Requirements Engineering for Safety-Critical Systems: An Interview Study with Industry Practitioners

Luiz Eduardo G. Martins, Tony Gorschek, *Member, IEEE*

Abstract— We have conducted in-depth interviews with experienced practitioners in the Safety-Critical Systems (SCS) domain in order to investigate several aspects related to requirements specification and safety analysis for SCS. We interviewed 19 practitioners from eleven SCS companies in different domains with the intention of verifying which approaches they use day-to-day, and what their perceptions are in relation to the approaches used to elicit, analyze, specify and validate safety requirements. The aim of this study is to obtain an in-depth understanding of how requirements engineering is carried out in companies that develop SCS.

Index Terms—Requirements, Specification, Software and System Safety, Requirements Engineering, Safety Critical Systems, Software Engineering, SCS

1 INTRODUCTION

ONE of the biggest challenges for companies that develop safety-critical systems (SCS) is to establish complete, correct, unambiguous, testable, and yet understandable requirements for all stakeholders [1]. This is crucial for the IT mainstream, but it is even more important for companies developing SCS, particularly considering the product/system safety certification processes with which they have to comply [1]. Several SCS companies have to comply with the international safety standards and regulations in order to offer their systems and products to the market.

Beyond compliance and even more importantly the literature on SCS has reported many cases where systems have failed due to the poor requirements specification or misunderstanding during the requirements definition and validation [1][17][20], leading to accidents that cause serious damage to the environment, injury to people and even loss of life [2][3][4]. When accidents happen they have a strong negative impact on the companies responsible for the associated SCS, resulting in financial losses and degradation of trust. Good requirements are essential for any system to be developed and even more for safety-critical systems. The lack of clear and precise requirements has been one of the main sources of errors in such systems. Moreover, the lack of clear and precise requirements makes it impossible to perform adequate verification of safety properties in SCS.

The SCS currently developed are increasingly complex, and the safety certification processes defined by governments and international agencies are becoming more difficult and expensive to follow. The requirement documents and related processes to specify the system requirements play a very important role during the safety certification processes [22] for complying with both the process-based and the goal-based standards. Moreover, with the system functions moving increasingly from hardware to software, the certification processes are becoming even more complex and the engineering areas of systems engineering (traditionally associated with SCS) and software engineering in general (and requirements engineering in particular) are colliding and becoming one and the same as software engineers and systems engineers collaborate in product development.

We have conducted in-depth interviews with experienced practitioners in the SCS area in order to investigate several aspects related to requirements specification and safety analysis for SCS. We interviewed 19 practitioners from eleven SCS companies among six different domains – defence & military aerospace, automotive, medical devices, industrial machinery, telecom, and maritime - with the intention of determining which requirement approaches they use day-to-day, and what their perceptions are in relation to the safety requirements approaches available today. Considering the 19 practitioners that participated in our study, 14 of them have seven or more years of experience in SCS domains, and 5 of them have from two to four years of experience. This interview study was performed in 2015 and 2016.

Based on the results obtained from this interview study we present and discuss at the end of this paper some recommendations for the companies, such as: (i) improving

- *L.E.G. Martins is with the Institute of Science and Technology, Federal University of São Paulo, São José dos Campos 12231-280, Brazil. E-mail: legmartins@unifesp.br.*
- *T. Gorschek is with the School of Computing, Blekinge Institute of Technology, Karlskrona 37179, Sweden. E-mail: tony.gorschek@bth.se.*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSE.2018.2854716, IEEE Transactions on Software Engineering

the requirements elicitation approach; (ii) improving meetings and coordination; (iii) adopting requirements management tools; (iv) using safety standards to specify requirements; (v) improving communication process; (vi) improving certification process; (vii) adopting models and templates, and others.

The remainder of this paper is organized as follows: related work is presented in Section 2. Section 3 describes the research methodology, and Section 4 presents the results and analyses of the findings. Section 5 brings a summary of the main conclusions.

2 Related Work

Requirements engineering (RE) is an important topic which is gaining increased attention in the SCS area. Special attention has been given to the relationship among safety requirements, safety standards and certification processes, as well as how the safety requirements influence costs and damages in SCS.

Sujan et al. [14] reviewed safety case practices in six UK industries and identified drivers and developments in the adoption of safety cases. The review considered six safety–critical industries: automotive, civil aviation, defence, nuclear, petrochemical and railways. They argue that safety cases might best be used in healthcare to provide an exposition of risk rather than as a regulatory tool to demonstrate acceptable levels of safety. Beckers et al. [15] discuss the creation of a functional safety concept (FSC) in which they show how the functional safety requirements are systematically derived. They show the applicability of the proposed method by executing the method steps to a case study, which is an electronic steering column lock system.

Wu et al. [16] presented a modelling methodology including a UML profile for specifying safety requirements on a component-based architecture model and a set of design guidelines on avionics software. These safety requirements were identified from both standards - mainly DO-178B/C - and current engineering practices in the domain of avionics system. The researchers applied the methodology on industrial autopilot systems and several previously uncaught faults were revealed. Abdulkhaleq et al. [17] presents a comprehensive safety engineering approach based on STPA (Systems-Theoretic Processes Analysis), including software testing and model checking approaches for the purpose of developing safe software. The authors defend that the proposed approach can be embedded within a defined software engineering process or applied to existing software systems, which allow software and safety engineers integrate the analysis of software risks with their verification. The proposed approach is illustrated with an automotive software controller application.

Panesar-Walawege et al. [18] proposed a model for capturing both the information requirements for demonstrating compliance with IEC 61508 and the traceability links necessary to create a seamless chain of evidence. In this work they further describe how their generic model can be specialized according to the needs of a particular context, and discuss some important ways in which the model can facilitate software certification. Hutchinson et al. [45] presented extensive results from a survey Model-Driven Engineering practices in industry. Additionally, they complement the results with qualitative data obtained from semi-structured and in-depth interviews performed with the practitioners in industry. Zheng et al. [46] presented results from semi-structured interviews with nine practitioners across four continents in relation to the use of formal methods in cyber-physical systems verification and validation.

Sadraei et al. [47] performed an empirical study that examines requirements engineering practices in 16 Australian companies. They collected data from 28 software projects at the companies. The most of the projects were aimed at the development of a software product to support the daily business of the companies. In contrast to our research, the survey authors investigated the requirements engineering practices analyzing the distribution of the RE effort amongst activities, the distribution of RE activities, and RE awareness. In our interview study we are more interested in knowing what approaches the practitioners have adopted to perform their RE activities. Although the referred study was not focusing on RE practices specifically for SCS, some findings go to the same direction as ours. For example, the authors mentioned that their findings "indicate that still many RE activities are performed implicitly". We found the same tendency in the companies we have studied, even they being companies developing SCS.

Fernàndez and Wagner [48] have conducted research with a long-term goal of establishing an empirically sound basis for understanding practical trends and problems in requirements engineering. They report on the design of the family of surveys, its underlying theory, and the full results obtained from Germany with participants from 58 companies. The participants of the survey have experience in several software domains, mostly in IT consulting, custom software development, project management consulting, and software process consulting. Among the interesting finds obtained from the survey, the authors pointed that incomplete requirements, communication flaws, and terminological problems are among the most critical problems selected by the survey participants. These problems were also reported as important issues during our interview study, as discussed in sections 4.8, 4.9, 4.10 and 4.12 of this paper.

Liu et al. [49] performed a survey of requirements practices in China. The web-based survey of requirements engineering practices focused on requirements elicitation techniques and requirements representation techniques. They collected answers from 377 participants coming from 237 software companies or research organizations. The business areas of the involved companies cover various industry segments including banking, healthcare, power generation, telecom, retail and electronics. Among the findings reported by the authors, the face-to-face meetings were indicated as the main approach to elicit requirements.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSE.2018.2854716, IEEE Transactions on Software Engineering

This finding is in accordance with our findings. In our interview study when we asked practitioners what approaches they used to elicit requirements, the most often mentioned approach was interview, which is a type of face-to-face meeting (see the discussion presented in section 4.1 about requirements elicitation). In relation to the requirements representation techniques investigated in the referred survey, they found that diagrams were the most often selected option by the respondents. Our findings in relation to this topic point in a different direction, being the requirements textual representation the preferred option reported by the SCS practitioners.

Despite the fact that these last three mentioned works focus on requirements engineering practices, they are not specific for SCS. The main difference of this article in relation to those mentioned before is that we only considered companies in the SCS domain. As we can see there are some works related to requirements engineering practices in industry, reinforcing the importance of this research topic. However, we believe that much more should be done to get the real perceptions of the practitioners in relation to the approaches they use to elicit, specify and validate safety requirements. The contribution of this article is precisely in this direction.

In order to make clear some adopted safety terms and concepts used in this paper, we present the following definitions. **Accident** - An undesirable negative event involving damage, loss, suffering or death. **FMEA** – Failure Modes and Effects Analysis. **FTA** – Fault Tree Analysis. **Hazard** - A system state that might, under certain environmental conditions, lead to a mishap. Hence, a hazard is a potentially dangerous situation that may lead to an accident. **Safety Requirement** - A requirement that describes the constraints or actions to support and improve the system safety.

3 Research Methodology

In order to conduct the investigation of requirements engineering in SCS we chose to apply a qualitative research approach adopting in-depth semi structured interviews [10], [13] as the strategy to reach the goals of the study.

