An Industrial Survey of Safety Evidence Change Impact Analysis Practice

Jose Luis de la Vara, Markus Borg, *Member, IEEE*, Krzysztof Wnuk, and Leon Moonen, *Member, IEEE*

Abstract—*Context.* In many application domains, critical systems must comply with safety standards. This involves gathering safety evidence in the form of artefacts such as safety analyses, system specifications, and testing results. These artefacts can evolve during a system's lifecycle, creating a need for change impact analysis to guarantee that system safety and compliance are not jeopardised. *Objective.* We aim to provide new insights into how safety evidence change impact analysis is addressed in practice. The knowledge about this activity is limited despite the extensive research that has been conducted on change impact analysis and on safety evidence management. *Method.* We conducted an industrial survey on the circumstances under which safety evidence change impact analysis is addressed, the tool support used, and the challenges faced. *Results.* We obtained 97 valid responses representing 16 application domains, 28 countries, and 47 safety standards. The respondents had most often performed safety evidence change impact analysis during system development, from system specifications, and fully manually. No commercial change impact analysis tool was reported as used for all artefact types and insufficient tool support was the most frequent challenge. *Conclusion.* The results suggest that the different artefact types used as safety evidence co-evolve. In addition, the evolution of safety cases should probably be better managed, the level of automation in safety evidence change impact analysis is low, and the state of the practice can benefit from over 20 improvement areas.

Index Terms—Safety-critical system, safety evidence, change impact analysis, state of the practice, survey research

1 Introduction

Society increasingly depends on complex computer-based and software-intensive systems. They penetrate many aspects of our daily life, such as transport, energy, and healthcare, and their malfunction can have considerably negative consequences. Many of these systems are safety-critical and subject to some form of safety assessment by a third party (e.g., a certification authority) in order to ensure that the systems do not pose undue risks to people, property, or the environment. This includes an analysis of how software contributes to system safety and of software safety risks. A common type of assessment is compliance with safety (or safety-related) standards, usually referred to as safety certification [23]. Examples of safety standards used in industry include IEC 61508 for electrical, electronic, and programmable electronic systems in a wide range of industries, and more specific standards such as DO-178C for avionics, the CENELEC standards for railway (e.g., EN 50128), and ISO 26262 for the automotive sector [48].

- *J.L. de la Vara is with the Computer Science Department, Carlos III University of Madrid, Avda. de la Universidad 30, 28911 Leganes, Madrid, Spain. E-mail: jvara@inf.uc3m.es.*
- *M. Borg is with the Software and Systems Laboratory, SICS Swedish ICT AB, Ideon Science Park, Building Beta 2, Scheelevägen 17 SE-223 70, Lund, Sweden. E-mail: markus.borg@sics.se.*
- *K. Wnuk is with the Software Engineering Research Lab, Blekinge Institute of Technology SE-371 79, Karlskrona, Sweden. E-mail: krzysztof.wnuk@bth.se.*
- *L. Moonen is with the Certus Centre for Software V&V, Simula Research Laboratory, P.O. Box 134 1325, Lysaker, Norway. E-mail: leon.moonen@computer.org.*

Manuscript received 11 June 2015; revised 9 Mar. 2016; accepted 3 Apr. 2016. Date of publication 10 Apr. 2016; date of current version 16 Dec. 2016. Recommended for acceptance by G.H. Travassos. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TSE.2016.2553032

Demonstrating compliance with a specific standard involves gathering and providing convincing safety evidence [35], defined as artefacts that contribute to gaining confidence in the safe operation of a system and that are used to show the fulfilment of the criteria of a safety standard [48]. Examples of artefact types that can be used as safety evidence include safety analysis results, system specifications, testing results, reviews, and source code.

Many of the artefacts used as safety evidence evolve during a system's lifecycle, including software artefacts. As a consequence, the corresponding changes must be managed and impact analysis might be necessary in order to guarantee that the changes do not jeopardise system safety or compliance with a standard [26]. In software engineering, impact analysis can be defined as the activity that aims at identifying the potential consequences of a change in some software product [4]. By Safety Evidence Change Impact Analysis (SECIA), we refer to the activity that attempts to identify the potential consequences of a change in the body of safety evidence [46], [51]. This body constitutes the collection of artefacts managed as safety evidence for a system, usually a large set of artefacts that is difficult to overview. Possible consequences of a change can be the need for adding, modifying, or revoking safety evidence artefacts. Changes during system development, system modification and re-certification, and component reuse are examples of situations in which SECIA can be necessary [12].

Change impact analysis (hereafter referred to as impact analysis) is a crucial activity in the lifecycle of any safety-critical system. Indeed, it is prescribed in most of the safety standards used in industry, e.g., [10], [26], [45], [58]. However, the standards do not explain in detail how to perform an impact analysis, but just provide general guidance [13], [28]. In some cases, the standards do not even clearly state

when impact analysis should be performed. This lack of clarity can lead to an inadequately performed analysis resulting in overlooked impact. Examples of accidents, or near-accidents, because of inadequate impact analysis can be found in practically every application domain, e.g., [24], [28], [29], [37], [68], from classical examples such as the Ariane 5 accident to recent airplane crashes.

Although safety evidence management and impact analysis are two research areas that have received significant attention in the last decades, previous research barely reflects on the state of the practice. The number of publications that report insights into how practitioners deal with these activities is low [36], [47], and there is a lack of publications that study how industry addresses SECIA. Previous studies focused on specific practices related to a reduced set of companies (e.g., the partners of a specific project [50]), standards (e.g., only IEC 61511 [5]), domains (e.g., automotive [16]), or artefact types (e.g., requirements and test cases specifications [2]). Therefore, a comprehensive picture of current SECIA practices does not exist. Without this knowledge, it is difficult to effectively determine industry needs and to shape future research towards them.

This paper presents a survey aimed at gaining insights into how SECIA is addressed in practice. We designed a web-based questionnaire targeted at practitioners that were or had been involved in SECIA. This includes people who provide, check, or request safety evidence. We asked questions about the circumstances under which SECIA was addressed, the tool support used, and the challenges faced. We obtained 97 valid responses from 16 application domains, 28 countries, 47 safety standards, nine types of organizations, and five overall roles.

To the best of our knowledge, this survey is the largest empirical study, to date, concerning the state of the practice on safety evidence management and on impact analysis for safety-critical systems. Therefore, this work provides strong empirical evidence on SECIA practices that should help academia to identify areas in which further research is necessary. Practitioners can benefit by gaining new insights into how they can or should deal with SECIA, and use the survey results as a benchmark for their own practices.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the research method. Section 4 presents the results and our interpretation. Section 5 summarises our conclusions. Appendices A to D can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TSE.2016.2553032.

2 Related Work

Related work is divided into general literature on impact analysis, whose insights can apply to safety evidence, and specific literature on impact analysis for safety-critical systems. Related work indicates artefact types that might be involved in SECIA, possible tool support and its characteristics, and possible challenges. Special attention is given to publications that have provided insights into the state of the practice on impact analysis and SECIA.

Fig. 1 presents the main concepts of the survey and thus of the review of related work, and the relationships between them. The figure aims to facilitate paper understanding. Appendix A (survey questionnaire), available in the online supplemental material, includes definitions and examples of the artefact types. The rest of concepts are introduced in this or the previous section.

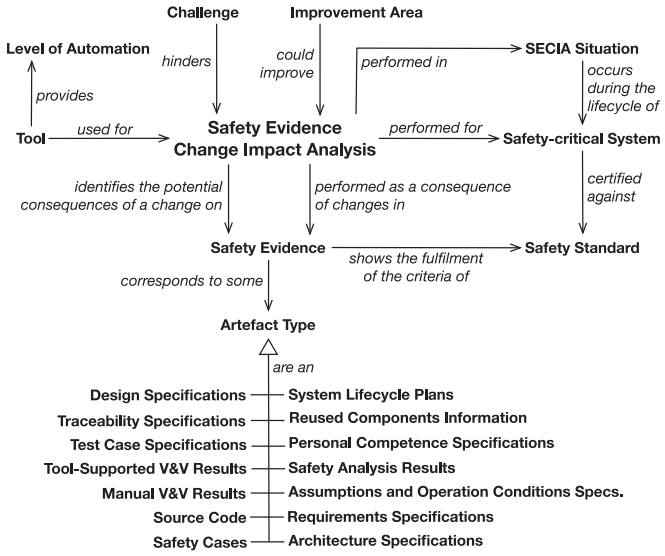

Fig. 1. SECIA conceptual model.

2.1 General Literature on Impact Analysis

Impact analysis has been the subject of extensive research for the last four decades, especially in the context of software evolution and software maintenance [4].

Most research on impact analysis has focused on source code [36], studying both change effects between source code artefacts and on other artefact types (e.g., test cases to re-execute after a change). Another area that has received significant attention is impact analysis for requirements, especially during requirements management and traceability [30]. Impact analysis for architecture specifications [27], software components [70], or test cases [2] has also been frequently investigated.

Carrillo-de-Gea et al. analysed requirements management tools and their support for requirements change management [9]. Li et al. assessed the state of tooling for impact analysis on source code [38], and report that most tools are academic prototypes and that only JRipples seems to be stable and mature.

According to Jamshidi et al. [27], most of the research on architecture-centric software evolution provides impact analysis tool support, at times with full automation. The literature also reports on the extension and adaptation of commercial tools for impact analysis purposes [67]. Regarding the current approaches for automated traceability and impact analysis, they have been validated on small data sets [7] and limited artefact types [49]. Therefore, their actual support to industrial needs remains unconfirmed because practitioners usually have to deal with tens of artefact types and thousands of artefacts for SECIA [48]. Buckley et al. suggest that some manual work is always necessary for impact analysis [8].

The state of the practice in software impact analysis is reported in several publications. Goeritzer conducted a case study in industry and report that most software engineers

manually perform impact analysis on source code and would like to have further tool assistance [21]. Tao et al. focused on how software engineers understand software source code changes [64]. This study reports the need for more tool support and the difficulty in determining (1) the completeness and consistency of a change and (2) the effect on other software components. Babar et al. conducted a survey on the usefulness of design rationales for software maintenance and conclude that documenting the rationale can facilitate the identification of the elements impacted by a change [1]. Rovegård et al. interviewed software engineers and report impact analysis challenges related to the lack of resources, the need for experience and expertise, inadequate traceability, insufficient tool support, and the need for more structured information [57]. They suggest a number of improvement areas, which include arranging meetings to discuss impact analysis and introducing tool and method support.

A recent survey on requirements volatility indicates that requirements changes have recurring nature and evaluating their consequences can be complex and time-consuming [18]. Other authors have reported challenges related to requirements change impact, such as the need for having several development roles involved to properly understand the impact [71] and difficulties in accurately predicting change management cost [39]. Challenges in tracing requirements and test cases and in maintaining alignment between them have been identified in a survey of six companies [2].

2.2 Impact Analysis for Safety-Critical Systems

Publications focusing on impact analysis for safety-critical systems have dealt with the evolution of safety-targeted artefact types (e.g., safety cases), tool support, and safety-specific concerns and challenges in system evolution.

Safety cases are arguably among the main evidence types for a safety-critical system. They are a documented argument aimed at providing a compelling, comprehensive, and valid case that a system is acceptably safe for a given application in a given operating environment [32]. Safety cases can be provided in only text or also with structured graphical representations in order to more clearly show how evidence supports the main arguments about the truth of a system's safety claims [47]. These safety arguments are typically based on the measures taken for ensuring that technical safety risks have been mitigated or avoided. The evolutionary nature of safety cases has been discussed in previous works such as [32], [44], [63], which indicate that safety arguments should evolve and be created incrementally as system development progresses.

Conducting SECIA using safety cases can be very challenging because they typically contain hundreds of references to other artefacts for supporting their safety arguments, and these artefacts evolve during a system's lifecycle. Prior work studied the evolution of safety analyses and assessments [40] and the possible impact of architectural changes in safety cases [3]. Recent models for safety certification explicitly address SECIA needs (e.g., [13]), such as the specification of the effects that a change in an artefact type can have in other types.

According to Lloyd and Reeve [41], widely available tools can facilitate impact analysis of safety-critical systems, and change management can be tracked with workflow tools or wikis. The authors also argue for the suitability of manual procedures. However, we conjecture that such procedures will be too time-consuming and error-prone, and can have problems of scalability. ASCE and Reqtify are examples of commercial tools that have been referred to in the SECIA literature [35], [44], where Reqtify was used only in an avionics hardware development project.

An important aspect regarding tool support is tool qualification [35], a formal assurance of output suitability. In many domains, the artefacts that a tool produces during a safety-critical system's lifecycle need to be formally reviewed unless the tool is qualified, including SECIA tools. In this sense, tools can be regarded as safety-critical because their malfunction can lead to safety risks. As an example, Reqtify is formally qualified for avionics and railway.

A survey with 52 practitioners [48] precedes our current study. The previous survey studied general safety evidence management practices regarding the information provided as evidence, evidence change management, structuring of evidence, evidence adequacy assessment, and challenges in evidence provision.

To better understand change management, the previous survey asked how the effect on other pieces of evidence was checked when a piece changed and whether details about how the change of a piece of evidence had affected others were managed. These aspects overlap with and are studied in more depth in this paper. The survey also asked how the degree of evidence completeness was checked and how traceability between different pieces of evidence was recorded.

The previous survey suggests that evidence change management is mainly performed manually and highlights the need for further analyses. Whereas the previous survey investigated general aspects of change management for safety evidence, the current study describes a completely new survey that was conducted to explore specific artefact types in depth. In addition, the current study provides novel insights into SECIA-specific situations, challenges, and tool support. Finally, the population of the survey reported in this paper is a subset of the population for [48]: practitioners involved in safety evidence management in general versus practitioners involved in SECIA, a part of safety evidence management.

Surveys among the partners of industry-academia research projects [50], [59] have reported tools for the development and assurance of safety-critical systems suitable for impact analysis and change management purposes (e.g., Reqtify and VectorCAST). Although their contributions are valuable, these surveys focus on safety evidence management in general, and not on, for instance, how often different artefact types trigger SECIA. An interview study with engineers from four companies in different application domains [53] reports on the execution of safety analysis activities after requirements changes and on the need for allocating sufficient resources to handle change and for awareness of change impact on system safety.

Other authors have analysed information from previous projects to study impact analysis for safety-critical systems. Borg et al. analysed over 10,000 impact analysis reports from a company in the power and automation domain [5]. The authors identified both source code and other artefact types (e.g., requirements, design specifications, and test cases) involved in source code impact analysis in the past.

Case studies in the automotive domain indicate the advantages of adequate architecture structures for guiding impact analysis [16], challenges for change management in relation to tool support and to systematic testing procedures [31], and the use of safety cases as an impact analysis tool in system changes and with respect to system safety [65]. In the medical domain, problems related to traceability (e.g., unclear trace granularity) have been reported [42], as well as previous system failures and issues such as incomplete impact analysis and insufficient verification and validation (V&V) after changes [68].

