

Model-based impact analysis is also related to MDE-based evolutionary chains of evidence. Various techniques exist for this purpose, with often differing requirements about the traceability links that need to be defined and also the semantics of the links. For example, a traceability information model and an algorithm based on this model for automatically analysing the impacts of change in UML models have been proposed in [5]. While useful, the conceptualization of the traceability links in most of the existing work is at a coarse level of abstraction, hindering their application for safety certification. A reference, better-suited approach can be found in [21], which addressed impact analysis regarding requirements and design.

2.6 Related Projects and Initiatives

When reviewing existing work on evidence management as a part of the work in OPENCOSS, and in addition to some works mentioned above, we have found several projects and initiatives that addressed or are addressing this issue:

- DECOS [6], which dealt with reuse of pre-validated hardware and software components and of functional blocks for design and certification purposes.
- EVOLVE [8], which aimed to create of a methodological framework for early V&V of evolutionary products.
- FormalSafe [11], which provided a framework to reuse development artefacts targeted at providing evidence for safety certification.
- ModelME! [20], which studied the use of MDE technologies for supporting safety certification.
- Open-DO [27], which aims to build a community around certification-oriented free software tools and libraries, addressing continuous certification.
- SafeCer [36], which aims to increase system development efficiency and reduce time-to-market by facilitating compositional certification.

More details about these and other projects can be found in [25]. Although they have addressed evidence evolution and/or management, application of MDE for evolutionary chains of evidence has seldom been explored.

3 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.

Situation 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 non-certified 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 related to other scenarios reported in research such as incremental certification and compositional/modular certification [10]. Nonetheless, the envisioned solution presented in this paper does not address adequate composition of evidence, beyond having all the necessary pieces of evidence of a chain. That is, the envisioned solution will not deal with composition adequacy assessment in a semantic way, but simply in a syntactic way (i.e., a chain of evidence must be complete). Such a semantic analysis will also be addressed in OPENCOSS, but not mainly by the authors of this paper.

Situation 2) System modification and recertification

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

In relation to tools for development of critical system, the safety assessment of the tools is not referred to as certification, but as qualification [18]. A tool is not certified "as safe", but qualified in the sense that its results (e.g., source code) can be used as evidence for safety 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 of the tool 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 new evidence to provide.

Situation 3) Modifications during the development process of a system

While a critical system is developed, and even though a waterfall process is followed, changes in a system and its associated documentation (which can be used as evidence) can occur at any moment. 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 of other piece of evidence.

Situation 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 on 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 is complete (i.e., it allows justification of the fulfilment of all the criteria of a safety standard). Such a type of approach can be found, for instance, in [31]. Nonetheless, there are cases in which adequacy assessment can be more complex, based on specific pieces of evidence that are qualitative or quantitative assessed (e.g.,

[35]). 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).

Situation 5) New context for a system

When an already certified 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 (or arguments) 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). For example, new evidence might have to be provided for a system certified against DO-178B because of the release of DO-178C.

Situation 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, in relation to Situation 1 (incomplete evidence).

Situation 7) Component reuse

The last situation presented and in which evidence for safety certification can evolve is related to component reuse in a system. Although closely related to Situation 1, 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 in Situation 1, semantic analysis (of a component-based chain of evidence) is out of the scope of the envisioned solution presented in this paper.

4 Envisioned Solution

This section outlines the envisioned solution for model-based evolutionary chains of evidence. More concretely, a (research) process for realising the solution is presented. In addition, MDE technologies such as those described in [30, 31] will be used as a reference for the development of the tool support resulting from the solution. These technologies might be also combined with non-MDE ones (e.g., with [41]).

The process consists of six activities: (1) specification of the lifecycle of a chain of evidence; (2) identification of chains of evidence in safety standards; (3) impact analysis of the change of a piece of evidence on the rest of pieces of a chain; (4) validation of the chains identified; (5) analysis of the chains of evidence in actual projects, and; (6) determination of how the chains can be mapped into the common certification language specified in OPENCOSS. An activity that is not described is the evaluation of the (improvement) effect of the solution on practice.