TABLE 1
RESEARCH QUESTION

Research Questions	**Aim**
RQ1: What are the main challenges in relation to the requirements engineering of SCS aspects?	To get an overview of the main challenges in relation to the requirements engineering in the SCS domain.
RQ1.1: What are the approaches used during the requirements elicitation, analysis & negotiation, specification, validation, and/or management of requirements or equivalent, with special focus on safety requirements?	To identify techniques, methods, models and/or processes used by the practitioners to elicit, analyse, specify, validate and manage safety requirements.
RQ1.2: What are the benefits, challenges, and shortcomings associated with these approaches?	To identify associated benefits and issues to the used approaches.
RQ1.3: To what extent have safety aspects and specific safety engineering approaches been integrated into the requirements engineering process?	To verify how the safety analysis and requirements engineering processes are integrated.
RQ1.4: What issues have emerged from the requirements communication throughout SCS lifecycle?	To identify issues in relation to requirements communication among stakeholders throughout SCS lifecycle. To cross check the practitioner's view with the SLR results in relation to the requirements communication issues.
RQ2: What certification process is the company subject to?	To identify the certification process which the company is subject to become its products/systems marketable.
RQ2.1: What are the challenges in relation to the certification process?	To identify the challenges in relation to certification process.
RQ2.2: How does the certification processes impact the requirements engineering of the SCS?	To identify the relationship between requirements engineering and certification processes.

The aim of this study is to obtain in-depth understanding of how requirements engineering has been carried out in companies that develop SCS, i.e.: the state-of-the-practice of requirements engineering in SCS industry. Table 1 shows the research questions that drove our investigation. We are interested in understanding the main challenges in relation to the requirements engineering of SCS (RQ1) according to the practitioners' perspective. To support this general research question we investigated the approaches used by the practitioners to elicit, analyse and negotiate, specify and validate requirements, with special focus on safety requirements (RQ1.1), as well as the benefits and shortcomings associated with these approaches (RQ1.2). Additionally, we investigated to what extent the safety engineering approaches are integrated into the requirements engineering process (RQ1.3). Moreover, we are interested in knowing how the requirements are communicated throughout the SCS lifecycle between different stakeholders, process steps and functions (RQ1.4).

Another aspect of interest is the relationship between the requirements engineering process and the certification processes with which the SCS companies have to comply. In order to accomplish it we need to know what types of certification processes the companies are subjected to (RQ2); what challenges in relation to the certification processes should be overcome according to the practitioners' view (RQ2.1); and how the certification processes impact the requirements engineering of the SCS (RQ2.2).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSE.2018.2854716, IEEE Transactions on Software Engineering

TABLE 2
OVERVIEW OF THE COMPANIES THAT PARTICIPATED IN THE INTERVIEW STUDY

	Domain	Type of Customer	# of Employees	Development Process	# of Req. In Typical projects	% of Safety Req. In Typical Projects
C1	Defence & Military Aerospace	B2G	~14500	Waterfall - Iterative	~1000	~5%
C2	Automotive	B2B	~15000	Waterfall & Agile - Scrum	~2000	~20%
C3	Medical Device	B2B & B2C	21	Waterfall - Iterative	~200	~70%
C4	Defence & Military Aerospace	B2G & B2B	~1500	Waterfall - Iterative	~250	~2%
C5	Automotive	B2B	20	Waterfall & Agile - Scrum	~2000	~20%
C6	Industrial Machinery	B2B	~23000	V-Model & Agile	~300	~10%
C7	Industrial Machinery	B2B	~4000	V-model - Iterative	~300	~10%
C8	Industrial Machinery	B2B	14	Agile - Scrum	~20	~20%
C9	Telecom	B2B	~1900	Waterfall - Iterative	~100	Unknown
C10	Maritime	B2G & B2B	800	Agile - Scrum	~200	~20%
C11	Defence & Military Aerospace	B2G	10	V-Model	~150	~25%

3.1 Study Design

This investigation was divided into three parts: planning, data collection and analysis.

Planning. The sampling strategy adopted in this study was a combination of maximum variation sampling and convenience sampling [9], [10] using our industrial collaboration network and variation in the domain of the companies. The companies that participated work in six different domains: defence & military aerospace, automotive, medical device, machinery, telecom, and maritime. All 11 companies that participated develop SCS. They vary in relation to size (in number of employees and number of requirements in typical projects) and type of customers. The companies are categorized based on the type of customers [12]: business-to-business (B2B), business-to government (B2G), and business-to customer (B2C). The characterization of the companies can be seen in Table 2. For confidentiality reasons we cannot show more details about the companies, according to the recommendations from Ivarsson and Gorschek [11].

TABLE 4
ASSOCIATION BETWEEN RQS AND OPEN-ENDED AND CLOSE-ENDED QUESTIONS

RQs	Open-ended questions	Close-ended questions
RQ1	Q12.1	
RQ1.1	Q5.1, Q5.2, Q5.3, Q5.4	Q1, Q2, Q3, Q4
RQ1.2	Q6.1, Q6.2	Q5, Q6, Q10, Q11, Q12
RQ1.3	Q7.1, Q7.2, Q7.3, Q7.4, Q7.5	Q7, Q8, Q9
RQ1.4	Q8.1, Q8.2, Q8.3	Q9, Q10, Q11, Q12, Q13
RQ2	Q9.1, Q9.2, Q9.3	
RQ2.1	Q11.1	Q14, Q15, Q16
RQ2.2	Q10.1, Q10.2, Q10.3, Q10.4	Q14, Q15, Q16

The researchers contacted a "broker" (someone with whom the researchers had a well-established connection) at each company who identified practitioners to be interviewed. Then we searched and queried the companies internally to find senior experts in SCS. Each company tried to provide at least two participants: one subject responsible to specify the requirements labelled as "requirements supplier" (RS), e.g.: requirements engineer or safety engineer; and other subject that received the requirements to use them throughout SCS lifecycle, labelled as "requirements client" (RC), e.g.: product manager, system engineer, software engineer, tester, auditor. The brokers were independent of the requirements clients /suppliers used in the interviews. In some companies only one subject was interviewed as he/she could represent both the roles of RS and RC. A total of 19 practitioners participated in the interview study. The characterization of the practitioners can be seen in Table 3 (see Appendix 1).

Data Collection. The strategy used for data collection was based on semi-structured interviews [13]. We used a questionnaire during the interviews, which was composed by open-ended and close-ended questions (see Appendix 2 for details). Each interview varied from 50 to 90 minutes, and was conducted in the interviewee's workplace. With the permission of the interviewees all interviews were recorded using an audio recorder, subsequently transcribed into spreadsheets and used in the analysis process. The complete transcription of each interview was sent to the interviewee to check the content and provide amendments or corrections, if necessary.

In the beginning of each interview the study aim was explained to the interviewee, as well as a general consideration about requirements engineering for SCS. After this short prologue all open-ended questions were discussed in detail. At the end the interviewee answered a set of close-ended questions (the answers were collected in Likert's scale format).

Analysis. In this study data was analysed based on the content analysis [44], which includes marking and discussing of interesting written extensive notes. The analysis process was performed based on a set of answers obtained from the 19 interviewees (all answers were considered during the analysis process). They answered the open-ended and close-ended questions, which are associated to one or more RQs. Table 4 shows the association between the RQs and the open-ended and close-ended questions. The interviewees did not have access to the RQs, but only to the questions available in the questionnaire (available in Appendix 2).

We coded all the answers obtained from the open-ended questions in order to create categories to organize the analysis process [19, 21]. For each question we examined all the answers searching for patterns that could be considered as categories. For instance, examining the answers from the question Q5.1 "How do you elicit requirements?", we looked for approaches used by the practitioners to elicit requirements. We found 13 different approaches mentioned by the practitioners, which were considered as categories of how they elicit requirements. For the question Q7.3 "How do you do the system/hazard analysis" we consider each method or technique mentioned by the practitioners as a category, in this case we found nine categories. We proceeded this way for all the questions of the part B of our questionnaire. The results from the analysis process are presented and discussed in Section 4.

3.2 Validity

In this section we discuss the threats to validity in relation to the research design and data collection. In order to conduct such discussion, we adopt the four perspectives of validity and threats presented by Wohlin et al. [19].

Conclusion Validity. Threats to conclusion validity are concerned with issues that relate the treatment and the outcomes of the study, which include for example the choice of size of the sample, and care taken in the implementation and measurement of a study [19]. In our research, the variables were measured through interviews, including open-ended and close-ended questions, where the participants were asked to express their own perceptions and point-of-view. The interviews were conducted at different companies and each interview happened in only one work session, thus avoiding bias through subjects discussing the interview amongst themselves. In addition, we did our best to select senior experts at each company to avoid our sample not being mature enough to have the best knowledge about our area of study. Thus, our sample might not have been representative for each company, rather more senior and experienced than subjects chosen from a purely random sample. In addition, we collected data in this manner from several companies endeavoring to also make sure that we did not get influenced too much by any one company, rather get input from several. This by no means assures a representative sample overall - but our study done more in-depth - precludes the collection of adequate datapoints to alleviate all sampling bias. We also do not claim the study being generalizable beyond the setting of our study.