Other identified challenges in impact analysis for safety-critical systems include: the impact of component reuse and evolution on safety [14], determining if a component can be reused [25], the vast amount of artefacts to trace and the need for safety assessors' confidence [46], the need for planning and documenting impact analysis [55], and the difficulty in ensuring system safety after a change [66].

To summarize, the main differences between our survey and related work are as follows:

1) Prior SECIA-related empirical studies have dealt with a reduced number of application domains, countries, and safety standards.
2) Previous research has acknowledged the existence of many phenomena (e.g., artefact types involved in impact analysis or challenges faced by practitioners), but does not provide insights into how often the phenomena occur in SECIA.
3) Most prior work has only studied single or a reduced number of artefact types (e.g., source code).
4) Very little information exists about the tools used for SECIA in industry, and this information is practically non-existence for particular artefact types (e.g., assumptions and operation conditions).

We have used observations in related work for creating the survey questionnaire (see Section 3.2) and discussing the results (Section 4). We also use the lack of information in related work for result discussion.

3 Research Method

We utilized the survey approach and employed a web-based questionnaire because of the following main advantages [17], [48], [54], [62]:

1) They allow us to understand the views of many individuals that work in different companies or industries in a unified way
2) They support data collection for many variables in a short time
3) They offer unified data collection framed by survey questions
4) They bring the potential of collecting a larger number of responses than with interviews
5) When compared to interviewing practitioners in our industry network, a wider and more heterogeneous sample can be reached by advertising the survey in different industry-oriented forums

Prior work delivers limited understanding of how SECIA is handled in practice. The available theories around SECIA are either partial for some phenomena or inexistent for other phenomena. For example, there is evidence that SECIA can be performed when a component is reused but not of how often it happens, and there is no evidence of the level of automation of SECIA from manual V&V results. To address this gap, we designed an exploratory survey aimed at investigating how SECIA is performed within its industrial context and at seeking new insights, ideas, and possible hypotheses for future research [56]. We collected and analysed both quantitative and qualitative data provided by practitioners via a self-administered questionnaire.

We used the recommendations on surveys in software engineering research by Kitchenham and Pfleeger [34] as the main basis for defining and executing the research process. Some adaptation was necessary because of aspects specific to this survey, such as the analysis of free-text questions and the use of a social network for sampling.

The following sections present the research questions, survey design, instrument evaluation, data collection, data analysis, and validity. Further details about the research method can be found in [11].

3.1 Research Questions

The goal of the survey was to gain insights into how industry deals with SECIA. As explained in Section 2.2, aspects that characterise how SECIA is addressed include when it is performed (e.g., for component reuse), the artefact types involved, the tool support used and the level of automation that it offers, and the challenges faced. The goal was decomposed into the following Research Questions (RQs).

RQ1. Under what circumstances is safety evidence change impact analysis addressed?

RQ1.1. How often do these circumstances occur?

The purpose of RQ1 and subsequent RQ1.1 is to explore the circumstances during a system's lifecycle when SECIA is actually conducted (general situations, and SECIA from and on specific artefact types), and how often these circumstances occur. For example, system re-certification is acknowledged as a situation in which evidence evolves and thus SECIA might be necessary [12]. However, the information about the frequency of this situation in industry has not been provided. Moreover, we aimed to study the artefact types that trigger SECIA and the artefact types affected by the changes. To the best of our knowledge, no publication has studied a large range of artefact types that can be involved in SECIA, or if some artefact types trigger SECIA more often than others.

RQ2. What tool support for safety evidence change impact analysis is currently used?

The purpose of RQ2 is to collect the information about the current level of automation for SECIA and the tools currently used by industry. Such tools include those used for storing evidence of safety evidence change management. There is little knowledge about SECIA supporting tools in relation to, for instance, safety cases. We have found only ASCE in the literature [44], but without evidence of use in practice, see Section 2.2 for details.

RQ3. What challenges are faced when dealing with safety evidence change impact analysis?

RQ3.1. How often are the challenges faced?

RQ3.2 How could safety evidence change impact analysis be improved?

The purpose of RQ3 is to explore the current issues in industry regarding SECIA. Many different SECIA challenges are acknowledged in the literature (see Section 2.2 for details), but there exists no in-depth study yet on how often practitioners face them and how practitioners consider that state-of-practice SECIA could be improved.

We acknowledge that further phenomena can be studied to gain insights into how industry deals with SECIA, such as the activities executed and the roles involved. In this survey, designed according to an expected completion time of 20 minutes (see Section 3.2), we decided to prioritise the above RQs.

3.2 Survey Design

We designed a structured cross-sectional web-based survey [34], aimed at obtaining information from the participants at a fixed point in time based on their previous experience in dealing with SECIA. We used SurveyMonkey (https://www.surveymonkey.net) as supporting tool. Appendix A, available in the online supplemental material, contains the final questionnaire.

The survey was targeted at practitioners that were or had been involved in SECIA. This included people who provided safety evidence (e.g., safety engineers or testers of a company that supplies components), people who checked safety evidence (e.g., an independent safety assessor), and people who requested safety evidence (e.g., a person that represents a certification authority). These professionals correspond to the target population. To ensure that we obtained valid information about practice, we explicitly provided this characterisation of the target population as well as the definition of SECIA in the introduction of the questionnaire. We also gathered the level of experience in SECIA (number of projects and years; Q7 and Q8), and asked about how often certain phenomena had happened (i.e., in how many projects; e.g., in Q9). Section 4 provides further details about the roles of the organizations and of the respondents of the survey sample.

The questionnaire was created taking related work into consideration. We adopted and adapted information in relation to:

- Respondents' background [48] (Q2-Q8);
- SECIA situations [12] (Q9);
- Artefact types that can be used as safety evidence (Q11, Q13, Q15 and Q17), by synthesising and selecting artefact types from a taxonomy of safety evidence [48] (from 70 to 14 artefact types; e.g., Manual V&V Results as a generalisation of Inspection Results and Review Results)
- Likert scales on frequency [61] (Q9, Q11, Q13, and Q20);
- Levels of automation [52] (Q15), and;
- Challenges in impact analysis and SECIA (Q20):
  - Difficulty in estimating the effort required to manage a change (e.g., [14]);
  - Too coarse granularity of the traceability between artefacts to accurately know the consequences of a change (e.g., [18])
  - Excessive detail of the traceability between artefacts, making traceability management more complex than necessary for impact analysis purposes (e.g., [25])
  - Unclear meaning of the traceability between artefacts in order to know how to manage a change (e.g., [42])
  - Insufficient traceability between artefacts to accurately know the consequences of a change (e.g., [46])
  - Long time for evaluating the consequences of a change (e.g., [46])
  - Insufficient confidence by assessor or certifiers in having managed a change properly (e.g., [46])
  - Vast number of artefacts to trace (e.g., [48])
  - Insufficient tool support (e.g., [48])
  - Lack of a systematic process for performing impact analysis (e.g., [49])
  - Difficulty in determining the effect of a change on system safety (e.g., [49])
  - Difficulty in deciding if a component can be reused (e.g., [57])
  - Difficulty in assessing system-level impact of component reuse (e.g., [66])

The pages and the options of the questions were presented in a randomized order to mitigate threats to validity, particularly errors and omissions due to respondents' fatigue. Definitions and clarifications were provided for those parts of the questionnaire in which the risk of misinterpretations was identified. For example, we provided examples of the artefact types used as safety evidence the first time they appeared in a questionnaire page. Respondents were given the possibility to mention other options in the questions.

3.3 Instrument Evaluation

We evaluated the survey questions in two stages (i.e., with two pilots). First, we invited two senior software engineering researchers (one of them with experience in safety-critical systems) and one safety-critical system developer to read the questionnaire and provide feedback on its readability, understandability, potential ambiguities, and length. The feedback led to the removal of four questions and to improving several (e.g., adding an explanation about internal tools in Q17 and allowing respondents to indicate "I don't know" in Q11). Second, we requested one safety assessor, one safety assurance manager, and one safety-critical system developer to complete the revised version of the questionnaire and to provide feedback on the same points. This evaluation resulted in the removal of two questions and in some minor clarifications.

The final version of the questionnaire consisted of 23 questions and it was estimated to require maximum 20 minutes to complete.

3.4 Data Collection

Data collection started on November 21st of 2013 and finished on January 11th of 2014. We advertised the survey on several LinkedIn groups related to safety-critical systems. Some groups were on specific application domains (e.g., automotive), some on specific safety standards (e.g., IEC 61508), and others on more general subjects (e.g., functional safety). The complete list of groups can be found in [11]. This advertisement was aimed at reaching a large number of

practitioners of the target population (see Section 3.2) worldwide, and with different backgrounds. Two reminders were posted on each group. The benefits of using LinkedIn have been discussed in the literature (e.g., [15]), and include the increase in subjects' heterogeneity, the increase in the level of confidence in the representativeness of a sample, and the possibility of reaching a population for which no centralized bodies of professionals exist.

In addition, we advertised the survey on two mailing lists on safety-critical systems (general-opencoss@listserver.tue.nl and systemsafety@lists.techfak.uni-bielefeld.de). We knew that some members of the lists were part of the target population. This second advertisement aimed to complement the social network advertisement, since we could not know how many practitioners would regularly check the updates on LinkedIn. One reminder was posted on each mailing list.

Finally, we contacted practitioners that we personally knew and participants of the prior survey [48] that agreed upon being contacted for follow-up studies. In both cases, we asked the practitioners to forward the invitation to additional relevant colleagues. We sent one reminder to the practitioners that we personally knew.

Regarding the size of the population, we refrain from providing an estimate because we could not sufficiently substantiate it. Even if we use the number of members of the LinkedIn groups and the mailing lists as a basis, we cannot accurately estimate the number of members involved in SECIA. The groups and the lists are on topics more general than SECIA (e.g., functional safety) and some people might be members of multiple LinkedIn groups.

3.5 Data Analysis

We obtained 129 responses, and rejected 28 of those because the respondents only completed the background information. We examined the remaining 101 responses to detect careless responses [43] that should be rejected. Responses were considered careless if they fulfilled one of the following criteria: (a) the response did not provide relevant information (e.g., the respondent only indicated "I don't know" to all the questions answered); (b) the response contained clear and significant inconsistencies (e.g., between Q9 and Q11), or; (c) the response displayed patterns for which we could not find a justification (e.g., selection of "always" for all the options of the questions about the frequency of some phenomenon in Q9).

The final number of valid responses was 97 (75.2 percent of all responses), including incomplete but non-careless responses, as long as they provided answers to some RQs. The respondents that completed the whole questionnaire, and that in our opinion did not make any interruptions (less than 40 minutes of completion time), needed 20 minutes and 47 seconds on average.

Afterwards, we reviewed the free-text responses. We unified some answers so that they had the same format. For example, DO-178 was referred to in different ways (e.g., DO178). We conducted open coding on the answers to the question about the respondents' role (Q6) and to another about how they think that SECIA could be improved (Q22). This resulted in the iterative creation of a classification for the two questions. For example, a respondent indicated "software designer and architect" as his role, which was classified first as software engineer and finally as engineer, and another respondent indicated "section manager for hardware development", which was classified first as product manager and finally as manager [11]. We provide details about the coding on how SECIA could be improved in Section 4.

The first author conducted the initial unification and coding of answers. The third author validated the outcome from answer unification and coding of respondents' role. For the answers on how to improve SECIA, the second author coded them with the codes defined by the first author in the first, initial open coding iteration. They then discussed the answers to which different codes had been assigned and the possibility of adjusting the codes and their definitions. The codes and their definitions were refined, and then the first author revised the coding scheme. The second author reviewed the outcome, both authors discussed the revision, and they finally agreed upon the final coding.

In the last step of the data analysis, we calculated Spearman's rank-order correlation coefficients [22] for the ordinal scale questions, including the questions about respondents' experience (Q7 and Q8). We aimed to study the relationship between the occurrences of the corresponding phenomena and determine if e.g., some appear to co-occur. Appendix B, available in the online supplemental material, shows an example of how the coefficients can be calculated.

3.6 Validity

We discuss validity according to the four perspectives presented by Wohlin et al. [69], complemented by survey-specific validity aspects [19], [20], [34].

Construct validity is concerned with the relationship between a theory behind an investigation and its observation. Construct validity affects the rest of the validity perspectives. As explained above, the current insights into SECIA practice are limited, thus there is not a fully developed theory yet. Nonetheless, we consider that an initial theory can be derived from prior publications (see Section 2) and we used these publications as a basis in the survey to e.g., create the questionnaire (see Section 3.2).

We guaranteed confidentiality and anonymity of the individual responses and allowed the respondents to complete the survey without identifying themselves in order to mitigate potential threats to collection of inaccurate information due to evaluation apprehension. Providing predefined lists in the questionnaire (e.g., of challenges) based on the literature on software and systems engineering is a limitation of this study. This threat was mitigated by allowing the respondents to specify additional information. Selecting a subset of SECIA phenomena to ask about (RQs topics) and discarding others (e.g., SECIA activities) affect content validity. Furthermore, the phrasing of questions can be a threat to construct validity, including face validity. We mitigated this threat by creating the questionnaire with close reference to related work and with the two-stage instrument evaluation. The background information collected contributes to criterion validity.

Internal validity deals with the relationship between a treatment and its results. We provided an introduction to the survey to make the respondent familiar with the context of the study and the kind of information to provide.

This contributes to result validity. When ambiguity could exist, we included information about the intent of the questions and definitions of the terminology used. Instrument evaluation allowed us to mitigate ambiguity and misinterpretation (instrumentation threat). Designing the survey instrument so that it could be completed in approximately 20 minutes helped to mitigate maturation. We applied a non-random sampling strategy, thus selection bias was not fully avoided. Moreover, the performance of the volunteers may be different from the entire population's performance. Although 25 percent of the responses were discarded (attrition threat), we are confident that the results provide a valid picture of SECIA in practice (see the discussion in Section 4).

Conclusion validity is concerned with obstacles to draw correct conclusions from a study. Obtaining a heterogeneous sample of respondents, of which most can be regarded as senior practitioners (five or more years or projects of experience; see Section 4), contributes to conclusion validity. Based on the recommendations by Kitchenham et al. [33], we focused on the analysis of strong (corr. > 0.59) and very strong (corr. > 0.74) correlations to identify relationships of practical importance between phenomena. The p-values of these correlations are below 1e-08. We use the lack of strong or very strong correlations and the existence of weak or very weak ones (corr. < 0.3) as indications that the relevance in practice of some relationships cannot be guaranteed. Although further correlations could have been calculated, we did not do it to avoid fishing for results.

Conclusion validity is further strengthened by observer triangulation in answer unification and coding. Nonetheless, we estimate that a minimal risk remains of having misinterpreted some free-text answers. Other threats to conclusion validity relate to the amount of free-text responses and to correlation interpretation. A low number of free-text responses impacts the extent to which a phenomenon is characterised from the survey. Readers must be careful when interpreting correlations because e.g., they do not indicate cause-effect.