Although the process is presented sequentially, backward steps might be necessary as the solution is developed. For example, "validation of the chains identified" might result in the discovery of some new piece of evidence of a chain. Some activities might also be performed in parallel. For example, "determination of how the chains can be mapped into the common certification language" can be executed at any moment of the process, which will be performed in parallel to the OPENCOSS tasks aimed at specifying the language.

The activities of the process are described as follows.

1) Specification of the lifecycle of a chain of evidence

The first activity will be to define and model a lifecycle for chains of evidence. Although no proposal for such a lifecycle exists yet, we plan to base it on existing proposal for evidence lifecycle. We will focus on the lifecycle proposed in the safety assurance evidence metamodel by OMG [24] because of being a standard. Nonetheless, we will also analyse other alternatives in order to try to specify the most suitable lifecycle for chains of evidence. We will study current practice (i.e., other lifecycles for evidence or chains of evidence used in industry, such as the one proposed by GoedelWorks [2]) and the notion of (software) certificate [34, 38].

The main issue for this activity will be to determine how evidence lifecycle relates to the lifecycle of a chain of evidence, having to address the possible needs found. In addition, since automation of management of chains of evidence is planned, we will have to analyse which transitions between states might be fully automatic. Others might require validation by users. In this sense, we think that fully automation will depend on the chains of evidence (i.e., the evidence types of its pieces). For example, a change in a requirement can automatically make its associated test case inadequate. Indeed, tools such as Parasoft Concerto [33] provide this functionality. However, human intervention might necessary, for instance, in scenarios related to the change of the confidence on a piece of evidence.

2) Identification of chains of evidence in safety standards

The second activity will aim to discover chains of evidence. For this purpose, (1) existing metamodels of safety standards (e.g., [29, 42]) will be used, and/or (2) metamodels for relevant standards will be created (e.g., for CENELEC standards of the railway domain), and subsequently used.

For each relation between two entities of the metamodel, it will have to be determined if the change of one of the entities can affect the other. For example, and using Fig. 2 as a reference, if (an instance of) "Source Code" changes, then its associated "Software Module Testing" will not be adequate. In addition, a finer analysis might be necessary. Once the chains of evidence have been identified, we will have to analyse what characteristics of the evidence types (i.e., attributes of the entities) can make a chain inadequate as a results of a change. That is, a change in some attributes might not have any impact on the adequacy of a chain of evidence.

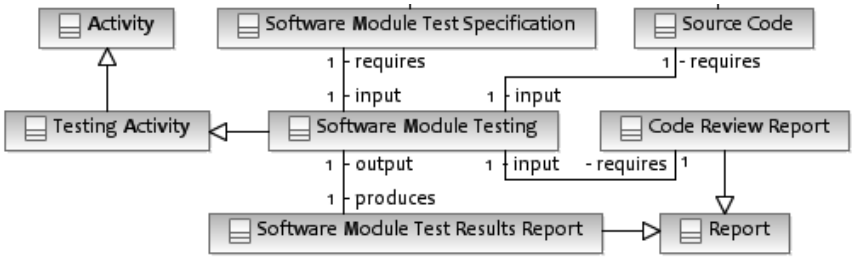

Fig. 2. Fragment of an IEC61508-based metamodel regarding software module testing [29]

3) Impact analysis of the change of a piece of evidence on the rest of pieces of a chain of evidence

After identification of the chains of evidence, mechanisms for model-based impact analysis must be determined in order to assess the effect that the change of a piece of evidence of a chain will have on the rest of pieces of the chain.

The most basic mechanism will be the specification of constraints (probably in the form of OCL [23]) aimed at enforcing the syntactic correctness of a chain of evidence. Evaluation of such constraints can automatically detect if some piece of evidence of a chain is missing.

Impact analysis related to, for instance, the change of the confidence on a piece of evidence will require further study. Using existing works as a reference (e.g., [5, 21]), we will have to decide on the most suitable and precise way to assess change impact. Probabilistic-based approaches such as the one proposed in [35] seem to be a promising possibility. However, it is based on quantitative assessment, which might pose challenges related to elicitation of expert knowledge. An alternative is qualitative assessment (e.g., [24]). Even a combination of both types of approaches might be the most suitable solution.