One aspect that is critical for validity is the quality of the experimentation material. To ensure that the questionnaire is of high quality in order to get highly reliable measures, a pilot was carried out to avoid poor questions before conducting the interviews. In addition, our presence and live discussions with the subjects at data collection made sure that the subjects and the researchers shared a common understanding of the topic and that the questionnaire, in this case, was not a base for misunderstanding. To avoid the major threat of concluding false relationships (making false conclusions based on the interview data collected), and/or missing relationships, in fact, present, we were careful to validate our interviews and our findings/conclusions with the subjects as we performed analysis, sometimes asking for follow-up clarifications. This enables us to not overreach, but also confirm our understanding.

Internal Validity. Threats to internal validity are concerned with issues that may indicate a causal relationship, which include how the subjects are selected and treated during the experiment [19]. In our research the subjects that participated in the interview study were selected by a "broker" at each company who identified the practitioners to be interviewed. In essence as we contacted the company often by personal and well-established contacts these contacts could be considered as "brokers", then searched and queried the company internally to find senior experts in the area of study. It should be observed that relatively few experts were available in each company. This procedure may partially alleviate threats in relation to the subject selection since there was not any contact between interviewees and interviewer before the interviews.

Construct Validity. Threats to construct validity are concerned with the extent to which the setting of the experiment reflects the construct under study, which include the potential problem of evaluation apprehension, among others [19]. In our research the evaluation apprehension was ameliorated by the anonymity of the participants, as well as the guaranty that all information obtained during the interviews would be used only by the researchers.

External Validity. Threats to external validity are concerned with the possibility to generalize the experiment results outside the experiment setting [19]. Our research is characterized as a qualitative study, which rarely can be generalized beyond the actual setting where the study was performed. Moreover, considering the nature of our research the replication is not possible since identical circumstances cannot be recreated. Considering the number of participants (companies and practitioners) and the different domains they work, the findings may be generalized with a certain degree of confidence. Although we do not have many interviews or many hours, the focus of the interview subject was rather narrow, and the interviewees were to be considered as experts in their respective companies. At the companies and in the area of Requirements Engineering for SCS that also were certification experts there exists relatively few in each company and access to them for any time was hard to attain.

4 Results and Analysis

In this section we present the analysis of the results. The subsections are following the questionnaire organization, which were divided into three parts: characterization of the company and interviewee experience (Part A); open-ended questions (Part B); and close-ended questions (Part C). The complete questionnaire is presented in Appendix 2. Subsections 4.1 to 4.12 are related to the open-ended questions and subsection 4.13 is related to the close-ended questions. For each subsection we indicate the RQ that is being analysed.

4.1 Requirements Elicitation (RQ1.1)

Figure 1 shows what are the approaches currently used by the practitioners to capture requirements in their projects (requirements elicitation). As we can see "interview" [24] is by far the preferred approach for capturing requirements, followed by "market analysis". Interview is the most popular approach to capture requirements in the IT mainstream [24], and we found the same situation in the SCS companies that participated in our interview study.

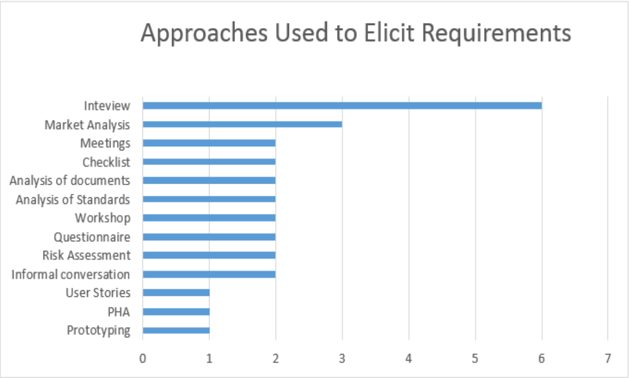

Fig. 1. Approaches used for requirements elicitation (Q5.1).

The preference for "interview" is not a surprise. However, the practitioners did not report any efforts towards improvements of the interview methods adopted by them, e.g. to use structured interviews based on a well-defined protocol. Considering the importance of the interviews as instrument to get requirements we hoped to see a more structured and organized way to conduct the interviews. According to the practitioners the interviews performed by them to capture requirements from the stakeholders are done regularly, but yet in a non-systematic way.

Only three of eleven companies use hazard analysis techniques (as PHA and Risk Assessment) [25] during requirements elicitation. The three companies are large companies with extensive experience in SCS development. Two of the three companies that use hazard analysis techniques during requirements elicitation also perform analysis of standards (ISO and IEC safety standards) during this phase. It seems to indicate that these companies are more mature and have a better perception of the influence of the requirements engineering on the safety of their products. Moreover, this may indicate that requirements team and safety team are not well integrated in most of these eleven companies, particularly in the small companies.

4.2 Requirements Analysis & Negotiation (RQ1.1)

Figure 2 shows the activities performed by the companies to analyze and negotiate requirements. "Requirements Prioritizing" and "Review meeting" are the most common activities to analyze and negotiate requirements, reported by practitioners from six companies. These two activities are intrinsically linked to each other, since the requirements prioritizing occurs during the meetings with the stakeholders. Meetings are a natural way to negotiate requirements among the stakeholders both in traditional and safety-critical systems [1]. However, the practitioners did not report whether they adopt a systematic approach to organize, conduct and deliver the results from the meetings.

It seems that the review meetings usually are performed in an informal way. Since these meetings are unstructured there are risks. For example, if decisions and decision rationale are not structured, ``waste'' or miscommunication may be introduced

Practitioners from two companies reported "Analysis of standards" as an approach to analyze the system requirements. The adoption of "Analysis of standards" seems to be a common practice in companies that have to comply the safety standards and regulations, in order to guaranty the safety of their products. Basically, the "Analysis of standards" is the cross verification between the system requirements and the norms and regulations present in the international safety standards that the companies have to follow. However, such analysis of standards is done in *ad-hoc* way without any systematic methodology or tool to support the practitioners.

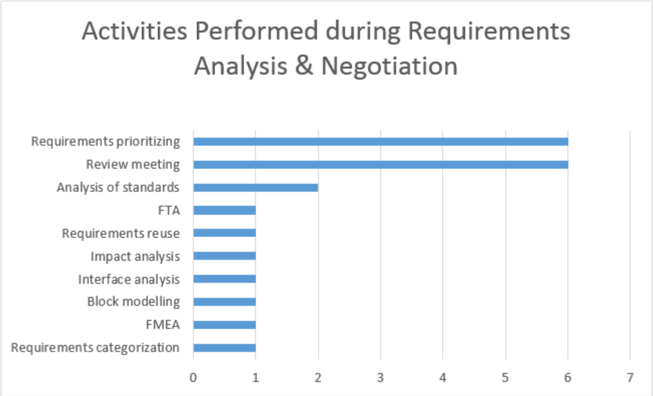

Fig. 2. Activities performed for requirements analysis & negotiation (Q5.2).

In one company only the practitioners reported that they do requirements reuse during requirements analysis. Nevertheless, this reuse is performed in a non-systematic and controlled way, but is instead performed based on the experience of their engineers. This is risky, as recognized by the practitioner according to his/her statement: "We do some requirements reuse but it is based on the previous experience without a systematic and controlled approach, which is very risky". In one company the practitioners mentioned that they use FMEA [26] during the requirements analysis. Surprisingly, in one company only the practitioners reported that they adopt regular modelling practice during requirements analysis. In this company the practitioner stated that "we use a kind of modelling of blocks to analyze the interfaces between the subsystems". Interestingly, none of the known modelling techniques in the software engineering community such as UML or SysML was reported by the practitioners as being used during the requirements analysis & negotiation. It seems that the practitioners we interviewed had little knowledge about these software/system modelling languages, specially about their potential to help them during requirements analysis & negotiation. However, the UML/SysML

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSE.2018.2854716, IEEE Transactions on Software Engineering

languages were reported as being used during the requirements specification process (as discussed in Section 4.3).

All companies have strong interaction with components/systems suppliers, but it seems they do not adopt a systematic approach to involve suppliers during requirements analysis. This means, e.g., that knowledge on the supplier side may not be utilized. Additionally, an opportunity for coordination between vendor and supplier is lost. Only review meetings were mentioned to get suppliers involved. Only in one company the practitioners mentioned that system tester participates in the meetings to review requirements, which seems indicate that companies are giving little importance to integrate requirements with the test phase. However, early involvement of testers is proven to be beneficial as discussed by Unterkalmsteiner [39].

4.3 Requirements Specification and Documentation (RQ1.1)

Figure 3 shows the most used approaches by the companies to specify and document requirements. Six companies reported that they use templates to help them in the requirements specification and documentation. UML diagrams were reported as being used in a complementary and *ad hoc* way to specify requirements (reported in five companies). However, UML diagrams are not used extensively and their use was not an institutional decision, i.e. the UML diagrams are used experimentally, just because the practitioners think their use may support the understanding of the requirements. This way of working with “design models” is not unusual, as discussed by Gorschek et al. [23].

All companies write their requirements documents in natural language. UML diagrams and other types of diagrams (as FTA and flowchart) [27] are seldom used to describe the requirements. The templates used by the practitioners to organize the requirement documents are defined by them in an *ad hoc* way, and none of the practitioners mentioned that they use templates based on the requirements engineering literature. Instead, they use their own in-house developed requirements templates. However, the safety standards are used to guide how to organize the system requirements, which can help the practitioners to define templates for requirement documents.