External validity is concerned with the generalization of the conclusions. We believe that the results constitute a good representation of SECIA in practice. It is uncommon that a survey on a narrow topic in systems and software engineering receives almost 100 valid responses. In addition, the sample is heterogeneous, more heterogeneous than in related surveys (e.g., [48]) regarding the number of countries, application domains, and safety standards represented. Although the number of respondents from Sweden (17; see Section 4) could be considered high, we expect that it has a minor impact on external validity. Overall, the rest of the background information is similar to [48], and we argue that it sufficiently covers industry characteristics. For example, respondents' background is in line the characteristics of LinkedIn groups. The domain-specific group in which the survey was advertised with the highest number of members was on aerospace and the standard-specific group was on DO-178. The group on ISO 13849 had around a fourth of the members of the group on ISO 26262, and some emerging country-specific groups (e.g., for India) exist. The organization and respondents' roles also cover the whole value chain of safety-critical systems engineering, and we consider that USA and Europe are world leaders in safety-critical systems engineering and assurance (see e.g., [23], [32], [37], [47], [50], [55]).

4 Results and Interpretation

This section reports upon and interprets the survey results. A section has been created for each principal RQ (RQ1, RQ2, and RQ3), and these sections are decomposed into specific aspects for answering the RQs. We discuss the possible implications for research and practice and compare the results with related work. Section 4.4 presents a summary.

Tables 1 to 5 present survey results. The cells with bold text indicate the mode of the phenomenon under study (i.e., for each row), whereas the shaded cells indicate the most often reported phenomenon for each possible answer (i.e., for each column). For example, in Table 1 (frequency of situations for SECIA) the mode of *Modification of a new system during its development* is "most projects", and *Reuse of existing components in a new system* is the situation most often reported as happening in "some projects". The results are presented as frequencies in percentages (ratio of respondents) and data points (in brackets). We report all the strong and very strong correlations found between ordinal scale questions (Spearman's rank-order correlation coefficients; corr. > 0.59 and corr. > 0.74, respectively; $p < 1e-08$).

Fig. 2 summarises the respondents' demographics. For the application domains, countries, and safety standards, we only present the answers provided by three or more respondents. The complete lists and descriptions of the safety standards are available in [11]. Based on the sample characteristics, and as discussed in the next paragraphs and in Section 3.6, we consider our sample to be representative of the safety-critical system industry.

Aerospace dominates the 16 application domains represented in the survey. The respondents mentioned 47 individual safety standards, with DO-178 as the most frequent. Thirty-four respondents reported more than one safety standard. The respondents had worked upon SECIA in 28 individual countries while 26 respondents specified more than one country. USA was the country indicated by the highest number of respondents. Most of the companies for which the respondents worked were developing final systems, and most of the respondents were engineers, had five or more years of experience in SECIA, or had been involved in five or more projects. All the respondents reported the occurrence of some SECIA phenomenon in some project.

4.1 Circumstances under Which Safety Evidence Change Impact Analysis Is Addressed (RQ1)

RQ1 was answered by 84 out of 97 respondents (questions Q9–14 of the questionnaire; Appendix A, available in the online supplemental material).

4.1.1 Situations Frequency

The results summarized in Table 1 show that SECIA is an activity that the respondents had dealt with in several situations. *Modification of a new system during its development* is the situation with the highest median ("most projects"), most frequently indicated as happening in every project, and the least frequently indicated as never happening. Fig. 3 shows the number of situations reported by the

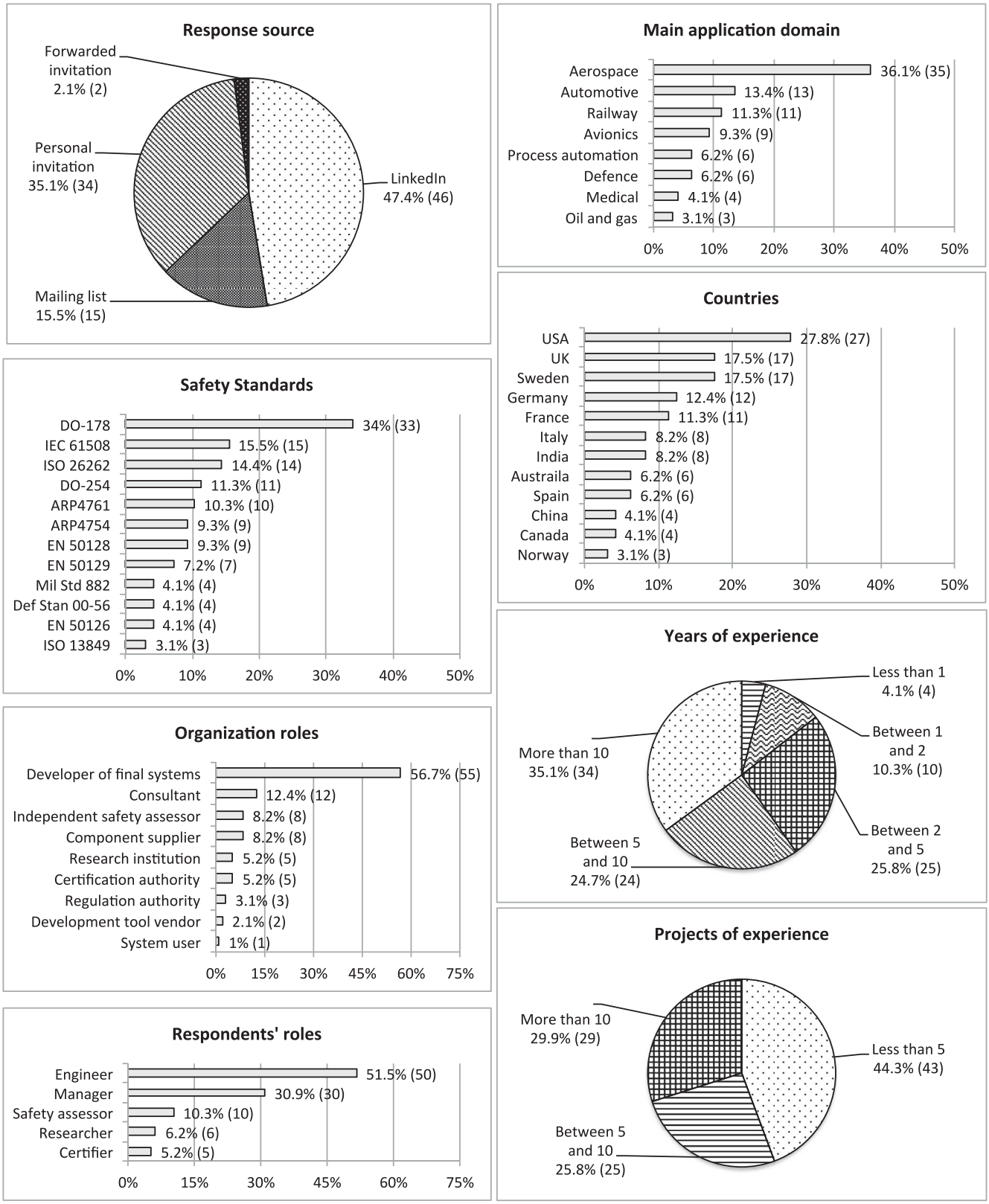

Fig. 2. Summary of respondents' demographics.

respondents. Most of the respondents reported involvement in more than six situations. Research institution, development tool vendor, and system user are the only organization roles for which no respondent reported involvement in all the situations.

The only strong correlation found for the situations for SECIA is between *Modification of a new system during its development* and *Modification of a new system as a result of its V&V* (corr. = 0.6). This relationship appears reasonable because system development and V&V are usually

TABLE 1
Frequency of Situations for SECIA

	N	Never	Few projects	Some projects	Most projects	Every project	*Median*
Modification of a new system during its development	84	7.1% (6)	13.1% (11)	28.6% (24)	**31% (26)**	20.2% (17)	*Most projects*
Modification of a new system as a result of its V&V	84	13.1% (11)	21.4% (18)	**25% (21)**	**25% (21)**	15.5% (13)	*Some projects*
Re-certification of an existing system after some modification	84	23.8% (20)	15.5% (13)	17.9% (15)	**34.5% (29)**	8.3% (7)	*Some projects*
Reuse of existing components in a new system	84	13.1% (11)	19% (16)	**33.3% (28)**	28.6% (24)	6% (5)	*Some projects*
Modification of a system during its maintenance	84	23.8% (20)	**29.8% (25)**	23.8% (20)	17.9 (15)	4.7% (4)	*Few projects*
New safety-related request from an assessor or a certification authority	84	26.2% (22)	**35.7% (30)**	25% (21)	10.7% (9)	2.4% (2)	*Few projects*
Re-certification of an existing system for a different operational context	84	**40.5% (34)**	23.8% (20)	21.4% (18)	11.9% (10)	2.4% (2)	*Few projects*
Re-certification of an existing system for a different standard	84	**50% (42)**	20.2% (17)	17.9% (15)	10.7% (9)	1.2% (1)	*Few projects/ Never*
Re-certification of an existing system for a different application domain	84	**59.5% (50)**	13.1% (11)	15.5% (13)	10.7% (9)	1.2% (1)	*Never*

regarded as intertwined [2], thus they can be intertwined for SECIA too.

Fewer respondents than expected reported that they had never dealt with SECIA for Re-certification of an existing system for a different standard and Re-certification of an existing system for a different application domain. We consider this an indication that SECIA in these situations happens more often ('non-never' answers) than most people think. Our pre-understanding is based on discussions among different practitioners and researchers. Given the difficulty in cost-effectively managing re-certification in these situations [13], research efforts targeted at the situations are necessary. They can have an important impact, and the number of publications dealing with safety assurance and certification for different standards and domains is very small [47]. The need for re-certification (and thus for SECIA) and the associated effort and cost are also among the main demotivating factors for system modifications [12]. Practitioners have also reported that system re-certification poses challenges for provision of safety evidence in general [48].

Regarding additional situations in which the respondents reported to have been involved in SECIA when asked about them (Q10), we consider it particularly interesting to study the practices after accidents (reported by one respondent) and for system of systems reuse (reported by another respondent). Our hypothesis is that SECIA after an accident might be performed more thoroughly than in other situations, as no one wants to be blamed for a second accident. Similarly, SECIA for systems of systems seems to be a situation in which existing practices might not be effective and efficient. The size and complexity of these systems very likely give rise to new challenges for SECIA, or make other challenges more difficult to address.

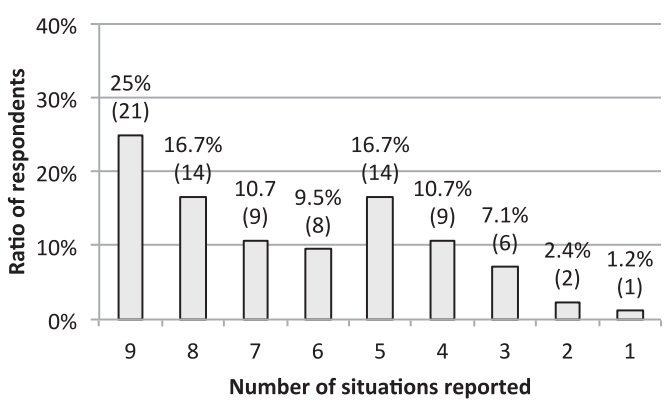

Fig. 3. Number of situations reported by the respondents.

4.1.2 Frequency of Impact Analysis from Artefact Types

Table 2 shows how often the respondents had performed SECIA as a consequence of changes in different artefact types. Column "N" indicates the number of respondents that provided an answer other than "I don't know".

The median of six out of the 14 artefact types *(Design Specifications, Requirements Specifications, Safety Analysis Results, Source Code, Test Case Specifications,* and *Traceability Specifications)* is in "most projects", and the mode for all these artefact types is in "every project". *Requirements Specifications* is the artefact type most commonly reported as triggering SECIA in every project and with the highest ratio of answers other than "never".

Overall, the results are in line with insights from related work. Requirements changes and thus subsequent impact analyses are commonly acknowledged as frequent occurrences [18]. SECIA from the artefact types was expected based on prior work (see Section 2.2), but its relative frequency was unknown for most artefact types. It was hard to judge for instance the extent to which SECIA is performed from *Safety Cases*. The results also uncover an important gap in prior research: *Safety Analysis Results* seem to trigger SECIA in most projects, but their evolutionary nature and impact analysis from them have received little attention.

TABLE 2
SECIA Frequency as a Consequence of Changes in Artefact Types

	N	Never	Few projects	Some projects	Most projects	Every project	*Median*
Requirements Specifications	78	3.8% (3)	9% (7)	25.6% (20)	23.1% (18)	**38.5% (30)**	*Most projects*
Source Code	74	13.5% (10)	16.2% (12)	16.2% (12)	20.3% (15)	**33.8% (25)**	*Most projects*
Test Case Specifications	77	9.1% (7)	16.9% (13)	22.1% (17)	20.8% (16)	**31.1% (24)**	*Most projects*
Traceability Specifications	78	10.3% (8)	21.8% (17)	12.8% (10)	24.3% (19)	**30.8% (24)**	*Most projects*
Design Specifications	76	7.9% (6)	13.1% (10)	25% (19)	23.7% (18)	**30.3% (23)**	*Most projects*
Safety Analysis Results	76	3.9% (3)	22.4% (17)	19.7% (15)	26.3% (20)	**27.7% (21)**	*Most projects*
Manual V&V Results	76	9.2% (7)	23.7% (18)	**26.3% (20)**	14.5% (11)	**26.3% (20)**	*Some projects*
Safety Cases	77	10.4% (8)	22.1% (17)	**27.2% (21)**	14.3% (11)	26% (20)	*Some projects*
Assumptions and Operation Conditions Specifications	73	11% (8)	20.5% (15)	**32.9% (24)**	16.4% (12)	19.2% (14)	*Some projects*
Tool-Supported V&V Results	76	18.4% (14)	22.4% (17)	**25%(19)**	13.2% (10)	21% (16)	*Some projects*
Architecture Specifications	71	**22.6% (16)**	21.1% (15)	18.3% (13)	19.7% (14)	18.3% (13)	*Some projects*
System Lifecycle Plans	76	23.7% (18)	**25% (19)**	18.4% (14)	15.8% (12)	17.1% (13)	*Some projects*
Reused Components Information	72	20.8% (15)	**29.2% (21)**	16.7% (12)	18% (13)	15.3% (11)	*Some/Few projects*
Personnel Competence Specifications	70	**40% (28)**	24.3% (17)	14.3% (10)	8.6% (6)	12.8% (9)	*Few projects*

When asked about further artefact types from which SECIA was performed, the individual free-text responses referred to:

a) Critical component maintenance information for security assurance
b) Project methodology and regulation authority documentation
c) Compliance plans
d) Means for verification

We argue that this additional information shows two characteristics of the current state of practice. First, there is a growing interest in the relation between safety and security. Second, changes in safety standards and how to perform SECIA according to these changes is an important concern, including changes in the way to comply with the standards.

4.1.3 Frequency of Change Impact on Artefact Types

Table 3 shows how often the artefact types had been affected by changes to the body of safety evidence. Column "N" indicates the number of respondents that provided an answer other than "I don't know". *Manual V&V results* obtained the highest median, whereas *Requirements Specifications* were reported as being affected in every project by the highest ratio of respondents.