For deciding on the final alternative to adopt, we think that we will need input from practitioners in relation to (1) how they assess evidence adequacy, and (2) how they would like to do it, if they consider that improvements are necessary. At the end, the goal is to develop a solution that fits practice and meets industry needs and wishes.

4) Validation of the chains of evidence identified

Another activity that will follow the identification of chains of evidence is their validation. Even though we find (potentially) relevant chains, they might not be so in practice. At the same time, we might miss some chain when analysing the metamodels of the standards.

Two tasks are planned for validation of the chains of evidence. First, we will aim to obtain feedback from practitioners (both suppliers and certifiers). They will indicate if the chains identified are so in practice, as well as how they deal with their evolution. Second, we will aim to analyse data from real projects in order to determine if the chains can be found in documentation of past projects, and how traceability was kept (e.g., by means of hyperlinks in electronic documentation).

When interacting with practitioners, we will also study the development tools that they use and allow them to generate evidence (e.g., V&V tools). The tool support resulting from the development of the solution will be integrated with tools used in the development process (external tools) in order to automatically collect evidence.

5) Analysis of the chains of evidence in actual projects

The next activity will aim to analyse how the chains of evidence are instantiated in new, actual projects. This activity will be facilitated in OPENCOSS, in which three different case studies will be conducted to initially evaluate the solutions proposed in the project (including the one presented in this paper).

In addition, we will try to reach other companies that might be interested in the solution developed. For this purpose, we will make use of our industry network, for instance, in the maritime and energy domain.

We expect that it will be necessary to tailor (the metamodels of) the safety standards to the specific projects used in this activity. That is, specific interpretations and instantiations of the standards will be necessary.

6) Determination of how the chains of evidence can be mapped into the common certification language

The last activity will correspond to the mapping of the chains of evidence identified in specific standards to the common certification language defined in OPENCOSS. Otherwise, a cross-domain solution would not be provided.

The chains of evidence must be reflected and supported, in an abstract way, by the common certification language. In addition, the solution must be flexible and customizable, allowing adaptation of the chains of evidence to the specific characteristics of a development/certification project (e.g., requirements imposed by a certification authority).

A goal of the common certification language is to facilitate cross-domain (or cross-standard) certification. The language must help practitioners to determine, on the basis of a set of evidence compliant with a given standard, the degree of compliance with another standard. As a result, the need of providing new evidence could be indicated. In relation to chains of evidence, the common certification language must make their cross-domain correspondence possible. Therefore, (1) the language must support (cross-domain) chains of evidence (i.e., the relations between evidence types of a chain must be reflected in the language), and (2) it must be possible to determine, for a given standard, how its chains of evidence correspond to the ones of the common certification language.

5 Challenges

The previous section has outlined our envisioned solution for model-based evolutionary chains of evidence. However, a realisation of the solution might be curtailed because of the existence of challenges (and open issues) related to execution of the process described and to the adoption of the solution.

We have identified the following eight main challenges.

1) Involvement of practitioners. Practitioners (both system suppliers and certifiers) will have to participate in the project for (1) validation of the interpretations of the standards, (2) validation of the chains of evidence identified, and (3) provision of input about current industry practices and needs. Otherwise, the solution might not fit practice and thus might not be accepted in industry.

2) Development of a common, cross-standard, and cross-domain solution. OPENCOSS aims to provide common solutions for the railway, avionics and automotive domain. Therefore, this aspect must be taken into account in the solution, which must be suitable for the three domains. Each domain has its own standards, with a different approach and terminology. There is certainly an overlap, but they are different from a certification point of view.

3) Need of agreement with certification authorities. Also in relation to their involvement, it is essential that certifiers agree upon the solution. For example, they should agree upon and accept the results of impact analysis provided by the solution.

4) Intellectual property issues. This challenge is related to the need of (1) using data from actual projects, and (2) being provided with suppliers and certifiers' know-how. In both cases, sensitive information must be properly handled.

5) Immature MDE technology. Based on past experience, we think that some problems might arise as a result of the use of some MDE technologies. For example, we might face problems regarding model scalability, transformation and management.