Only two companies mentioned that some type of formal method is used to support the requirements specification. These companies are from the industrial machinery and automotive domains. In both cases the formal language mentioned are equations used in the context of control engineering. By “formal methods” we mean any mathematically based method or technique used for the analysis, specification or validation of the system requirements. We explained our view about “formal methods” for the practitioners during the interviews. In SCS literature we find several works defending the use of formal methods for requirements specification [40, 41, 42, 43]. However, no practitioners that we interviewed adopt formal methods for specifying requirements. As one practitioner stated: “formal methods are difficult to use and demand a lot of training, it is difficult to communicate requirements based on the formal methods because the stakeholders are not prepared to understand them.”

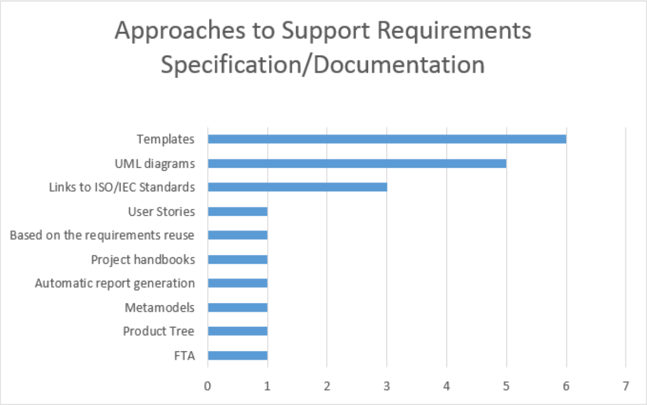

Fig. 3. Approaches used to specify/document requirements (Q5.3).

In three companies the interviewees reported that they use “links to ISO/IEC standards” as a style of requirements specification/documentation. As one practitioner stated “from the safety perspective, almost everything that we need are specified in the standards. The specification of a safety requirement uses to be short, because we refer to pre-defined concepts in the safety standards and this means we don't have to specify too much”. This practice saves a lot of work, as otherwise they would have to write the safety norm/recommendation in the requirements document [28]. However, this can bring interpretation problems because the norms/recommendation sometimes are subjective [1]. As other practitioner stated "now we see that not all vendors are reading those standards so carefully". As we can see, in requirements specification practice the standards/recommendations are a quite important artefact to communicate safety requirements.

In relation to the software tools used by the companies to support the requirements specification and documentation, 61% use only word processing & spreadsheets to support this process. Considering the dynamic nature of the requirements along the projects it is a surprise that most companies studied do not adopt a more sophisticated tool to manage their requirements. Although there was not a specific question about tools in Appendix 2, all the practitioners were explicitly asked if they adopted requirements management tool.

In more traditional software industry tool use is common [38], but it seems that in the development of SCS requirement tool use is much less common than in traditional software companies. A possible explanation for that is the background and education of the SCS people involved in requirements specification, which are mostly from system engineering background. Only 21% of the participants that were interviewed have formal education in software engineering or computer science. Not coincidently they are the practitioners that are using proper requirements engineering tools in their companies. Fig. 4 shows the tools used by the companies to specify and document requirements [50, 51, 52].

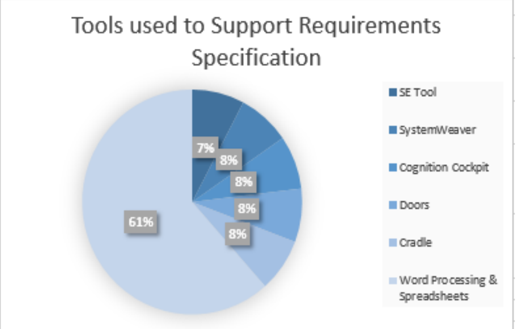

Fig. 4. Requirements specification/documentation tools.

4.4 Requirements Validation (RQ1.1)

The practitioners did not mention that they use approaches to do requirements verification; they only reported approaches used for requirements validation. It seems they do not differentiate between requirements validation and requirements verification. Therefore, in this section we only consider the approaches which are clearly used to validate requirements, which means the approaches that help the practitioners to be sure they have the proper requirements for the system. Figure 5 shows the most used approaches by the companies to validate the requirements. Again, meetings are by far the most widely used approach when the companies need to validate the system requirements. However, the practitioners did not report that they do meetings to validate requirements adopting a systematic way to conduct and deliver the results from the meetings (for instance, using documents inspection and reviews) [29]. When practitioners do meetings in an *ad hoc* way there is a high probability of not identifying relevant flaws in the specified requirements.

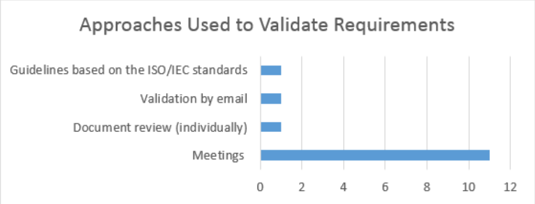

Fig. 5. Requirements validation approaches adopted by the companies (Q5.3).

Practitioners from one company only mentioned that verification and validation (V&V) people participate in the requirements validation. This is a quite surprise, considering the gains in communication and understanding of the system requirements with the early involvement of the V&V people, particularly during the requirements validation [39].

Figure 6 shows the distribution of the safety engineer participation during requirements validation reported by the companies. At the large companies it is common the participation of a safety engineer or a safety officer during the requirements validation [30]. However, that is not the case in small and medium companies, which usually don't even have a position for safety engineer or safety officer. The small companies that we studied told us that was not possible for them to afford a safety engineer. Instead the role of the safety engineer usually was played by a system engineer.

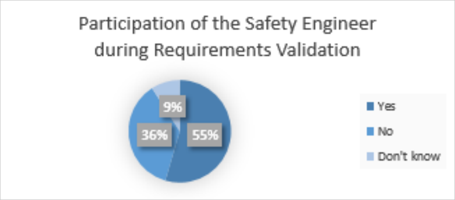

Fig. 6. Distribution of the safety engineer participation during requirements validation at the studied companies.

4.5 Special Treatment for Safety Requirements (RQ1.1)

Only two companies reported they highlight safety requirements with tags in the requirements specification. The most part of the companies do not differentiate requirements and safety requirements. As stated by one practitioner: "we take the requirements into account and assemble people with different experience to do the risk analysis, i.e., the traditional risk assessment. We don't put a tag saying this requirement is safety-critical and that is not," this may indicate that the safety requirements concept is not widely adopted by the companies. Moreover, this may indicate a weak link between requirements team and safety team, whose interaction should happen mainly working on the safety requirements specification.

On the other hand, only three companies related they do not have any special treatment for safety requirements (see Figure 7). Four companies mentioned that they adopt FMEA [26] to support in the requirements analysis, despite that they do not use the term "safety requirements".

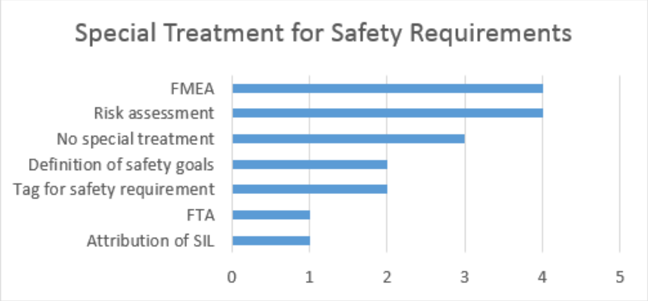

Fig. 7. Approaches used as special treatment for safety requirements (Q5.4).

Surprisingly, FTA [27] was reported by practitioners from one company only as an approach to do safety analysis and handle safety requirements. This is in contradiction with the results from our SLR [2], which indicated FTA as the most used approach for handling safety requirements in the academic research setting. This seems to indicate that there is a gap between academic research and industry practice. In our SLR [2] we found that 15.23% of the analyzed studies (23 studies out of 151) indicated the use of FTA to do safety analysis and safety requirements specification. However, that is not the reality we found in the studied companies.

4.6 Benefits and Shortcomings Associated with the Requirements Engineering Approaches (RQ1.2)

After we collected the preferences of the practitioners about the approaches used by them for eliciting, analyzing, specifying and validating requirements, we asked them what the benefits and shortcomings they realized with their requirements engineering approaches. The specific asked questions were:

- Mention two things that work in your requirements engineering approach (Q6.1)
- Mention two things that do not work (Q6.2)

When we analyzed the answers from the practitioners for these two questions we did not find any pattern. Therefore, it was not possible to create a category of analysis for the answers of these questions. However, we got interesting answers which we will highlight and discuss in the following.

One practitioner said that "we are really trying to listen to our clients and their needs. Of course we try to fulfill their needs, but it is not always possible. We have personal discussion with them. With the new customer is quite delicate, because you never know exactly what they need, or what part of our solution would be really important for them." Analyzing this quote and considering what we observed at the eleven companies we visited we may say that in safety-critical systems industry it is very common to attempt to reuse ready-made solutions.

As we can see in Figures 1 and 2 the practitioners are preferring to use interviews and reviewing meetings to elicit and analyze the requirements, however they did not report that they are using these approaches combined with a systematic requirements reuse approach. As one practitioner stated: "We do some requirements reuse but it is based on the previous experience without a systematic and controlled approach, which is very risky." Only one practitioner reported the use of a systematic requirements reuse approach during requirements analysis (see Figure 2).