These results, in combination with those in Table 2, indicate that *Requirements Specifications* probably have the most important role in SECIA, whereas *Personnel Competence Specifications* probably have the least important one. A possible explanation for the latter can be that personnel competence rarely changes during a system's lifecycle because of the stringent requirements from safety standards on the involved people's experience and education. Another reason could be that *Personnel Competence Specifications* barely depend on other artefact types, and vice-versa. Nonetheless, we show below that some strong correlations with *Personnel Competence Specifications* have been found.

It can be interesting to compare the differences between the use of the artefact types as safety evidence (according to Nair et al. [48]) and their role as SECIA triggers and as affected by changes. For example, *Requirements Specifications*

TABLE 3
Change Impact Frequency in Artefact Types

	N	Never	Few projects	Some projects	Most projects	Every project	*Median*
Manual V&V Results	74	4.1% (3)	18.9% (14)	25.7% (19)	24.3% (18)	**27% (20)**	*Most projects*
Test Case Specifications	77	3.9% (3)	15.6% (12)	**31.1% (24)**	27.3% (21)	22.1% (17)	*Some projects*
Source Code	74	2.7% (2)	14.9% (11)	**33.8% (25)**	21.6% (16)	27% (20)	*Some projects*
Safety Cases	73	6.9% (5)	21.9% (16)	23.3% (17)	21.9% (16)	**26% (19)**	*Some projects*
Requirements Specifications	76	5.3% (4)	18.4% (14)	**31.6% (24)**	15.8% (12)	28.9% (22)	*Some projects*
Safety Analysis Results	73	4.1% (3)	23.3% (17)	**30.1% (22)**	17.8% (13)	24.7% (18)	*Some projects*
Design Specifications	76	1.3% (1)	25% (19)	**32.9% (25)**	17.1% (13)	23.7% (18)	*Some projects*
Traceability Specifications	74	10.8% (8)	24.3% (18)	**25.7% (19)**	14.9% (11)	24.3% (18)	*Some projects*
Architecture Specifications	75	10.7% (8)	25.3% (19)	**37.3% (28)**	10.7% (8)	16% (12)	*Some projects*
Assumptions and Operation Conditions Specifications	71	14.1% (10)	**29.6% (21)**	26.7% (19)	12.7% (9)	16.9% (12)	*Some projects*
Tool-Supported V&V Res.	73	13.7% (10)	**37% (27)**	17.8% (13)	13.7% (10)	17.8% (13)	*Few projects*
System Lifecycle Plans	75	22.7% (17)	**29.3% (22)**	22.7% (17)	10.7% (8)	14.6% (11)	*Few projects*
Reused Components Info.	70	21.4% (15)	**31.4% (22)**	25.7% (18)	11.5% (8)	10% (7)	*Few projects*
Personnel Competence Spec.	68	**39.7% (27)**	30.9% (21)	16.2% (11)	7.3% (5)	5.9% (4)	*Few projects*

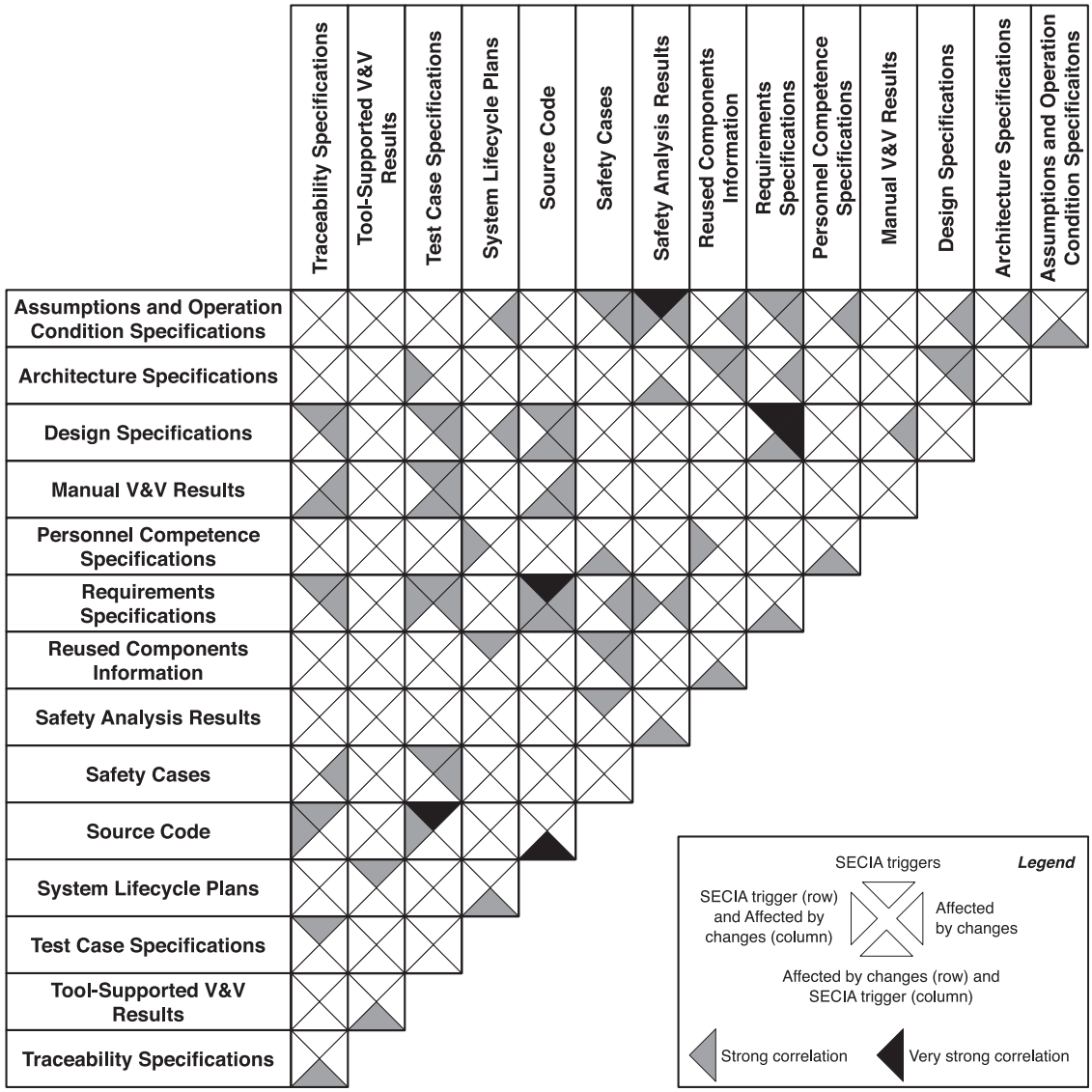

Fig. 4. Strong correlations (shaded triangle) and very strong correlations (black triangles) between artefact types, for the roles indicated by the triangles according to their position.

was reported to be used as evidence by 87 percent of the participants in [48]. Among the respondents that provided information about RQ1, 89 percent reported SECIA from the artefact type and 85 percent reported change impact. All these figures are very close, which can be interpreted as an indicator of the changing nature of requirements. The same applies to *Design Specifications*, *Test Case Specifications*, and *Traceability Specifications*.

When asked about further artefact types affected by changes, the respondents referred again to security information and to compliance plans (one respondent each).

No strong or very strong correlations have been identified between the situations in Table 1 and the artefact types in Tables 2 and 3. Therefore, we cannot claim that the frequency of SECIA from certain artefact types, or of change impact on them, greatly depends on the situation in which a SECIA is performed.

4.1.4 Correlations Between Artefact Types

Fig. 4 shows all the strong and very strong correlations identified between the artefact types as SECIA triggers (results in Table 2) and as types affected by changes (results in Table 3). The values of these correlations are provided in Appendix C, available in the online supplemental material. Their p-values are below 1e-08.

For example, *Requirements Specifications* and *Traceability Specifications* are strongly correlated both as SECIA triggers (in Fig. 4) and as affected by changes (in Fig. 4), and *Architecture Specifications* as SECIA trigger are strongly correlated to *Test Case Specifications* as affected by changes (in Fig. 4). Our interpretation is as follows: if *Requirements Specifications* trigger SECIA, so likely do *Traceability Specifications*; also, if *Requirements Specifications* are affected by changes, so likely are *Traceability Specifications*; finally, if *Architecture Specifications* trigger SECIA, *Test Case Specifications* are likely affected by changes.

Strong and very strong correlations have also been identified between the results in Tables 2 and 3 for a given artefact type (in the diagonal of Fig. 4). We refer to these correlations as correlations of an artefact type with itself, and interpret them as either (1) correlations between individual components of a given artefact type (e.g., requirements in

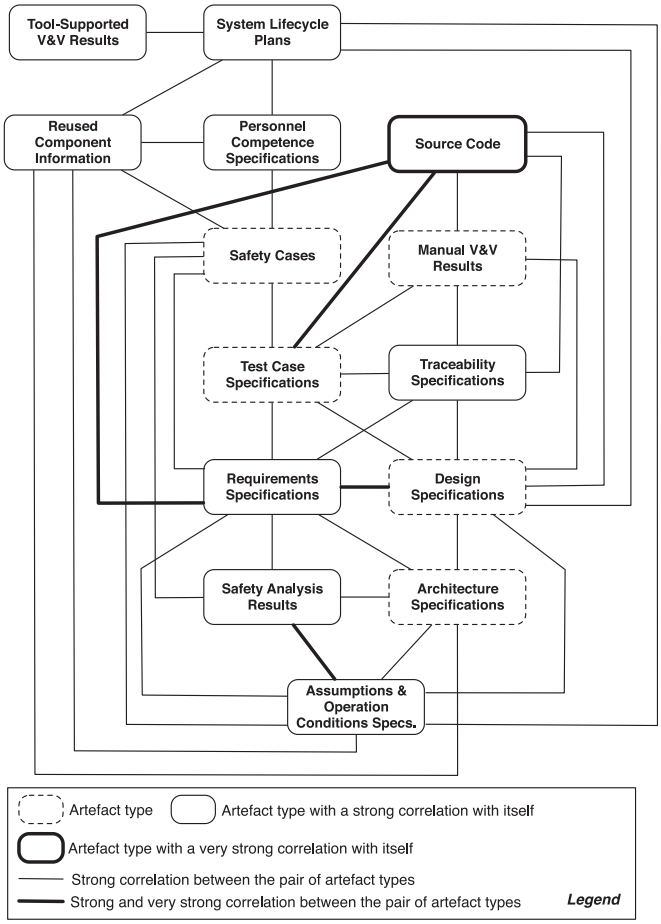

Fig. 5. Artefact types correlations graph.

Requirements Specifications, classes in *Source Code*, or elements of *System Lifecycle Plans*), or (2) correlations between instances of a given artefact type (e.g., individual functional specifications for *Requirements Specifications*, different *Source Code* files, or various verification reports for *Manual V&V Results*). For the correlations of an artefact type with itself, only one correlation is shown in Fig. 4 (i.e., ⧓ instead of ⧓) because the correlation from Table 2 to Table 3 is the same as from Table 3 to Table 2.

Fig. 5 synthesises all the correlations between artefact types by means of a graph. The figure shows which pairs of artefact types have some strong correlation (only strong correlations) and which have both strong and very strong correlation. Artefact types with a strong or very strong correlation with themselves are also indicated.

We interpret these correlations as the evidence of the joint involvement of the artefact types in SECIA. More importantly, these correlations indicate relationships whose documentation and maintenance is arguably of utmost importance. The relationships show the artefact types that will likely be involved in SECIA when other types are. This kind of information is not provided in detail in safety standards but can help practitioners know the artefact types to consider in a SECIA. Standards typically only state that system suppliers need to analyse impact as a result of system or software changes and maintenance, and to determine reassessment needs after a change.

We identified 17 strong correlations and four very strong correlations between the artefact types as SECIA triggers. A possible explanation for this is that SECIA is performed for pairs of artefact types (e.g., *Requirements Specifications* and *Design Specifications*) in a same SECIA effort or in related activities.

There is a very strong correlation between *Requirements Specifications* and *Source Code* as SECIA triggers (corr. = 0.82). A possible explanation for this correlation might be that requirements change after source code has already been implemented. Such a change could happen for example at late system development stages or when a new version of a system is developed. The very strong correlation between *Assumptions and Operation Conditions Specifications* and *Safety Analysis Results* (corr. = 0.75) confirms the importance of the former artefact type for creating the latter. The same applies to the very strong correlation between *Requirements Specifications* and *Design Specifications* (corr. = 0.78). The very strong correlation between *Source Code* and *Test Case Specifications* (corr. = 0.75) also seems logical to us. It is noteworthy that no strong correlation as SECIA triggers has been found between some pairs of artefact types commonly studied together in the literature, e.g., *Requirements Specifications* and *Architecture Specifications*.

We found 27 strong correlations between the artefact types that were reported as affected by changes. Again, *Requirements Specifications* and *Design Specifications* are very strongly correlated (corr. = 0.78), and it is the only very strong correlation between artefact types as affected by changes.

We detected 25 strong correlations and one very strong correlation regarding artefact types as SECIA triggers and as affected by changes. These correlations indicate the existence of many important relationships between the artefact types for impact analysis sequences. We interpret the very strong correlation of *Source Code* with itself (corr. = 0.76) as a clear indicator of ripple effects on safety-critical source code.

Nine pairs of artefact types in Fig. 4 have three or four correlations:

1) *Requirements Specifications* and *Source Code*
2) *Requirements Specifications* and *Design Specifications*
3) *Requirements Specifications* and *Test Case Specifications*
4) *Test Case Specifications* and *Source Code*
5) *Design Specifications* and *Source Code*
6) *Traceability Specifications* and *Source Code*
7) *Test Case Specifications* and *Manual V&V Results*
8) *Design Specifications* and *Test Case Specifications*
9) *Safety Analysis Results* and *Assumptions and Operation Conditions Specifications*

These pairs can be regarded as the most relevant for SECIA in practice. It is important to note that prior research has studied most of them. Nonetheless, publications on SECIA and traceability related to *Manual V&V Results* are scarce [47], and some pairs (5–9) have been considerably less studied than others (1–4).

Approaches for analysing previous impact analysis reports such as the one presented by Borg et al. [5] can facilitate further analyses of all these relationships and provide new insights. This approach supports the analysis of how changes in a given artefact type affect other artefact types, and of the extent to which the changes propagate.