6) Evidence collection from external tools. Although this challenge has been and is being addressed in other projects (e.g., [14]), the need of collecting evidence from external tools can pose interoperability problems in the tool support for the solution.

7) Impact of changes in a chain of evidence on arguments, and vice versa. An aspect that will require further study is the possible relationships between chains of evidence and arguments, and how their changes can affect each other. This might also affect safety case development and maintenance.

8) Determination of the best-suited perspective for impact analysis. So far, we have focused on information-based impact analysis (i.e., based on the information provided as evidence). However, it must be determined if an activity-based perspective would be more suitable for industry. That is, practitioners might prefer to explicitly know what activities they have to (re)execute for having adequate evidence.

Finally, Table 1 shows a summary of the impact of the challenges on the solution. Such an impact indicates if the corresponding challenge can hinder development, validation, or acceptance by industry of the solution.

Table 1. Summary of the impact of the challenges

	Challenge							
Aspect affected	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Development		X	X	X	X	X	X	X
Validation	X		X	X				
Industry Acceptance	X	X	X		X	X		X

6 Conclusions and Future Work

Safety assurance and certification can become very costly as a result of changes in the development and certification processes of a system, or in the system itself. Industry thus needs effective and efficient means that support identification of the evidence that becomes inadequate after such changes, and of the new evidence to provide.

This paper has presented a possible solution to deal with evidence and chain of evidence evolution. The solution will be developed as part of the OPENCOSS project,

and is mainly based on the use of model-driven technology. The suitability of this technology can be argued on the basis of current practice and past research.

For realising the solution, we plan to (1) define the lifecycle of a chain of evidence (2) identify chains of evidence in safety standards, (3) analyse the impact of the changes of a piece of evidence on the rest of pieces of a chain, (4) validate the chains, (5) analyse the chains in actual projects, and (6) determine how the chains of evidence can be translated in an abstract, common certification language. We have also identified eight challenges that could hinder development, validation, and acceptance by industry of the solution.

As future work, we plan to continue working on the development of the envisioned solution presented in this paper. Therefore, modifications might be made based on, for instance, the challenges faced. Once the solution has been implemented, it will be validated in case studies as part of the work in OPENCOSS. Validation will allow us to assess the actual, potential improvements that the solution can provide to industry.

Acknowledgments. The research leading to these results has received funding from the FP7 programme under grant agreement n° 289011 (OPENCOSS) and from the Research Council of Norway under the project Certus SFI. The authors would also like to thank the OPENCOSS partners who have provided information and feedback about evidence evolution, chains of evidence, and possible solutions to manage them, and Leon Moonen for his suggestions regarding impact analysis literature.

References

1. Altreonic: Survey on Certification Issues (online) http://www.altreonic.com/content/survey-certification-issues (Accessed May 15, 2012)
2. Altreonic: Trustworthy Systems Engineering with GoedelWorks (online) http://www.altreonic.com/category/products/goedelworks (Accessed May 15, 2012)
3. Atego Workbench: www.atego.com/products/atego-workbench/ (Accessed May 15, 2012)
4. Bohner, S.A., R.S. Arnold: Software Change Impact Analysis. IEEE Press (1996)
5. Briand, L., Labiche, Y., Yue T.: Automated traceability analysis for UML model refinements. Information & Software Technology 51(2): 512–527 (2009)
6. DECOS project: http://www.decos.at (Accessed May 15, 2012)
7. Ericson, C.A.: Concise Encyclopedia of System Safety. Wiley (2011)
8. EVOLVE project: http://www.evolve-itea.org (Accessed May 15, 2012)
9. Falessi, D., et al.: Planning for Safety Evidence Collection. IEEE Software 29(3): 64-70 (2012)
10. Fenn, J., et al.: The Who, Where, How, Why And When of Modular and Incremental Certification. In: 2nd IET International Conference on System Safety (2007)
11. FormalSafe project: http://www.dfki.de/web/research/projects/base_view?pid=456 (Accessed May 15, 2012)
12. Habli, I.M.: Model-based assurance of safety-critical product lines. PhD thesis, University of York (2009)
13. Herrmann. D.S.: Software Safety and Reliability. IEEE Press (1999)
14. iFEST project: http://www.artemis-ifest.eu (Accessed May 15, 2012)