Another interesting quote: "We don't want to invent/rewrite what is available at the standards and that already specify exactly what we need. We try to have our own requirements as low as possible, avoid overlapping with the standards we adopt." We observed that safety standards have a great importance in the requirements engineering process for SCS. Practitioners are using the definitions and specifications from safety standards as a large part of your own requirements specification. They do that putting "links" in the requirements documents pointing out to safety standards.

In Figure 3 we see that three practitioners reported this practice of requirements documentation. Although this practice seems to save a lot of effort during requirements documentation it brings uncertainty if the stakeholders, especially the suppliers, are really understanding and following what is specified into the safety standards. As two practitioners stated: "Maybe one of the biggest problem that we have is not in getting the requirements but is to be sure that they are being followed in the design phase, which depends on the maturity of the suppliers." And "Now we have chosen that we only have requirements saying 'looking at that standard'. It is a bit risky in some sense, the designers are supposed to know the standards but they might don't know all the details." Although the practitioners recognize that this practice is risky they are still keeping this way of requirements documentation.

4.7 Organization between Requirements and Safety Teams (RQ1.3)

Five companies do not have a safety team or even a safety officer responsible for safety in the company or in the development projects (see Figure 8). It seems that such situation is more common in small companies (we observed this in two small companies). One company mentioned that they don't have control or knowledge of how their suppliers are organized in relation to safety, however they believe the suppliers do their part, as stated by one practitioner: "Of course it gets a lot harder when something is done by the supplier. We are not involved in the supplier risk assessment. We rely on they do their jobs," which is a risky assumption when part of the system is developed by a third party.

As we can see in Figure 8 we found the there is a weak integration between requirements and safety teams. Only three companies adopt a cross-function team with the simultaneous participation of requirements engineers and safety engineers. Three companies have a safety engineer participating in the requirements team.

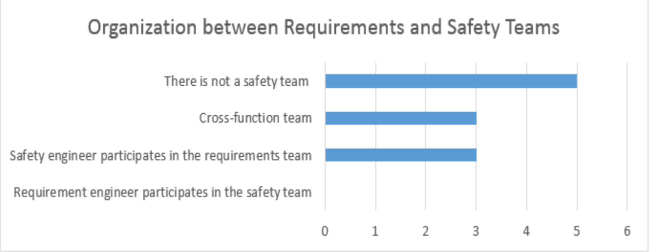

Fig. 8. Main types of organization between requirements and safety teams (Q7.1).

However, no company has a requirements engineer participating regularly in the safety team. This is surprising, considering that the hazard/safety analysis is normally done before the requirements specification would be beneficial to improve the communication between the teams the participation of the requirements engineer in the safety team since the early stages of the hazard/safety analysis.

4.8 Communication between Requirements and Safety Teams (RQ1.3)

Despite the fact that FMEA has been mentioned by four practitioners as an approach to support communication between requirements and safety teams, only in one company a proper software tool is used to integrate the FMEA results with safety requirements. The tool mentioned by the practitioner to help in the communication between requirements and safety teams is called SystemWeaver™ [50]. The absence of use of a proper tool to integrate FMEA results to safety requirements may cause difficulties in the communication among the teams, as stated by one practitioner: "the connection between FMEA and requirements are manually performed by the system engineer responsible for the requirements. The FMEA and requirements specification are performed in parallel."

The communication between safety and requirements teams seems to be regular and based on the meetings and workshops (see Figure 9). Only two companies mentioned informal communication between requirements and safety teams.

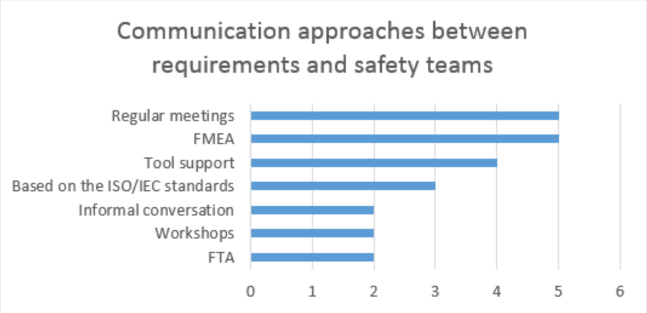

Fig. 9. Approaches supporting communication between requirements and safety teams (Q7.2).

Communication between requirements and safety teams is also supported by the safety standards, three companies reported that the ISO/IEC standards are widely used among the teams and suppliers, however this practice was observed more in large companies than in the small ones. The requirements communication based on safety standards has already been discussed in section 4.3 (in the context of requirements specifications). Again, we come across the practice of using safety standards as a vehicle for communication of requirements among stakeholders.

4.9 Safety Analysis Approaches (RQ1.3)

FMEA was the most reported approach used to support safety/hazard analysis [26], followed by standards recommendation and FTA [27] (see Figure 10). In requirements engineering literature the FTA is more often reported than the FMEA as supporting technique to help requirements and safety teams during safety/hazard analysis [2]. A possible reason for the FMEA preference in the industrial setting is the perception of the practitioners that FMEA is more practical for handling system requirements, and that it generates safety requirements in a direct way, as stated by one practitioner: "One part of the result of the FMEA would be prevention, which are requirements in themselves."

The correct identification of hazards and risks during the safety/hazard analysis is the major difficulty reported by the practitioners (see Figure 11). There is a lack of guidelines to perform such activity, consequently the practitioners tend to identify and analyze system hazards much based on their own experience, as we can see in the following statement: "We are not so structured about safety/hazard analysis, but we have experienced people, so we use their experience."

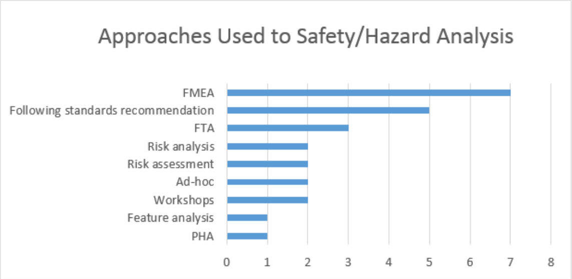

Fig. 10. Approaches adopted during the safety/hazard analysis (Q7.3).

The terminology used by safety engineers is also an important issue, because the requirements and software engineers are not familiar with safety terminology. The different terminology adopted by safety engineers and requirements engineers make the communication between them difficult.

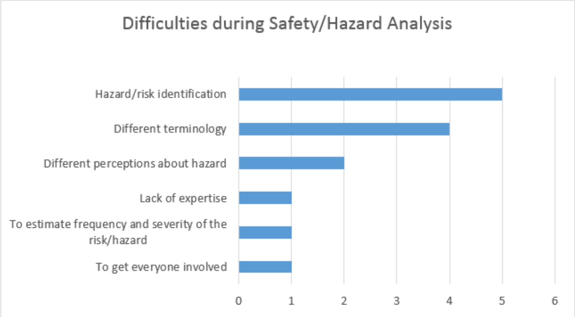

Fig. 11. Main difficulties reported during the safety/hazard analysis (Q7.4).

4.10 Requirements Communication throughout SCS Lifecycle (RQ1.4)

Although most part of the practitioners has reported that documentation is the main vehicle to communicate requirements throughout the development lifecycle [2, 31], informal conversation is very common among them (see Figure 12). As stated by two practitioners from large companies: "When someone does not understand a requirement a person conversation is performed, usually people go to the team leader, systems engineers or safety engineer to get the understanding. This is done in an informal way." And also "Much of the communication is fairly informal. If someone needs some information just go to the corridor to check it out with others and get what he/she needs."

Some reasons for the informal conversation to be so common are the following: (a) it is very difficult to maintain up-to-date documents tracking the project evolution (lack of human resources, time, and management) [28]; (b) it is difficult to write every requirement in details in order to mirror the decisions and discussion occurred during the meetings [34]; (c) the practitioners believe that their experience and tacit knowledge cannot be properly described into documents [33].

In mature companies, such as in the machinery domain, the ISO/IEC safety standards assume an important role during the SCS lifecycle. The standards are used during requirements analysis, specification & validation, safety/hazard analysis, requirements communication among development teams, and during certification processes. As discussed in Section 4.3, the safety standards are used as a way of supporting requirements specification, which complement the main requirements documents of the system by establishing links between system requirements and the norms and regulations defined in the safety standards. When the practitioners told us they use the safety standards to support their requirements specification, it does not mean that a complete safety standard is used or followed, because depending on the system or product to be developed only certain norms should be followed. The standards also have an important role during the communication with the SCS company suppliers.

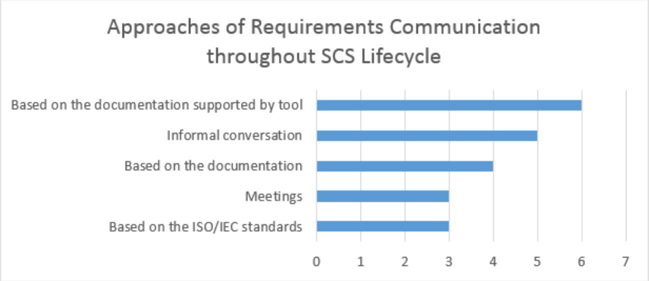

Fig. 12. Approaches adopted to communicate requirements throughout SCS Lifecycle (Q8.1).