One result subject to interpretation is change impact on *Safety Cases*. A safety case corresponds to a collection of

TABLE 4
Level of Automation Offered by Tools for SECIA from Each Artefact Type

	N	Fully Manual	Decision Support Available	Semi-Automated Recomm.	Highly-Automated Recomm.	Automatic Impact Analysis	*Median*
Source Code	73	**31.5% (23)**	16.4% (12)	**31.5% (23)**	17.8% (13)	2.8% (2)	*Semi-Automated Recommendations*
Traceability Specifications	79	25.3% (20)	26.6% (21)	**27.8% (22)**	15.2% (12)	5.1% (4)	*Decision Support Available*
Architecture Specifications	72	34.7% (25)	**41.7% (30)**	19.4% (14)	1.4% (1)	2.8% (2)	*Decision Support Available*
Tool-Supported V&V Results	79	**32.9% (26)**	21.5% (17)	24.1% (19)	17.7% (14)	3.8% (3)	*Decision Support Available*
Test Case Specifications	79	**39.2% (31)**	29.1% (23)	20.3% (16)	8.9% (7)	2.5% (2)	*Decision Support Available*
Requirements Specifications	80	**40% (32)**	33.8% (27)	16.2% (13)	8.7% (7)	1.3% (1)	*Decision Support Available*
Safety Analysis Results	76	**40.8% (31)**	23.7% (18)	23.7% (18)	10.5% (8)	1.3% (1)	*Decision Support Available*
Design Specifications	76	**42.1% (32)**	35.5% (27)	17.1% (13)	4% (3)	1.3% (1)	*Decision Support Available*
Safety Cases	73	**56.1% (41)**	27.4% (20)	13.7% (10)	1.4% (1)	1.4% (1)	*Fully Manual*
Manual V&V Results	78	**56.4% (44)**	23.1% (18)	16.7% (13)	3.8% (3)	0% (0)	*Fully Manual*
Reused Components Info.	71	**59.2% (42)**	31% (22)	7% (5)	1.4% (1)	1.4% (1)	*Fully Manual*
Personnel Competence Specs.	66	**63.6% (42)**	28.8% (19)	7.6% (5)	0% (0)	0% (0)	*Fully Manual*
System Lifecycle Plans	75	**65.4% (49)**	21.3% (16)	9.3% (7)	4% (3)	0% (0)	*Fully Manual*
Assumptions and Operation Conditions Specifications	72	**68.1% (49)**	20.8% (15)	9.7% (7)	1.4% (1)	0% (0)	*Fully Manual*

references to other artefacts to justify system safety and compliance. Therefore, changes in the referred artefacts affect safety and compliance justification. Looking at the medians, change impact on *Safety Cases* ("some projects", Table 3) seems to be less often than SECIA triggered from other six artefact types ("most projects"; Table 2). In addition, *Safety Cases* have no strong or very strong correlation as affected by changes when other artefact types trigger SECIA (Fig. 4). These results suggest that potential change impacts on *Safety Cases*, as a result of SECIA from referred artefacts, do not lead to actual changes to this artefact type.

We find three possible explanations. First, changes in other artefact types might be made before a safety case refers to them. Second, although some artefacts are referred to in a safety case, their changes might not impact the safety case. Third, industry might not be adequately addressing how changes in the body of safety evidence impact a safety case. Indeed, the insights that we provide can raise some concerns on how safety case evolution is managed.

It is recognised as a good practice to incrementally and iteratively create safety cases [32], as instances of other artefact types are created and maintained. In fact, this approach is recommended in some safety standards, explicitly (e.g., Defence Standard 00–56) or implicitly (e.g., EN 50129). Consequently, it would be logical that the median for *Safety Cases* in Table 3 was higher than "some projects", and that *Safety Cases* had strong or very strong correlations with more artefact types (e.g., *Architecture Specifications*, in line with [3]). Albeit "every project" was the mode for *Safety Cases* in Table 3, 28.8 percent of the respondents indicated that *Safety Cases* had never been affected by changes or had been affected in few projects.

Given the importance of safety cases, we argue that how they are affected by changes in other artefact types and how their evolution is managed are two areas that require further research. Safety case creation during late system development phases can lead to deficiencies such as confirmation bias and thus decrease their credibility. Many experts have discussed the adequacy of and the need for safety case regimes for safety-critical systems based on their own insights or single case studies (e.g., [37]). Our study appears to be the first that empirically shows that many practitioners might not be adequately managing safety case evolution.

As an overall conclusion, we argue that further research efforts on impact analysis for safety-critical systems are necessary. More specifically, we believe that further investigations on impact analysis regarding safety-targeted artefact types are essential, especially for those with over half a dozen strong or very strong correlations (*Safety Analysis Results*, *Assumptions and Operation Conditions Specifications*, *Manual V&V Results*, and *Safety Cases)*. Their adequate change management is essential for ensuring safety, especially for software-intensive systems. It has been acknowledged that many practitioners fail to understand and identify software safety risks (e.g., [37]), including software change impact on system safety.

Finally, a question remains why only one strong correlation has been found between *Tool-Supported V&V Results* and the rest of artefact types. Someone could expect a higher number of strong or very strong correlations, higher than for *Manual V&V Results*, and a higher correlation with *Test Cases Specifications* or *Source Code*. Changes in these artefacts might impact, for instance, existing testing results. The results suggest that *Test Cases Specifications* and *Source Code* most commonly change before *Tool-Supported V&V Results* are available.

4.2 Tool Support for Safety Evidence Change Impact Analysis (RQ2)

Eighty-four out of 97 respondents provided information for answering RQ2 (questions Q15–19 of the questionnaire; Appendix A, available in the online supplemental material).

4.2.1 Level of Automation in Safety Evidence Change Impact Analysis

Table 4 shows the level of automation for the tool support used for SECIA when the artefact types change. Column

"N" indicates the number of respondents that provided an answer other than "I don't know". The levels of SECIA automation were defined based on a previous study on human interaction with automation [52]:

- *Fully manual:* no automation in the process; e.g., impact determined by reading documentation and asking colleagues.
- *Decision support available:* limited support for narrowing down a selection of possible impact; e.g., search tool used to seek impact, repositories easy to browse thanks to information structure.
- *Semi-automated recommendations*: tools suggest artefacts that might be impacted but humans must confirm.
- *Highly automated recommendations*: tools report impact and humans have the authority to veto the suggestions.
- *Automatic impact analysis*: tools determine the impact without human involvement.

Except for the *Source Code* artefact type, the median for all artefact types is "decision support available" or "fully manual". In addition, the mode is different from "fully manual" only for three artefact types. These responses imply that SECIA is most often performed manually. Such manual work is not only time consuming and error prone, but might also lead to mistakes when detecting safety and compliance risks. Overall, the high number of "fully manual" responses is surprising because many basic tools with some search functionality such as Excel and Word (reported as used for SECIA; see Section 4.2.2) are probably available at the respondent's organizations. It remains for further investigation whether the awareness of these tools was low or the functionality that they provide is insufficient for SECIA.

SECIA from *Requirements Specifications* was reported as "fully manual" by 40 percent of the respondents, despite the existence of many requirements management tools that provide some automated support [9], and the relatively higher level of automation reported for *Traceability Specifications*. It is unexpected to us that 31.5 percent of the respondents indicated that SECIA from *Source Code* was "fully manual", since *Source Code* is typically created in development environments (i.e., software tools) that offer various automation features.

The level of automation for *Assumptions and Operation Conditions Specifications* could raise some concerns. It is not only the artefact type for which SECIA has been most frequently reported as "fully manual", but also a type whose inadequate change management led to e.g., the well-known accident of Ariane 5 in 1996 [37] and other more recent accidents or near-accidents in the space domain and in transportation systems [23]. Moreover, most of the respondents had dealt with SECIA from *Assumptions and Operation Conditions Specifications* (Table 2) and with *Re-certification of an existing system for a different operational context* (Table 1). To some extent, this result suggests that prevention measures to avoid prior accidents could be improved. Furthermore, *Assumptions and Operation Conditions Specifications* are essential for any safety-critical system, as they can only be deemed safe for a given operational context [32].

Regarding the relationships between the levels of automation for various artefact types, we found strong correlations ($p < 1e-08$) between the levels for *Design Specifications* and *Traceability Specifications* (corr. $= 0.62$), *Traceability Specifications* and *Tool-Supported V&V Results* (corr. $= 0.61$), and *Source Code* and *Safety Cases* (corr. $= 0.67$). These relationships make us suspect that there is tool support that can automate SECIA from both artefact types of these pairs, e.g., DOORS or some internal tool according to the results in Section 4.2.2. This hypothesis should be further investigated.

We have not found any strong or very strong correlations between the level of automation and the SECIA situations and artefact types (RQ1). This lack of correlations suggests that the level of automation does not vary much among the industrial contexts studied for RQ1.

Individual free-text responses emphasised that:

a) The level of automation is increasing
b) SECIA qualified tools are important and necessary
c) Although tools are used, change impact is always assessed manually
d) More advanced tools, whose results can be used as evidence, are necessary for evidence review

This additional information indicates that some practitioners regard the level of automation as increasing, but they still expect improvements – especially on tool qualification. The use of qualified tools would imply that manual assessment would not be necessary in (c), and that certification authorities would accept (d).

The results outlined in Table 4 confirm several observations from previous work. Research often focuses on impact analysis for *Source Code* [36], which has led to a considerable number of automated source code impact analysis proposals and thus more possible automated solutions for practice. Traceability information plays a major role for impact analysis and is typically maintained with some supporting tool [9], [55]. Therefore, we expected that most respondents reported some SECIA automation for *Traceability Specifications*. Previous work suggests that the overall level of SECIA automation is low (e.g., [21], [57], [64]), and also that automatic change impact seems to be used for some artefact types [48].

Our study refines these general insights by providing evidence of the level of automation for each artefact type, gathered from a much larger sample of practitioners. The level of automation seems to vary among the artefact types, and the existence of many commercial tools for managing certain artefact types (e.g., *Requirements Specifications*) does not result in a much higher level of SECIA automation. Finally, there seems to be a research gap in impact analysis automation for *Assumptions and Operation Conditions Specifications*, *Manual V&V Results*, *Personnel Competence Specifications*, *Reused Components Information*, *Safety Cases*, and *System Lifecycle Plans*.

4.2.2 Tools for Safety Evidence Change Impact Analysis

The respondents listed 98 different tools for SECIA purposes. *Traceability Specifications* and *Requirements Specifications* are the artefact types for which the highest ratio of respondents indicated some tool support (Fig. 6), whereas the highest variation of tools was found for *Source Code* (Fig. 7). Internal tools were reported as used for SECIA from all the artefact

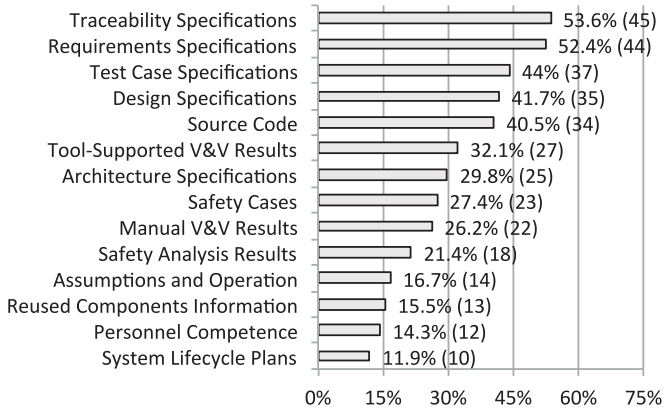

Fig. 6. Respondents that indicated some tool for each artefact type.

types and tailored extensions of commercial tools for most artefact types, see Fig. 8. Only two commercial tools (VeroTrace and DOORS) were mentioned for more than half of the artefact types. However, only one respondent reported the use of VeroTrace. Some model-based tools (e.g., Artisan Studio, ASCE, and Rhapsody) were reported for 10 out of the 14 artefact types (71.4 percent).

A summary of the tools for storing SECIA evidence and for SECIA from each artefact type is shown in Figs. 9 and 10, respectively. These figures list the tools reported by at least two respondents. Appendix D, available in the online supplemental material, provides a description of these tools. Information about the rest of tools reported by the respondents can be found in [11]. When reporting the use of compilers for SECIA, we understand that the respondents referred to software development environments.

Tools that are not specifically targeted at SECIA or systems engineering such as Excel and Word seem to be commonly used. Internal tools are the most frequently used for nine out of the 14 artefact types, and DOORS had the highest number of respondents reporting its use for a given artefact type (*Requirements Specifications* and *Traceability Specifications*). Thirty-seven different tools were mentioned for storing SECIA evidence, with internal tools mentioned most frequently.

Reviewing the results in the context of related work, our survey identified an extensive number of tools involved in SECIA. Prior publications [36], [47] have paid little attention

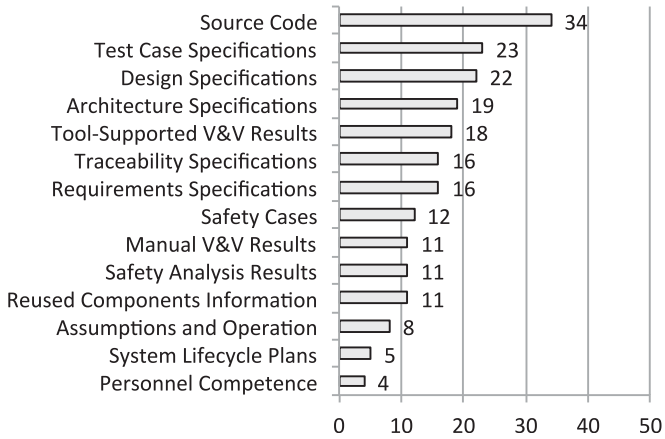

Fig. 7. Number of tools indicated for each artefact type.

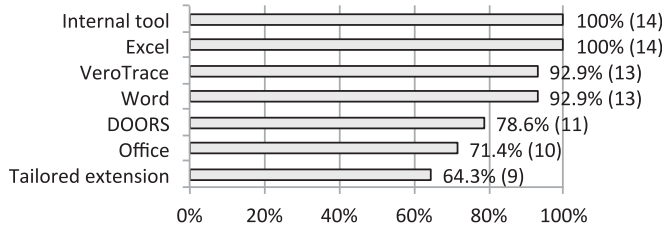

Fig. 8. Tools for SECIA reported for more than half of the artefact types and ratio of artefact types for which the tools were reported.

to the use of basic tools (e.g., Excel and Word) for impact analysis purposes, and some tools usually mentioned in the literature (e.g., JRipples) do not seem to have been adopted for safety-critical systems. Our survey provides evidence of how often tools are applied in industry for SECIA. The evidence includes practitioners' use of internal tools, the extension of commercial tools, the adoption of general-purpose tools, and the use of models. Based on our results and past surveys [48], [50], [59], DOORS appears to be one of the main tools for safety-critical system engineering and assurance.

The results help bridge the gap between the insights, assumptions, and claims presented in the literature and the state of the practice. In this sense, we argue that researchers should be careful when making statements regarding the suitability of impact analysis tools, or the need for them. Although more tool support is probably necessary, we wonder if, for instance, research on new source code impact analysis tools should be prioritized, given the high number of commercially available tools. The remaining question is what kind of new SECIA tool support is necessary, especially taking into account the challenges perceived by the respondents (Section 4.3), and thus what specific tool aspects should be investigated. Tools that integrate and use safety evidence meta-information from different sources for SECIA seem to be highly desirable [48].

4.3 Challenges in Safety Evidence Change Impact Analysis (RQ3)

Ninety out of the 97 respondents provided information for answering RQ3 (questions Q20–22 of the questionnaire; Appendix A, available in the online supplemental material).