15. Jackson, D., Thomas, M., Millet, L.I.: Software for Dependable Systems. NAP (2007)
16. Johansson, M., Nevalainen, R.: Additional requirements for process assessment in safety–critical software and systems domain. J. Softw. Maint. Evol., doi: 10.1002/smr.499 (2010)
17. Kelly, T. P.: Can Process-Based and Product-Based Approaches to Software Safety Certification be Reconciled? In: Improvements in Systems Safety. Springer (2008)
18. Kornecki, A., Zalewski, J.: Certification of software for real-time safety-critical systems: state of the art. Innovations in Systems and Software Engineering 5(2) 149-161 (2009)
19. Medini Analyze: http://www.ikv.de/index.php/en/products/functional-safety (Accessed May 15, 2012)
20. ModelME! project: http://modelme.simula.no/ (Accessed May 15, 2012)
21. Nejati, S., et al.: A SysML-Based Approach to Traceability Management and Design Slicing of Safety Certification. Info. & Software Technology (accepted paper) (2012)
22. OMG: Argumentation Metamodel (ARM) 1.0 – Beta 1 (online) http://www.omg.org/spec/ARM/ (2010) (Accessed May 15, 2012)
23. OMG: Object Constraint Language (OCL) Version 2.3.1 (online) http://www.omg.org/spec/OCL/2.3.1/ (2006) (Accessed May 15, 2012)
24. OMG: Software Assurance Evidence Metamodel (SAEM) 1.0 – Beta 1. (online) http://www.omg.org/spec/SAEM/ (2010) (Accessed May 15, 2012)
25. OPENCOSS: Deliverable D6.1 - Baseline for the evidence management needs of the OPENCOSS platform (2012)
26. OPENCOSS: http://www.opencoss-project.eu/ (Accessed May 15, 2012)
27. Open-DO initiative: http://www.open-do.org/ (Accessed May 15, 2012)
28. Oxford Dictionaries: evidence (online) http://oxforddictionaries.com/definition/evidence?q=evidence (Accessed May 15, 2012)
29. Panesar-Walawege, R.K., et al.: Characterizing the Chain of Evidence for Software Safety Cases: A Conceptual Model Based on the IEC 61508 Standard. In: ICST 2010
30. Panesar-Walawege, R.K., et al.: CRESCO: Construction of Evidence Repositories for Managing Standards Compliance. In: ER 2011 Workshops
31. Panesar-Walawege, R.K., Sabetzadeh, M., Briand, L.: Using UML Profiles for Sector-Specific Tailoring of Safety Evidence Information. In: ER 2011
32. Panesar-Walawege, R.K., Sabetzadeh, M., Briand, L.: Using Model-Driven Engineering for Managing Safety Evidence: Challenges, Vision and Experience. In: WoSoCER 2011
33. Parasoft Concerto: http://www.parasoft.com/jsp/products/concerto/home.jsp (Accessed May 15, 2012)
34. Programatica project: http://programatica.cs.pdx.edu/index.html (Accessed May 15, 2012)
35. Sabetzadeh, M., et al.: MODUS: A goal-based approach for quantitative assessment of systems (online) http://modelme.simula.no/assets/modus.pdf (Accessed May 15, 2012)
36. SafeCer project: http://www.safecer.eu/ (Accessed May 15, 2012)
37. Schmidt, D.C.: Model-Driven Engineering. IEEE Computer 39(2): 25-31 (2006)
38. Sherriff, M., Williams, L.: DevCOP. In: ISSRE 2006
39. Sommerville, I.: Software Engineering, 7th ed. Pearson (2004)
40. Squair, M.J.: Issues in the Application of Software Safety Standards. In: SCS'05
41. The Qualifying Machine: In [27]
42. Zoughbi, G., Briand, L., Labiche, Y.: Modeling safety and airworthiness (RTCA DO-178B) information: conceptual model and UML profile. SoSyM 10(3): 337-367 (2011)