4.11 Certification Process (RQ2, RQ2.1, RQ2.2)

Figure 13 shows the certification processes that the companies are subject to. Surprisingly, 45% of the companies are not subject to a certification process. That was the case of the companies in the Telecom, Maritime and defence domains. In the defence domain the companies have to comply the military requirements to get a sort of government authorization to deliver their products, however it is not an institutionalized certification process to verify the safety of the systems and products delivered by the companies. CE marking is a specific certification for products commercialized in European market; INMETRO is a specific certification for products in Brazilian market.

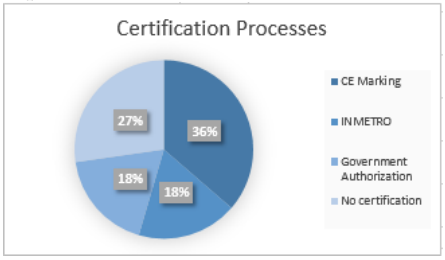

Fig. 13. The certification processes that the companies are subject to (Q9.1).

Figure 14 shows the safety standards and regulations followed by the companies. Even the companies that are not subject to certification processes usually follow some standards and regulations. That is the case of the companies in defence domain, which follow the military standard Mil-Std 882c [32]. Three companies reported that they are not obligated to follow any safety standards, these companies are in the domains of Telecom, Maritime and Metallurgy.

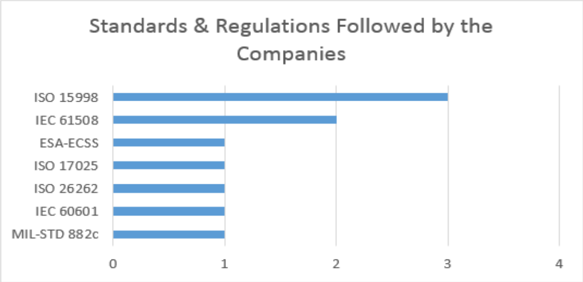

Fig. 14. Standards and regulations followed by the companies (Q9.2).

One company from the medical device domain mentioned that the certification process pushes them to define requirements with more details (they are a small and young company). It seems that the other companies didn't mention that because they are more mature in requirements engineering process, which may indicate they are more used to describe in-depth requirements.

Four companies mentioned that the certification process brings the need to link more documents and details (from standards and regulations) to the requirements documents, which increases the complexity during the requirements engineering process (see Figure 15). This kind of traceability is not necessarily supported by tools. The practitioners do much of these links “manually”. When a requirement needs to comply one norm or rule of a safety standard, they add the identification of the standard, plus the section and page of the document, together the correspondent requirement. Two companies mentioned that they have to deal with the military and government people, which increases the complexity in relation to the stakeholders’ negotiation. For these two companies the model of business is "business-to-government". Five companies mentioned that the certification process impacts how the safety evidences in the system/product documents is shown, including requirements documents [28]. “Safety evidence” in Figure 15 means that the certification process shows or assures “safety evidences” of the systems/products to the clients/users of the companies. None of the companies reported they make safety cases. Nevertheless, they were not explicitly asked about that.

In relation to the benefits of the certification process (see Figure 16) the most cited benefit is that the certification process pushes the companies to adopt the best practices of safety and requirements engineering (reported by four companies). It seems that these companies have improved the quality of their systems development lifecycle as they follow the orientation prescribed by the safety standards. Two companies reported that the certification process improves the relationship with their suppliers, because everyone must follow the same safety standards, which are used as artefacts of communicating between the company and its suppliers. Three companies answered that they do not have experience with certification process, that was the case with companies in the telecom, maritime and metallurgical domains. This is a surprise, particularly for the domains of metallurgy and maritime, which are directly involved with safety-related systems and procedures. When we interviewed a practitioner from a telecom company that offers systems and infrastructure for mobile communication, he did not consider he was working in a SCS domain. As more people move to only use mobile devices instead of also having a land-line of mobile phone, we hoped to see in the telecom domain at least some sort of safety normative to guide their systems development.

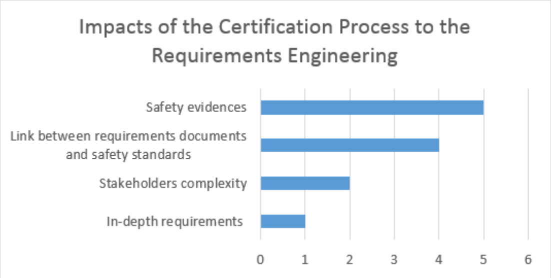

Fig. 15. Impacts of the certification process to the requirements engineering (Q10.1).

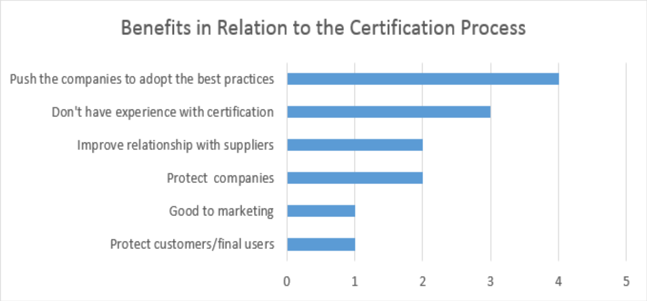

Fig. 16. Benefits in relation to the certification process (Q10.2).

Figure 17 presents the main challenges pointed out by the practitioners in relation to the certification process. To make sure that "people follow the rules" during the system development was reported by two practitioners as a challenge to be overcome. Safety standards and requirements documents define a lot of rules and recommendations that should be followed by developers and suppliers. In order to get the certification of their systems and products, the companies must show evidence that the rules and norms prescribed in the safety standards are being accomplished.

However, such evidence is usually just "statements on paper" provided by the companies during the certification process. Are the companies really following the rules? Considering the universe of companies analyzed in this study, at best the answer would be: partially. Two practitioners mentioned that "to manage a lot of information" is a challenge to be faced. They reported that it is a real challenge to manage the huge amounts of information that they have to deal with during the certification process, which makes the certification process a time consuming, exhaustive, and expensive process. The other challenges were mentioned once by different practitioners.

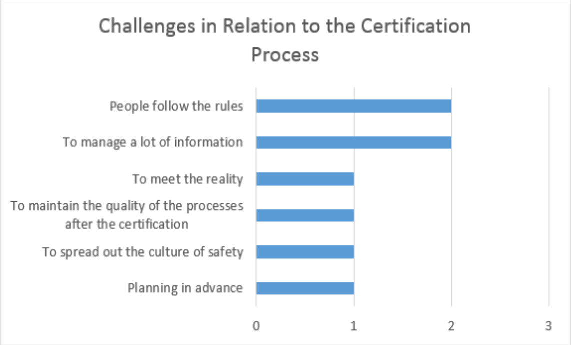

Fig. 17. Challenges in relation to the certification process (Q11.1).

4.12 Challenges in Relation to the Requirements Engineering (RQ1)

Practitioners from three companies mentioned that somehow it is necessary to produce more useful documents in order to meet the daily needs of the system developers. It seems that the requirements documents still are more to show compliance than to be really used by development teams. As one practitioners stated: "The designers and programmers know what to do, of course the requirements specification is there to drive it, but in details it is very common that the requirements don't really drive the designers/programmers." According to this statement the developers "know what to do", however not because the requirements specification is driving them. So, what is really driving the developers to know what to do? According to our investigation the answer might be: (1) the expertise of the practitioners based on their development experience gathered along years; and (2) the norms and orientation provided by the safety standards. Thus, the challenge here is to capture and manage the practitioners' expertise in such a way that it can be preserved and transmitted to the other people throughout the SCS lifecycle. The current ways how the requirements specification is being produced are not accomplishing that.

This implies that a lot of "requirements" used are not explicitly specified, but informally communicated. This is indeed a potential issue in relation to compliance, but also the company developing the product also realize that requirements can be a powerful tool for communication and achieving better products, avoiding assumptions and misunderstandings. Other possible sources of problems can be, for example, lack of communication between parties (internally and between supplier and customer). By having a good-enough requirements engineering process some of these communication and coordination issues could be resolved. However, to utilize and establish a "good-enough" level for requirements engineering (not overdoing it, keeping compliant, and using requirements as a communication and coordination tool) it is important that the knowledge in relation to requirements engineering as a discipline is explored and improved among the stakeholders in question.

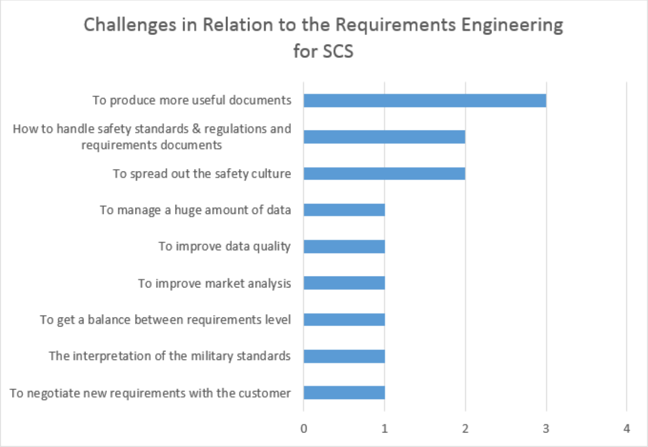

Fig. 18. Challenges in relation to the requirements engineering for SCS (Q12.1).