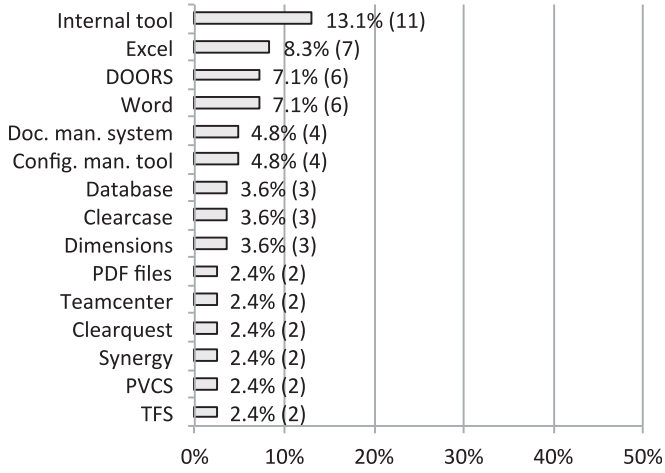

Fig. 9. Summary of the tools for storing SECIA evidence and respondents that reported them.

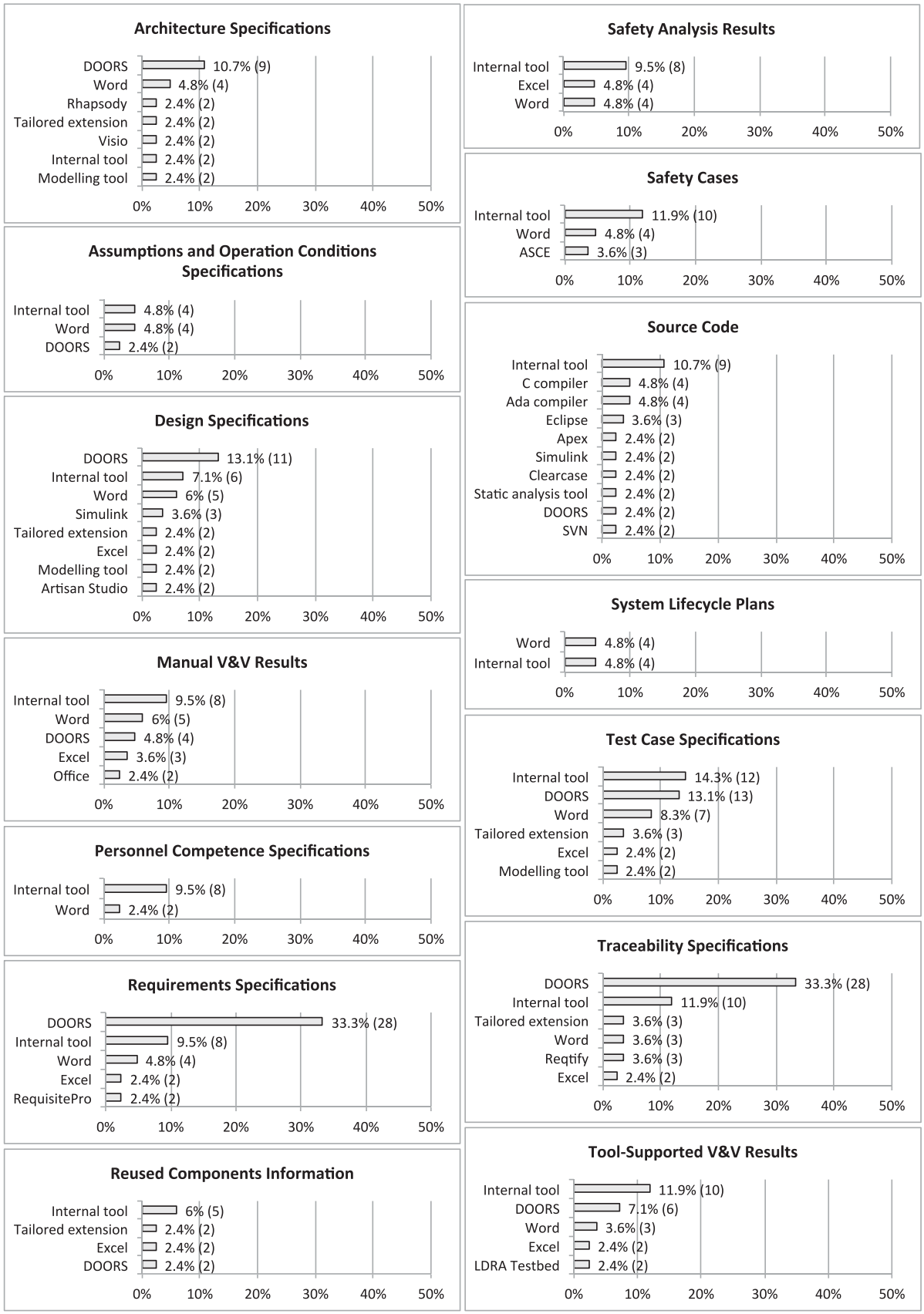

Fig. 10. Summary of the tools for SECIA per artefact type and respondents that reported them.

TABLE 5
SECIA Challenges Frequency

	N	Never	Few projects	Some projects	Most projects	Every project	*Median*
Insufficient tool support	90	11.1% (10)	21.1% (19)	17.8% (16)	**34.4% (31)**	15.6% (14)	*Most/ Some projects*
Difficulty in estimating the effort required to manage a change	90	4.4% (4)	17.8% (16)	31.1% (28)	**36.7% (33)**	10% (9)	*Some projects*
Vast number of artefacts to trace	90	7.8% (7)	15.6% (14)	**35.6% (32)**	25.6% (23)	15.6% (14)	*Some projects*
Too coarse granularity of the traceability between artefacts to accurately know the consequences of a change	90	10% (9)	25.5% (23)	26.7% (24)	28.9% (26)	8.9% (8)	*Some projects*
Insufficient traceability between artefacts to accurately know the consequences of a change	90	5.6% (5)	25.6% (23)	**32.2% (29)**	**28.9% (26)**	7.8% (7)	*Some projects*
Difficulty in determining the effect of a change on system safety	90	4.4% (4)	23.3% (21)	**38.9% (35)**	24.4% (22)	8.9% (8)	*Some projects*
Long time for evaluating the consequences of a change	90	5.6% (5)	27.8% (25)	**34.4% (31)**	25.6% (23)	6.7% (6)	*Some projects*
Difficulty in assessing system-level impact of component reuse	90	10% (9)	28.9% (26)	**38.9% (35)**	18.9% (17)	3.3% (3)	*Some projects*
Unclear meaning of the traceability between artefacts in order to know how to manage a change	90	15.6% (14)	26.7% (24)	**35.6% (32)**	17.8% (16)	4.4% (4)	*Some projects*
Insufficient confidence by assessor or certifiers in having managed a change properly	90	20% (18)	27.8% (25)	**32.2% (29)**	16.7% (15)	3.3% (3)	*Some projects*
Lack of a systematic process for performing impact analysis	90	12.2% (11)	**28.9% (26)**	27.8% (25)	20% (18)	11.1% (10)	*Some projects*
Difficulty in deciding if a component can be reused	90	21.1% (19)	30% (27)	**31.1% (28)**	14.4% (13)	3.3% (3)	*Few projects*
Excessive detail of the traceability between artefacts, making traceability management more complex than necessary for impact analysis purposes	90	25.6% (23)	**32.2% (29)**	26.7% (24)	11.1% (10)	4.4% (4)	*Few projects*

4.3.1 Challenges Frequency

Table 5 provides the frequencies of the experienced SECIA challenges. No challenge has "never" or "every project" as the mode. While almost 100 tools for SECIA were reported (Section 4.2.2), the challenge with the highest median is *Insufficient tool support*.

When analysing the relationship between pairs of challenges and between the challenges and other phenomena, we only identified a strong correlation between the challenges *Difficulty in assessing system-level impact of component reuse* and *Difficulty in deciding if a component can be reused* (corr. $= 0.61$; $p < 1e-08$). This correlation could be expected because both challenges refer to component reuse. The lack of relevant correlations with SECIA circumstances (RQ1) suggests that the challenges do not occur more often in certain situations or when dealing with certain artefact types.

We find it interesting that the correlation between the respondents' experience (see Fig. 2) and the frequency of the challenges is very weak, weak, or very close to weak ($-0.32 <= \text{corr.} <= -0.1$; average corr. $= -0.19$). This suggests that the challenges are visible as soon as someone gets involved in SECIA and that experience rarely mitigates them. This finding is of particular importance for experiments on SECIA in which students are subjects, especially on SECIA challenges. It should not be claimed that the results of the experiments would necessarily differ with experienced practitioners.

We cannot claim that a higher level of SECIA automation will decrease the frequency of the challenges because we have not found any strong or very strong correlation in that matter. This may seem counterintuitive for the *Insufficient tool support* challenge, which has weak or very weak correlations with the level of SECIA automation for each artefact type ($-0.27 <= \text{corr.} <= 0.02$; average corr. $= -0.15$). We consider that better tool support is beyond simple automation such as the identification of potential impacts. Tools can further guide users when performing SECIA, indicating e.g., how to manage the impact.

We received the following information in the individual free-text answers about further challenges:

a) Difficulty in understanding and following safety standards' indications
b) Difficulty in assessing safety impact using the available trace data
c) Lack of detail in existing data justification (e.g., trace link but no explanation), making it difficult to know if change compromises the justification
d) Insufficient attention to traceability
e) Finding the right balance between fine and coarse traceability

f) Traceability between all the different environments and tools used in system lifecycle
g) Difficulty in tracing the origin and real date of a change cause
h) Inefficient SECIA and variability in how to address it depending on the artefacts involved
i) Lack of an efficient regression verification strategy
j) Feature creep (scope creep), lack of support or documentation for programming language, undocumented source code, and ageing legacy systems
k) Lack of automated support for effort estimation
l) Involvement of many different people and organisations (sub-suppliers, customers, system engineers, safety engineers, lawyers...), with different interests
m) Change assessors' lack of knowledge to adequately assess an impact or subsequent impact(s)
n) Management's lack of knowledge about risks
o) Pressure to meet project time scale, and insufficient staff with the right level of competence

Although most of these challenges are acknowledged in the literature (e.g., [2], [13], [42], [46]), the respondents provided new details, e.g., in (c), (h), (l), and (m). Many answers also specialise some of the fixed challenges provided in Q20. For example, "Difficulty in assessing safety impact using the available trace data" can be regarded as a specialisation of *Difficulty in determining the effect of a change on system safety*. In our opinion, the combined set of challenges (those in the questionnaire and those mentioned by the respondents) shows that SECIA is very demanding and it should be carefully planned, performed, and verified. Indeed, Q21 was the free-text question for indicating additional information with the highest number of answers, considerably higher than for the others. The social challenges of SECIA (e.g., (l)) might be more difficult to address because they could require a change of attitude towards how to perform SECIA, not only the use of e.g., some new tool or guidance.

When comparing the results from Q20 and Q21 with the challenges reported by Nair et al. [48], we find that issues related to confidence in evidence, application of safety standards, evidence structuring and traceability, and component reuse are common to provision of safety evidence (i.e., the phenomenon under study in [48]) and SECIA (the focus of this study). Therefore, it seems that these aspects can affect the whole assurance process of a safety-critical system.

4.3.2 How to Improve Safety Evidence Change Impact Analysis

Seventy-six respondents provided information about how SECIA could be improved. We identified three main improvement areas: *Information aspects*, *Process aspects*, and *Tool aspects*. These areas were identified through the coding procedure described in Section 3.5, and are decomposed into the following sub-areas (answer codes).

Information aspects referred to the need for more information related to or for SECIA execution:

- *Communication*: information exchange among those involved in SECIA activities
- *Data used for analysis*: pieces of data that are consulted when deciding upon how to perform SECIA activities
- *Guidance*: information available for guiding SECIA
- *Knowledge*: existing information about how to deal with SECIA
- *Safety cases*: documented system safety justification
- *SECIA process transparency*: degree of knowledge about SECIA activities for those not directly involved in the activities
- *Standards*: industrially-accepted best practices followed for ensuring system safety
- *System specifications*: artefacts describing system structure, behaviour, or constraints
- *Traceability*: relationship between two artefacts
- *Training*: knowledge acquisition for those involved in SECIA

Process aspects referred to the need for better SECIA execution processes:

- *Analysis of impact on safety*: the effect that the changes in the body of safety evidence can have on system safety
- *Analysis process*: the process followed for SECIA
- *Coordination*: the degree to which those involved in SECIA activities work cooperatively
- *Credibility*: the degree to which someone would agree that SECIA activities have been adequately performed
- *Independence*: the degree of difference between those involved in SECIA activities
- *SECIA verification*: the activities targeted at guaranteeing that impact analysis and change management have been adequately addressed
- *System development*: the activities targeted at specifying and creating a system
- *System V&V*: the activities targeted at providing an assurance of certain system properties
- *Time aspects*: time resources necessary for and time constraints on SECIA

Tool aspects referred to the need for new or better SECIA tools:

- *Level of automation*: the degree of automatic support offered by tool support
- *Tool integration*: the degree of information exchange between the tools used in a system's lifecycle, including SECIA-related tools
- *Tool support*: available tools for performing SECIA

Based on the results from answer coding, we have created the taxonomy of SECIA improvement areas shown in Fig. 11 using a bottom-up process. We defined the three main improvement areas from the generalization and grouping of the codes (sub-areas) assigned to the answers during data analysis. Each answer had one or more codes. We also identified specialisation relationships between some pairs of codes (e.g., *Tool support* and *Level of automation*). Further relationships among the improvement areas can be established, e.g., information can be necessary for and tools can be used as means for process execution. As explained in Section 3.5, the coding was initiated by the first author and later iterated with the second author until the final codes were defined. Finally, the third and the fourth authors reviewed the taxonomy.

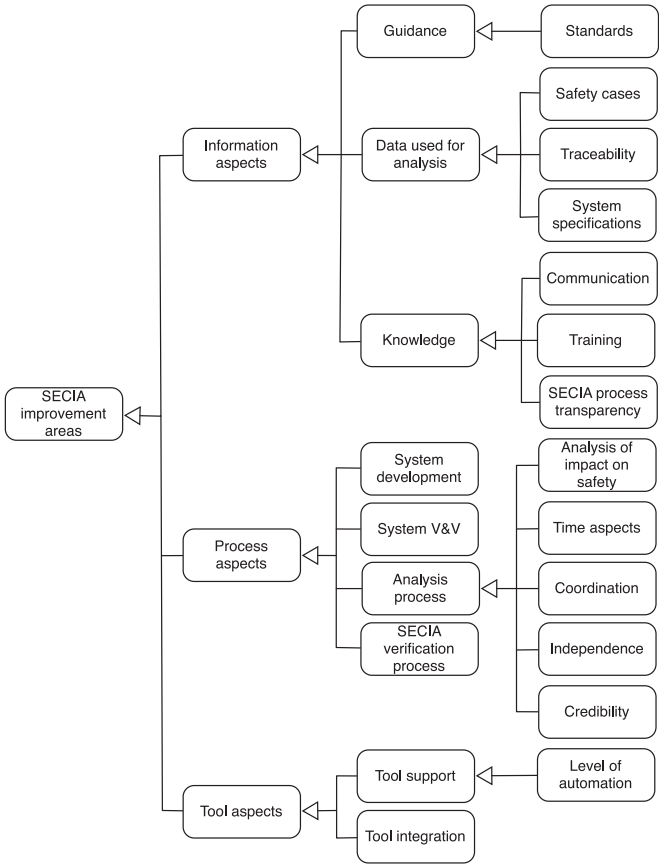

Fig. 11. Taxonomy of SECIA improvement areas.

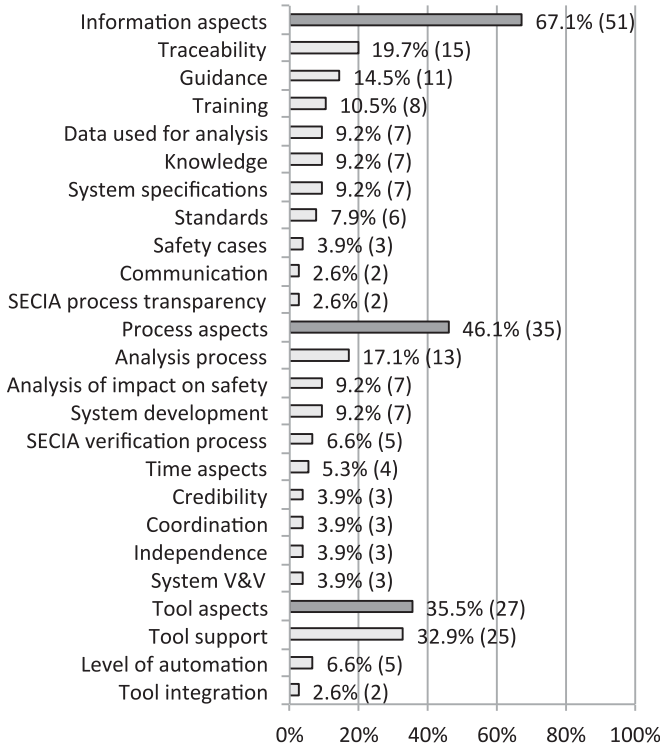

Fig. 12. Frequency of SECIA improvement areas.

Fig. 12 shows the percentage and the number (in brackets) of respondents that indicated each improvement area, from the 76 that answered this question. Over two thirds of the respondents mentioned *Information aspects*. The most frequent specific area was *Tool support*, followed by *Traceability*. Specific topics mentioned by individual respondents that we consider especially relevant are:

a) Impact analysis activities need to be more systematic
b) Better understandings of change effect semantics
c) Better safety engineering principles and methods
d) Wider knowledge about safety goals in development teams
e) Impact simulation, including simulation of system behaviour after a change
f) Higher quality of safety evidence, in particular better documentation of the scope, assumptions, and estimated impact of potential future changes
g) Use of modular safety cases
h) Standards' requirements clarification
i) Existing tools are either (1) too expensive and complex or (2) not very suitable and useful

We also analysed which improvement areas could contribute to mitigating each challenge listed in Table 5. The outcome is shown in Table 6. *Analysis process* and *Guidance* are the areas related to the highest number of challenges (10 out of 13). Therefore, it could be argued that they are the improvement areas that can have a wider impact on the mitigation of SECIA challenges.

Although references to the improvement areas can be found in safety standards and related work, the non-purely-technical areas have not received much attention in the past. For example, how to make SECIA more credible to assessors and how to overcome communication and coordination problems remain greatly unexplored. From the answers to RQ1 and RQ2 and from the insights provided in related work, we suggest the following future focus areas:

- Development of traceability guidelines, including heuristics for deciding upon suitable granularity and trace semantics for SECIA
- Identification of the artefact types for which traceability-related SECIA challenges are most often faced
- Establishment of the degree to which SECIA can, or should, be automated as a way to tackle tool support-related challenges, while conforming to certification authorities' expectations and tool qualification requirements.
- Safety cases in SECIA, for analysing change impact both from and on safety cases

4.4 Summary of Results and of Implications

SECIA seems to be most frequently addressed (RQ1) as a consequence of the *Modification of a new system during its development*, and triggered by changes in *Design Specifications*, *Requirements Specifications*, *Safety Analysis Results*, *Source Code*, *Test Case Specifications*, and *Traceability Specifications*. The artefact type most frequently affected by changes is *Manual V&V results*. In general, *Requirements specifications* can be regarded as the most central artefact type for SECIA.

Some less frequent phenomena require more research effort, including practices for which very few systematic

TABLE 6
Improvement Areas That Could Mitigate SECIA Challenges

Challenge	Improvement Area
Insufficient tool support	Tool aspects
Difficulty in estimating the effort required to manage a change	Guidance, Data used for analysis, Communication, Training, Analysis process
Vast number of artefacts to trace	Guidance, Data used for analysis, Tool aspects
Too coarse granularity of the traceability between artefacts to accurately know the consequences of a change	Guidance, Data used for analysis, Analysis process, Tool aspects
Insufficient traceability between artefacts to accurately know the consequences of a change	Guidance, Data used for analysis, Analysis process, Tool aspects
Difficulty in determining the effect of a change on system safety	Guidance, Training, Analysis of impact on safety
Long time for evaluating the consequences of a change	Data used for analysis, Knowledge, Analysis process
Difficulty in assessing system-level impact of component reuse	Guidance, Training, System development, Analysis process, Tool support
Unclear meaning of the traceability between artefacts in order to know how to manage a change	Guidance, Data used for analysis, Analysis process, Tool aspects
Insufficient confidence by assessor or certifiers in having managed a change properly	Communication, SECIA process transparency, System V&V, Analysis process, SECIA verification process, Tool aspects
Lack of a systematic process for performing impact analysis	Guidance, Analysis process, Tool support
Difficulty in deciding if a component can be reused	Guidance, Training, System development, Analysis process, Tool support
Excessive detail of the traceability between artefacts, making traceability management more complex than necessary for impact analysis purposes	Guidance, Data used for analysis, Analysis process, Tool aspects

means for SECIA exist (e.g., re-certification for a different application domain) and practices critical for ensuring system safety and demonstrating compliance with safety standards (e.g., changes in and verification of assumptions and operation conditions). Among the possible concerns about the current SECIA practices, industry should analyse whether safety case evolution is being properly addressed. Practitioners must be aware that SECIA can be necessary in a wide variety of situations and involve any artefact type. Security aspects are already considered when performing SECIA.

Sixty-nine strong SECIA-related correlations exist between the artefact types, and all the artefact types have some strong correlation. Six very strong correlations were found between *Requirements Specifications* and *Design Specifications* (two very strong correlations), *Requirements Specifications* and *Source Code*, *Test Case Specifications* and *Source Code*, *Safety Analysis Results* and *Assumptions and Operation Conditions Specifications*, and between components or instances of *Source Code*. These results suggest that SECIA usually affects several artefact types and that the artefact types usually evolve together. Practitioners should pay special attention to the possible change impacts between strongly or very strongly correlated artefact types; see Figs. 4 and 5. Software developers and maintainers must carefully analyse possible ripple effects from *Source Code* evolution, and its impact on system safety. Some relationships between artefact types that the correlations represent, and their implications for SECIA, require further investigation (e.g., the relationships of *Manual V&V Results* with other artefact types).

Regarding tool support (RQ2), the overall level of SECIA automation is low and practitioners perform a significant amount of manual work. Only the median of *Source Code* is above "decision support available". We report 98 different tools used for SECIA in practice and that practitioners commonly use internal tools and basic tools such as Excel and Word (which have some clear limitations, e.g., no tool qualification). No commercial tool is widely used in industry for SECIA or used for all artefact types, and the tools seem to vary among the organizations. The amount of "fully manual" work is even more surprising if we consider the tools with some search functionality (e.g., Excel) that practitioners could use. Current SECIA practices also include the use of model-based tools and tailored extensions. A question remains if industry has a greater need for solutions that integrate safety evidence information from different tools than new SECIA tools for specific artefact types. Both product developers and tool vendors could take advantage of the design of tool chains for SECIA, which should further address traceability needs.

Practitioners face a wide variety of challenges for SECIA (RQ3), with *Insufficient tool support* having the highest median frequency. Indeed, *Tool support* is the area where the highest ratio of respondents indicated potential SECIA improvements. Nonetheless, most of the respondents mentioned *Information aspects* in their SECIA improvement suggestions, and almost half of them referred to *Process aspects*. We identified and classified 22 improvement areas, and only three correspond to *Tool Aspects*, see Fig. 11. Relationships can be established between the challenges and the improvement areas, see Table 6. Based on the results, industry often fails to fulfil SECIA traceability needs. SECIA could also benefit from a wider knowledge of safety goals in development and maintenance teams. Finally, neither experience nor the level of SECIA automation seem to greatly help practitioners in reducing the frequency of the challenges, and social aspects (e.g.,

communication and coordination) should be considered for improving SECIA.

5 Conclusion

Safety evidence change impact analysis (SECIA) is essential for any safety-critical system. It is not only recommended in most safety standards, but its inadequate management has contributed to accidents or near-accidents. New insights into how practitioners deal with SECIA are necessary in order to determine what aspects must be carefully considered in practice and to identify improvement opportunities.

This paper presents an industrial survey on SECIA. The results provide a comprehensive picture of the circumstances under which SECIA is addressed, the tool support used, and the challenges faced. Further, the survey identifies aspects that practitioners should carefully consider when performing SECIA but which are not mentioned in the text of safety standards, such as that certain artefact types usually co-evolve (see Fig. 5), tools that can be used for SECIA (see Section 4.2.1), and possible improvement areas (see Fig. 11).

The survey results suggest that most practitioners deal with SECIA during system development and mainly from system specifications. Most survey respondents have performed SECIA in a wide variety of situations and changes in the body of safety evidence seem to usually affect several artefact types. Tool support is used in industry for all artefact types, but there appear to be many improvement opportunities. The level of automation in the process is low and insufficient tool support is the most frequent challenge. Nonetheless, SECIA might further benefit from improvements on information aspects rather than on tools aspects.

The results confirm insights provided in prior research. For example, requirements specifications appear to play a major role in SECIA, and practitioners expect improvements in tool support. Nonetheless, this survey is the first study that provides strong empirical evidence of how often the phenomena occur. More importantly, the results report on phenomena for which no evidence existed (e.g., use of internal tools for SECIA from all the artefact types studied), and suggest frequencies of phenomena in industry that were very likely unexpected (e.g., re-certification for different application domains) or that can raise some concerns about current industrial practices (e.g., safety case evolution management seems to often be inadequate).

In terms of software maintenance, we expect that our findings can help software developers gain further awareness and understanding of the need for carefully analysing the safety implications of software changes. Although no strong evidence of co-evolution of source code and safety-targeted artefact types was found, software safety risks and how software contributes to system safety are prime concerns in many application domains.

Several areas for future research can be identified from the results. Some examples are the study of SECIA needs for safety-targeted artefact types, the analysis of how tools such as Excel and Word are used for SECIA, and the definition of effective and efficient guidelines for tackling traceability-related SECIA challenges. Furthermore, the results highlight several aspects where industry could clearly benefit from improvements, such as tools that support SECIA for any artefact type and an increase in their level of automation. Some aspects that have not been addressed in the survey (e.g., a deeper study of SECIA activities and their level of automation) could also trigger future work.

The survey represents a significant milestone for other research in which we are currently involved, including cross-domain and evolutionary safety assurance and certification [13], component-based impact analysis [70], and recommendation-driven impact analysis [6] for critical systems. The results of the survey help us to identify directions for future work.

We plan to further investigate the potential relationships between different artefact types and their implications for impact analysis, how to improve safety case evolution management, and technologies for integrating safety evidence information from different sources. Finally, we would like to complement the survey results with insights from case studies in which we further analyse SECIA practices in specific companies or projects.

Acknowledgments

The research leading to this paper has received funding from the FP7 programme under the grant agreement n° 289011 (OPENCOSS), from the Research Council of Norway under the projects Certus-SFI and EvolveIT, and from the Industrial Excellence Center EASE - Embedded Applications Software Engineering. The authors would like to thank the people that participated in the pilot studies, the respondents of the survey, and the reviewers of the paper, especially David Callele for his suggestions towards improving the readability of the survey instrument and of the paper.

References

[1] M. A. Babar, A. Tang, I. Gorton, and J. Han, "Industrial perspective on the usefulness of design rationale for software maintenance: A survey," in *Proc. 6th Int. Conf. Quality Softw.*, 2006, pp. 201–208.

[2] E. Bjarnason, P. Runeson, M. Borg, M. Unterkalmsteiner, E. Engström, B. Regnell, G. Sabaliauskaite, A. Loconsole, T. Gorschek, and R. Feldt, "Challenges and practices in aligning requirements with verification and validation: A case study of six companies," *Empirical Softw. Eng.*, vol. 19, no. 6, pp. 1809–1855, 2014.

[3] S. Björnander, R. Land, P. Graydon, K. Lundqvist, and P. Conmy, "A method to formally evaluate safety case evidences against a system architecture model," in *Proc. 23rd IEEE Int. Symp. Softw. Rel. Eng. Workshops*, 2012, pp. 337–342.

[4] S. A. Bohner and R. S. Arnold, *Software Change Impact Analysis*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1996.

[5] M. Borg, O. Gotel, and K. Wnuk, "Enabling traceability reuse for impact analyses: A feasibility study in a safety context," in *Proc. 7th Int. Workshop Traceability Emerging Forms Softw. Eng.*, 2013, pp. 72–78.

[6] M. Borg and P. Runeson, "Changes, evolution and bugs—Recommendation systems for issue management," *Recommendation Systems in Software Engineering*, M. P. Robillard, W. Maalej, R. J. Walker, and T. Zimmermann, eds., New York, NY, USA: Springer, 2014, pp. 477–509.

[7] M. Borg, P. Runeson, and A. Ardö, "Recovering from a decade: A systematic mapping of information retrieval approaches to software traceability," *Empirical Softw. Eng.*, vol. 19, no. 6, pp. 1565–1616, 2014.

[8] J. Buckley, T. Mens, M. Zenger, A. Rashid, and G. Kniesel, "Towards a taxonomy of software change," *J. Softw. Maintenance*, vol. 17, no. 5, pp. 309–332, 2005.

[9] J. M. Carrillo-de-Gea, J. Nicolás, J. L. Fernández-Alemán, A. Toval, C. Ebert, and A. Vizcaíno, "Requirements engineering tools: Capabilities, survey and assessment," *Inform. Softw. Technol.*, vol. 54, no. 10, pp. 1142–1157, 2012.

[10] CENELEC, *Railway applications. Communication, signalling and processing systems. Safety related electronic systems for Signalling, EN 50129*, 2003.