Figure 18 shows the challenges mentioned by the practitioners in relation to the requirements engineering for safety-critical systems. Again, we see the importance of a better way to connect the safety standards to the requirements documents, as mentioned by practitioners from two companies.

4.13 Close-ended Questions

The last part of the questionnaire that we applied during the interviews is formed by a set of close-ended questions (Part C). 18 out of 19 practitioners answered the questionnaire Part C: 12 of them were "requirements suppliers" and 6 were "requirements clients". Only one practitioner did not send us the answers to the questionnaire Part C. One requirements supplier felt unable to answer the questions Q7 to Q16, and two requirements suppliers felt unable to answer the questions Q14 to Q16. We prepared 16 close-ended questions with the intention to complement the open-ended questions to better capture the view of the practitioners (see Appendix 2 - Part C). The close-ended questions were formatted following the style of Likert's scale, excepted the question 1 (Q1) which was formatted with four alternatives where the interviewee could choose more than one alternative. Figure 19 shows the result obtained from the Q1.

In Q1 we are interested to know what types of language are used by the practitioners to specify requirements [35]. In Figure 19 we can see that 50% of the practitioners use only informal language for requirements specification. By "informal language" we mean natural language, i.e.: requirements described only based on text. By "graphic models" we mean diagrams as UML/SysML and similar, as well as *ad hoc* diagrams. By "formal languages" we mean languages based on mathematical representation, such as logic based models and control engineering equations. We explained these terms to the practitioners before they answered the questionnaire.

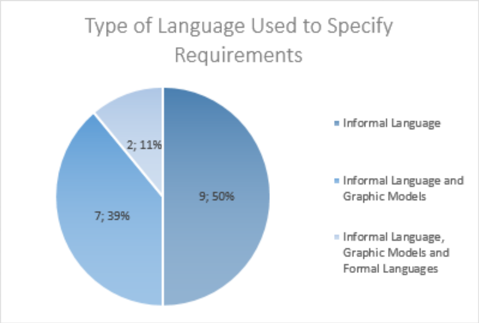

Fig. 19. Preference of the practitioners in relation to the type of language to specify requirements (Q1).

39% of the practitioners informed that they used some graphic models (as UML diagrams, block diagrams, and FTA diagrams) only to complement the requirements specification. Only 11% declared that they use some type of formal language to complement the requirements specification. The formal language mentioned is mathematical formula (equations used in the context of control engineering); none of the formal language/methods proposed by the software engineering community was mentioned by the practitioners.

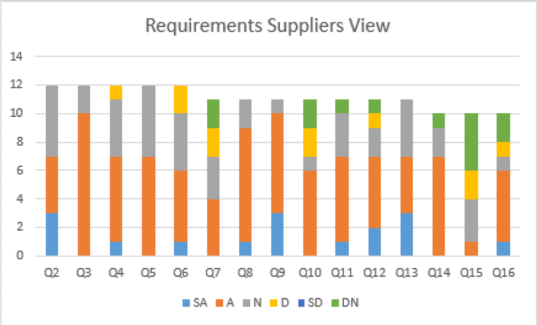

Fig. 20. Requirements suppliers view in relation to the close-ended questions.

Legend for Figures 20 and 21:
SA: Strongly Agree; A: Agree; N: Neutral; D: Disagree;
D: Strongly Disagree; DN: Do Not Know

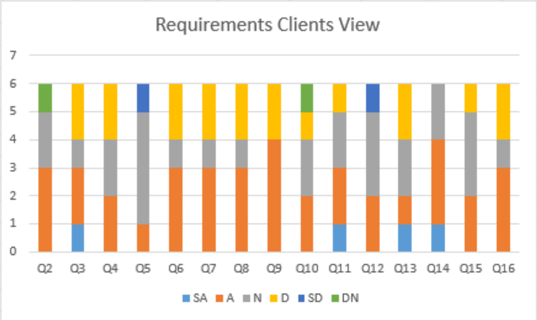

Fig. 21. Requirements clients view in relation to the close-ended questions.

As explained in Section 3.1 the practitioners were classified as requirements supplier (responsible to produce requirements) and requirements client (user of the requirements). In Figure 20 we show the results obtained from the Q2 to Q16 according to the requirements suppliers view. Figure 21 shows the results according to the requirements

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TSE.2018.2854716, IEEE Transactions on Software Engineering

clients view. A general result that we can observe contrasting Figures 20 and 21 is that the requirements suppliers believe that the requirements engineering approaches adopted by them are producing more benefits than is realized by the requirements clients, i.e.: the requirements clients view is more pessimistic than the requirements suppliers view in relation to how the requirements engineering processes are working in the companies.

The close-ended questions helped us to improve the answers of our RQs. Q2 to Q4 are related to the RQ1.1. In these questions we are interested to know if the approaches adopted to elicit, analyze & negotiate, specify & validate requirements are meeting the practitioners' needs. According to the requirements clients view the major problem is in relation to analysis and negotiation and specification and validation of requirements. 33% of the requirements clients think the approaches adopted to analyze & negotiate requirements are not meeting their necessities, the same percentage occurs for the approaches adopted to specify & validate requirements [29, 30].

Q5 and Q6 are related to the RQ1.2. In these questions we are interested to know if the requirements specifications are sufficient, clear and understandable [28]. In Q5 both requirements suppliers and requirements clients are more optimistic in relation to the requirements specification sufficiency. However, in relation to how clear and understandable the specifications are the requirements clients are more pessimistic (Q6): 33% of them don't think the requirements specifications are clear and understandable.

Q7 and Q8 are related to the RQ1.3. With these questions we want to know if requirements engineering and safety analysis are well integrated processes, as well as if the safety requirements are derived from the hazard analysis [15, 17]. Only 18% of requirements suppliers believe that the requirements engineering and safety analysis processes are not well integrated (Q7), whereas 33% of requirements clients agree that these processes are not well integrated. 82% of requirements supplier agree that the safety requirements are derived from the hazard analysis, on the other hand just 50% of requirements clients agree with this assertion (Q8). Again, the view from the requirements suppliers are more optimistic than requirements clients in relation to the processes adopted.

In Q9 we want to know if the safety engineers and system engineers are participating in the validation of the safety requirements. 82% of requirements suppliers answered that safety engineers and system engineers are participating in the validation, against 33% of requirements clients that answered the opposite.

From Q10 to Q13 we are interested to know if the requirement specifications produced by the practitioners make clear the safety needs to software engineers and testers, and also if the specifications evolve together the system and if they are traceable among the other system artefacts [36] (test cases, code, etc.). As to whether the requirement specifications make clear the safety needs to the software engineers (Q10), the suppliers and clients have a similar view: 18% of requirements suppliers think the requirement specification are not making clear the safety needs, whereas 17% of requirements clients have the same opinion. 55% of requirements suppliers believe that the requirement specifications they are producing make a clear understanding to produce test cases [37] (Q11); only 33% of requirements clients agree with them. 64% of requirements suppliers think the requirement specifications are following the system evolution (Q12), whereas just 33% of requirements clients think this way. 64% of requirements suppliers believe that the requirements specification can be traced throughout the project artefacts (Q13), only 33% of requirements clients think the same.

Finally, in Q14, Q15 and Q16 we focus on the relationship between requirement specifications and certification process. In Q14 we are interested to know if the requirement specifications are in compliance with the safety standards and regulations. Both requirements suppliers and requirements clients have practically the same view about this question: 70% of requirements suppliers agree that the requirement specifications they are producing are in compliance with the safety standards, 67% of requirements clients agree as well. Regarding the need to update and remake the requirement specifications to support the certification process (Q15), requirements suppliers and requirements clients have a different opinion: only 10% of requirements suppliers believe that it is necessary to remake the requirement specifications in order to properly support the certification process, whereas 33% of the requirements clients believe that is necessary to update or remake the requirement specifications to properly support the certification process. 60% of requirements suppliers agree that the requirement specifications produced by them are sufficient in order to support the safety arguments during the certification process (Q16), whereas 50% of requirements clients have the same opinion.

5 Conclusion

Based on the results presented in the previous sections, we summarize the identified gaps and present some recommendations.

Improving the elicitation (interviews). Interview was the most reported approach by the practitioners to collect requirements during the empirical study. However, the interviews have been carried out in a very informal way. Considering the importance of the interview during requirements gathering we recommend to the practitioners to create a protocol to better organize the interviews [5][6]. Such protocol should cover at least the following activities: planning, performing and validation. Companies should invest in training the practitioners to conduct structured interviews during requirements engineering process.

Improving the meetings. Practitioners have reported they use meetings for several reasons during SCS lifecycle, e.g.: to perform requirements negotiation, requirements validation, safety/hazard analysis, requirements communication and so on. In the same track of interviews, it seems that meetings have been carried out in an informal way. Considering the importance of the meetings during the whole SCS lifecycle, we recommend to the practitioners to create and follow protocols to get better results from the meetings [7][8]. Companies should invest in training the practitioners to prepare and conduct structured meetings during the whole SCS development lifecycle.