[11] J. L. de la Vara, M. Borg, K. Wnuk, and L. Moonen, "Survey on safety evidence change impact analysis in practice: Detailed description and analysis," Simula Research Laboratory, Tech. Rep. Nov. 2014. https://www.simula.no/files/publications/files/iasurvey_techrep.pdf

[12] J. L. de la Vara, S. Nair, E. Verhulst, J. Studzizba, P. Pepek, J. Lambourg, and M. Sabetzadeh, "Towards a model-based evolutionary chain of evidence for compliance with safety standards," in *Proc. Workshop Next Generation Syst. Assurance Approaches Safety-Critical Syst. Workshop*, 2012, pp. 64–78.

[13] J. L. de la Vara, A. Ruiz, K. Attwood, H. Espinoza, R. K. Panesar-Walawege, A. Lopez, I. del Rio, and T. Kelly, "model-based specification of safety compliance needs: A holistic generic metamodel," *Inform. Softw. Technol.*, vol. 72, pp. 16–30, 2016.

[14] R. de Lemos, "Safety analysis of an evolving software architecture," in *Proc. 5th IEEE Int. Symp. High-Assurance Syst. Eng.*, 2000, pp. 159–168.

[15] R. M. de Mello, P. C. da Silva, and G. H. Travassos, "Investigating probabilistic sampling approaches for large-scale surveys in software engineering," *J. Softw. Eng. Res. Develop.*, vol. 3, no. 1, p. 8, 2015.

[16] D. Durisic, M. Nilsson, M. Staron, and J. Hansson, "Measuring the impact of changes to the complexity and coupling properties of automotive software systems," *J. Syst. Softw.*, vol. 86, no. 5, pp. 1275–1293, 2013.

[17] S. Easterbrook, J. Singer, M. A. Storey, and D. Damian, "Selecting empirical methods for software engineering research," *Guide to Advanced Empirical Software Engineering*, F. Shull, J. Singer, and D. I. K. Sjøberg, eds., New York, NY, USA: Springer, 2008, pp. 285–311.

[18] S. Ferreira, D. L. Shunk, J. S. Collofello, G. T. Mackulak, and A. Dueck, "Reducing the risk of requirements volatility: Findings from an empirical survey," *J. Softw. Maintenance*, vol. 23, no. 5, pp. 375–393, 2011.

[19] A. Fink, *How to Design Survey Studies*. 2nd ed., Newbury Park, CA, USA: Sage, 2003.

[20] A. Fink, *The Survey Handbook*. 2nd ed., Newbury Park, CA, USA: Sage, 2003.

[21] R. Goeritzer, "Using impact analysis in industry," in *Proc. 33rd Int. Conf. Softw. Eng.*, 2011, pp. 1155–1157.

[22] D. E. Gray, *Doing Research in the Real World*. 3rd ed., Newbury Park, CA, USA: Sage, 2014.

[23] T. L. Hardy, *Software and System Safety: Accidents, Incidents, and Lessons Learned*. 2nd ed., BookLocker.com, Bloomington, IN, 2014.

[24] T. L. Hardy, *The System Safety Skeptic: Lessons Learned in Safety Management and Engineering*. 2nd ed., BookLocker.com, Bloomington, IN, 2014.

[25] M. Hinchey and L. Coyle, "Evolving critical systems: A research agenda for computer-based systems," in *Proc. 17th IEEE Int. Conf. Workshops Eng. Comput.-Based Syst.*, 2010, pp. 430–435.

[26] ISO, Road vehicles – Funcitonal safety, ISO 26262, 2011.

[27] P. Jamshidi, M. Ghafari, A. Ahmad, and C. Pahl, "A framework for classifying and comparing architecture-centric software evolution research," in *Proc. 17th Eur. Conf. Softw. Maintenance Reengineering*, 2013, pp. 305–314.

[28] C. W. Johnson and M. Bowell, "Using software development standards to analyse accidents involving eletrical, electronic or programmable, electronic systems: The blade mill PLC case study," in *Proc. 2nd Workshop Investigation Reporting Incidents Accidents*, 2003, pp. 111–127.

[29] C. W. Johnson and I. M. de Almeida, "An investigation into the loss of the Brazilian space programmes launch vehicle VLS-1 V03," *Safety Sci.*, vol. 46, no. 1, pp. 38–53, 2008.

[30] P. Jönsson and M. Lindvall, "Impact analysis," *Engineering and Managing Software Requirements*. A. Aurum and C. Wohlin, eds., New York, NY, USA: Springer, 2005, pp. 117–142.

[31] A. Kasoju, K. Petersen, and M. Mäntylä, "Analyzing an automotive testing process with evidence-based software engineering," *Inform. Softw. Technol.*, vol. 55, no. 7, pp. 1237–1259, 2013.

[32] T. Kelly, "A systematic approach to safety case management," in *Proc. Syst. Automotive Eng. World Congr.*, 2004.

[33] B. Kitchenham, S. L. Pfleeger, L. Pickard, P. Jones, D. C. Hoaglin, K. El Emam, and J. Rosenberg, "Preliminary guidelines for empirical research in software engineering," *IEEE Trans. Softw. Eng.*, vol. 28, no. 8, pp. 721–734, Aug. 2002.

[34] B. Kitchenham and S. L. Pfleeger, "Personal opinion surveys," *Guide to Advanced Empirical Software Engineering*, F. Shull, J. Singer, and D. I. K. Sjøberg, eds., New York, NY, USA: Springer, 2008, pp. 63–92.

[35] A. J. Kornecki and J. Zalewski, "Certification of software for real-time safety-critical systems: State of the art," *Innovations Syst. Softw. Eng.*, vol. 5, no. 2, pp. 149–161, 2009.

[36] S. Lehnert, "A review of software change impact analysis," Ilmenau Univ. Technology, Ilmenau, Germany, Tech. Rep., 2011.

[37] N. Leveson, *Engineering a Safer World: Systems Thinking Applied to Safety*. Cambridge, MA, USA: MIT Press, 2011.

[38] B. Li, X. Sun, H. Leung, and S. Zhang, "A survey of code-based change impact analysis techniques," *Softw. Testing, Verification Rel.*, vol. 23, no. 8, pp. 613–646, 2013.

[39] M. Lindvall and K. Sandahl, "How well do experienced software developers predict software change?" *J. Syst. Softw.*, vol. 43, no. 1, pp. 19–27, 1998.

[40] O. Lisagor and T. Kelly, "Incremental safety assessment: Theory and practice," in *Proc. 26th Int. Syst. Safety Conf.*, 2008.

[41] M. H. Lloyd and P. J. Reeve, "IEC 61508 and IEC 61511 Assessments—Some Lessons Learned," in *Proc. 4th IET Int. Conf. Syst. Safety*, 2009.

[42] P. Mäder, P. L. Jones, Y. Zhang, and J. Cleland-Huang, "Strategic traceability for safety-critical projects," *IEEE Softw.* vol. 30, no. 3, pp. 58–66, May/Jun. 2013.

[43] A. W. Meade and S. B. Craig, "Identifying careless responses in survey data," *Psychological Methods*, vol. 17, no. 3, pp. 437–455, 2012.

[44] M. Mistry and M. Felici, "Implementation of change management in safety cases," *Formal Aspects Safety-Critical Syst.*, 2008.

[45] Ministry of Defence, Safety management requirements for defence systems Part 1: requirements and guidance, interim defence standard 00–56 Part 1, no.5, 2014.

[46] S. Nair, J. L. de la Vara, A. Melzi, G. Tagliaferri, L. de-la-Beaujardiere, and F. Belmonte, "Safety evidence traceability: Problem analysis and model," in *Proc. 20th Int. Working Conf. Requirements Eng.: Foundation Softw. Quality*, 2014, pp. 309–324.

[47] S. Nair, J. L. de la Vara, M. Sabetzadeh, and L. Briand, "An extended systematic literature review on provision of evidence for safety certification," *Inform. Softw. Technol.*, vol. 56, no. 7, pp. 689–717, 2014.

[48] S. Nair, J. L. de la Vara, M. Sabetzadeh, and D. Falessi, "Management of evidence for compliance with safety standards: A survey on the state of practice," *Inform. Softw. Technol.*, vol. 60, pp. 1–15, 2015.

[49] S. Nair, J. L. de la Vara, and S. Sen, "A review of traceability research at the requirements engineering conference," in *Proc. 21st IEEE Int. Requirements Eng. Conf.*, 2013, pp. 222–229.

[50] OPENCOSS project, "Deliverable D6.1 - Baseline for the Evidence Management Needs of the OPENCOSS Platform," http://www.opencoss-project.eu/ (Accessed Dec 15, 2015).

[51] OPENCOSS project, "Deliverable 6.2 - Detailed requirements for evidence management of the OPENCOSS platform," http://www.opencoss-project.eu/ (Accessed Dec 15, 2015).

[52] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, "A model for types and levels of human interaction with automation," *IEEE Trans. Syst., Man, Cybern. - Part A*, vol. 30, no. 3, pp. 286–297, May 2000.

[53] J. Pedersen-Notander, M. Höst, and P. Runeson, "Challenges in flexible safety-critical software development—An industrial qualitative survey," in *Proc. 14th Int. Conf. Product-Focused Softw. Process Improvement*, 2013, pp. 283–297.

[54] L. M. Rea and R. A. Parker, *Designing and Conducting Survey Research: A Comprehensive Guide*. 4th ed., San Francisco, CA, USA: Jossey-Bass, 2014.

[55] L. Rierson, *Developing Safety-Critical Software: A Practical Guide for Aviation Software and DO-178C Compliance*. Boca Raton, FL, USA: CRC Press, 2013.

[56] C. Robson, *Real World Research*. 2nd ed., Oxford, U.K.: Blackwell, 2002.

[57] P. Rovegård, L. Angelis, and C. Wohlin, "An empirical study on views of importance of change impact analysis issues," *IEEE Trans. Softw. Eng.*, vol. 34, no. 4, pp. 516–530, Jul./Aug. 2008.
[58] RTCA, Software Considerations in Airborne Systems and Equipment Certification, DO-178C, 2011.
[59] SafeCer project, "Deliverable D1.0.1 - State-of-practice and State-of-the-art Agreed over Workgroup," http://www.safecer.eu/, 2011 (Accessed Dec 15, 2015).
[60] S. Siegel and J. Castellan, *Nonparametric Statistics for the Behavioral Sciences*. 2nd ed., New York, NY, USA: McGraw-Hill, 1998.
[61] D. Siegle, "Likert Scale," http://www.gifted.uconn.edu/ siegle/research/instrument%20reliability%20and%20validity/likert.html, 2010 (Accessed Dec 15, 2015).
[62] J. Singer, S. E. Sim, and T. C. Lethbridge, "Software engineering data collection for field studies," *Guide to Advanced Empirical Software Engineering*, F. Shull, J. Singer, and D. I. K. Sjøberg, eds., New York, NY, USA: Springer, 2008, pp. 9–34.
[63] T. Stålhane, T. Myklebust, and G. K. Hanssen, "The application of safe scrum to IEC 61508 certifiable software," in *Proc. 11th Int. Probabilistic Safety Assessment Manage. Conf. Annu. Eur. Safety Rel. Conf.*, 2012, pp. 6052–6051.
[64] Y. Tao, Y. Dang, T. Xie, D. Zhang, and S. Kim, "How do software engineers understand code changes?—An exploratory study in industry," in *Proc. 20th ACM SIGSOFT Symp. Foundations Softw. Eng.*, 2012, Art. no. 51.
[65] F. Törner and P. Öhman, "Automotive safety case: A qualitative case study of drivers, usages, and issues," in *Proc. 11th IEEE High Assurance Syst. Eng. Symp.*, 2008, pp. 313–322.
[66] N. Tracey, A. Stephenson, J. Clark, and J. McDermid, "A safe change oriented process for safety-critical systems," in *Proc. Int. Workshop Softw. Change Evolution*, 1999.
[67] A. von Knethen and M. Grund, "QuaTrace: A tool environment for (semi-) automatic impact analysis based on traces," in *Proc. 19th Int. Conf. Softw. Maintenance*, 2003, pp. 246–255.
[68] D. R. Wallace and D. R. Kuhn, "Failure modes in medical device software: An analysis of 15 years of recall data," *Int. J. Rel., Quality Safety Eng.*, vol. 8, no. 4, pp. 351–371, 2001.
[69] C. Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, and B. Regnell, *Experimentation in Software Engineering*. 2nd ed., New York, NY, USA: Springer 2012.
[70] A. R. Yazdanshenas and L. Moonen, "Fine-grained change impact analysis for component-based product families," in *Proc. 28th IEEE Int. Conf. Softw. Maintenance*, 2012, pp. 119–128.
[71] H. Zhang, J. Li, L. Zhu, D. R. Jeffery, Y. Liu, Q. Wang, and M. Li, "Investigating dependencies in software requirements for change propagation analysis," *Inform. Softw. Technol.*, vol. 56, no. 1, pp. 40–53, 2014.

Jose Luis de la Vara received the BEng degree in computer science, the MSc degree in software engineering, formal methods and information systems, and the PhD degree in computer science from the Technical University of Valencia, Spain, in 2006, 2008, and 2011, respectively. He joined Carlos III University of Madrid, Spain, as a visiting professor in 2015. Prior to that, he worked at the PROS Research Centre of the Technical University of Valencia and at the Software Engineering Department of Simula Research Laboratory, Norway. His research interests include requirements engineering, business process management, safety assurance and certification, model-driven development, and empirical software engineering. He has contributed to more than 40 international peer-reviewed publications on these topics.

Markus Borg received the MSc degree in computer science and engineering and the PhD degree in software engineering, both from Lund University in 2007 and 2015, respectively. He is a senior researcher at the Software and Systems Engineering Laboratory, SICS Swedish ICT AB. His research interests include alleviating information overload in large-scale software development, with a focus on increasing the level of automation in the inflow of issue reports. Prior to his PhD studies, he worked as a development engineer with ABB in safety-critical software engineering. He is a member of the IEEE.

Krzysztof Wnuk received the MSc degree from Gdansk University of Technology, Poland and the PhD degree from Lund University, Sweden, in 2006 and 2012, respectively. He is an assistant professor at the Software Engineering Research Group (SERL) of Blekinge Institute of Technology, Sweden. His research interests include market-driven software development, requirements engineering, software product management, decision making in requirements engineering, large-scale software, system and requirements engineering and management and empirical research methods. He is interested in software business, open innovation and open source software. He works as an expert consultant in software engineering for the Swedish software industry.

Leon Moonen received the MSc degree and the PhD degree in computer science from the University of Amsterdam in 1996 and 2002, respectively. He is a senior research scientist in the Software Engineering department at Simula Research Laboratory, Norway. His research aims at creating better techniques and tools to support the understanding, assessment and evolution of large industrial software systems. This work combines several subfields of software engineering, such as program comprehension, reverse engineering, program analysis, software visualization and empirical software engineering. Currently ongoing projects include recommendation systems to support smarter evolution and testing of safety-critical cyber-physical product families, and software analytics for measuring and managing technical debt (i.e., software quality and maintainability assessments). He is a member of the ACM, the IEEE Computer Society, EAPLS and the ERCIM Working Group on Software Evolution.