Adopting requirements management tools. Surprisingly, only 39% of the companies that participated in this interview study reported they use a proper requirements management software tool to support the requirements activities throughout SCS lifecycle. Even in the companies that reported they have a requirements management tool, it was common to see that the tools were underutilized by the practitioners. We strongly recommend the adoption of a proper software tool to support the whole requirements engineering process, as well as to improve the integration between safety and requirements teams. Interviews and meetings should be supported by proper software tools as well. Such tools should integrate the requirements elicitation, analysis and negotiation, requirements documentation and validation with the safety analysis and risk assessment (including the techniques such as FMEA and FTA). For instance, the tool called SystemWeaver [50] seems to be a possible choice to support these activities. SystemWeaver supports traceability and creation of safety cases as well. One advantage of the SystemWeaver in relation to the traditional requirements tool is a safety module offered by it to support the more specific activities performed during the requirements process for SCS, such as hazard analysis, safety analysis and risk assessment.

Using standards to specify requirements. The safety standards are very important in SCS industry. We verified that even companies that are not obligated to have a certification process usually they follow safety standards as a way to improve their practices. A common practice adopted by the practitioners in several companies that we have conducted the interviews is to "simplify" the requirements specification using links between the requirement specifications and safety standards documents. This is a way to reduce the effort to write requirements during the requirements specification and in somehow to reduce the ambiguity in the requirements. Although such practice seems to require less time and effort, the practitioners recognize that there are risks in such approach. Again, a proper software tool can help the practitioners in this practice, in a way that the requirements and the safety standards can be formally (hyper)linked, as well as they can be dynamically reused and updated. Moreover, we recommend that the safety officers review periodically the linkage of requirements documents and safety standards, as well as to make sure that the suppliers of the companies are doing the same with their requirement documents.

Integration between requirements and safety teams. Surprisingly, 45% of companies that participated in this study do not have a safety team supporting the SCS development. Only 27% of companies have a cross-function team with the participation of both the requirements engineers and safety engineers. In 27% of companies we found the participation of safety engineer in the requirements team. On the other hand we did not find any company where requirements engineer participates in safety teams. In order to reach a better integration between requirements and safety teams and to improve the communication of requirements and safety issues, companies should invest more in the formation of cross-function teams.

Requirements communication. Communicating requirements throughout the whole SCS lifecycle is still a challenge. 26% of practitioners declared that the informal conversation is the main way to communicate requirements between requirements suppliers and developers. Although the most part of practitioners reported that they formally document the system and software requirements (using natural language), some practitioners from relevant companies reported that the requirements documents are produced to be more in compliance with the standards & regulations rather than to be really used as requirements communication documents during the system/software development (as quoted in section 4.12).

Modelling and templates. The requirements specification and documentation are largely based on natural language: only 7 practitioners reported they use some modelling to complement the requirements specification but not in a systematic or institutionalized way. We believe that there are benefits in using modelling languages to support requirements specification in SCS. However, the benefits of use them need to be further researched. In relation to the use of templates, practitioners from six companies reported they adopt templates to help them in the requirements documentation. However, the templates adopted are defined by them in an *ad hoc* way, and no practitioners mentioned that they use templates based on the requirements engineering literature.

Different views between requirements suppliers and requirements clients. Regarding the benefits of the requirements engineering processes adopted in the SCS companies, requirements suppliers and requirements clients have a fairly different views. While the former have an optimistic view of how requirements are being managed throughout the projects, the latter do not feel that the requirements are being handled in the most appropriate way. This contrast between these views indicates that the requirements suppliers, especially requirements engineers and safety engineers, should revise and rethink their current practices, especially in relation to the requirement specification and documentation.

Certification process. Safety standards and regulations have a very important role in the SCS development and the system/product certification process, even for companies that are not obliged to certify their systems/products, as the companies in the defence domain. The practitioners are positive that safety standards and regulations help them to adopt good practices and consequently improve the safety of their systems, despite the ambiguity and subjectivity present in the safety standards. Regarding the impact of the certification process to the requirements engineering process, 36% of companies stated that it become more complex because the necessity to manage the requirements documents with the safety standards. This is a gap in which requirements engineering researchers should focus effort in order to develop approaches/tools that help practitioners in managing requirements documents with safety standards.

Challenges in relation to the Requirements Engineering. In 27% of companies the practitioners believe that the major challenge is to produce more useful requirements documents. Even at the companies where the practitioners didn't explicitly mention that, it was clear that the most part of the requirements documents produced by them are little used as a real reference for the developers. Again the difficulty to manage the requirements documents and safety standards & regulations was pointed out by the practitioners as a challenge, not only in relation to the certification process but along the SCS development. To spread out the "safety culture" among all SCS stakeholders remains a challenge, particularly among those that usually don't have a proper safety education during their graduation, as currently happens with requirements engineers and software engineers.

State-of-the-practice. We performed in-depth interviews with practitioners from 11 SCS companies and surprisingly we found a poor situation regarding requirements engineering practices. Considering these companies are focused on developing SCS we hoped to find a better situation in relation to the practices adopted by them to handle safety requirements, particularly, and systems/software requirements, in general. Despite the efforts of the Requirements Engineering community in developing new approaches to help the practitioners, what we saw along this study indicates that still there is an important gap between industry and academia. We found fragile and incipient practices/processes adopted by the practitioners, especially regarding to the requirements elicitation, requirements modelling and documentation, and the integration between requirements and safety analysis. The practitioners we interviewed, in most of cases, neglect or ignore the research of the Requirements Engineering community in relation to essential aspects in system/software requirements, such as the techniques and methods to elicit requirements, modelling languages to support requirements specification, templates to support requirements documentation, processes and software tools to manage requirements throughout SCS lifecycle, among others. Our recommendation is twofold: (a) Industry practitioners should follow the improvements and developments produced by the Requirements Engineering community, participating in related conferences, reading published papers, and participating in experimental projects together academia; (b) Academic researchers should work harder to engage industrial practitioners to make them aware of best practices and transition research results into industrial setting. Moreover, researchers should focus on the need to not only invent and describe new methods, models, processes and frameworks to enable industrial practitioners, but also to validate these in a real industrial setting so that they might be iteratively improved and validated towards usability, usefulness, and scalability. This will also enable technology and knowledge transfer, as well as increase trust in a solution. A model for technological transfer presented in [11, 53] is a good way to validate researches in industrial setting. The adoption of such model can help researchers and practitioners to be more engaged in cooperative projects.

We would like to thank the practitioners and companies that participated in this interview study. We hope the results presented in this paper can be useful for the companies to rethink their current practices, as well as to point out future improvements in their requirements and safety engineering processes. Moreover, we believe that the findings and analysis presented in this paper may indicate new research directions to the requirements engineering researchers.

ACKNOWLEDGMENT

This work was partially funded by the Federal University of São Paulo (UNIFESP), and the Brazilian research agency CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico - under grant agreement no. 249059/2013-04. The work is partially supported by a research grant for the ORION project (reference number 20140218) from The Knowledge Foundation in Sweden.

REFERENCES

[1] J. Hatcliff, A. Wassyng, T. Kelly, C. Comar, and P. Jones, "Certifiably safe software-dependent systems: challenges and directions", Proceedings of the on Future of Software Engineering - FOSE, pp. 182-200, 2014.

[2] L. E. G. Martins and T. Gorschek, "Requirements Engineering for Safety-Critical Systems: A Systematic Literature Review", Information and Software Technology, vol. 75, pp. 71-89 July 2016.

[3] N. G. Leveson, Safeware: System Safety and Computers. Addison-Wesley, 1995.

[4] N. G. Leveson, Engineering a Safer World: Systems Thinking Applied to Safety. The MIT Press, 2011.

[5] R. Opdenakker, "Advantages and Disadvantages of Four Interview Techniques in Qualitative Research", Forum: Qualitative Social Research, vol. 7, no. 4, art. 11, September 2006.

[6] A. Aurum and C. Wohlin, Engineering and Managing Software Requirements. Springer, 2005.

[7] R. Schwartz, "How to Design an Agenda for an Effective Meeting", Harvard Business Review, March 19, 2015.

[8] D. Rousmaniere, "What Everyone Needs to Know About Running Productive Meetings", Harvard Business Review, March 13, 2015.

[9] M. Q. Patton, Qualitative Research and Evaluation Methods. Sage Publications, 2002.

[10] R. B. Svensson, T. Gorschek, B. Regnell, R. Torkar, A. Shahrokni, and R. Feldt., "Quality Requirements in Industrial Practice - An Extended Interview Study at Eleven Companies", IEEE Trans. Software Engineering, vol. 38, no. 4, pp. 923-935, Jul/Aug 2012, doi: 10.1109/TSE.2011.47.

[11] M. Ivarsson and T. Gorschek, "Technology Transfer Decision Support in Requirements Engineering Research: A Systematic Review of REj," Requirements Eng. J., vol. 14, no. 3, pp. 155-175, July 2009.

[12] E. Turban, D. King, J.K. Lee, and D. Viehland, Electronic Commerce: A Managerial Approach. Prentice Hall, 2006.

[13] C. Robson, Real World Research. Blackwell, 2002.

[14] M. A. Sujan, I. Habli, T. P. Kelly, S. Pozzi, and C. W. Johnson, "Should healthcare providers do safety cases? Lessons from a cross-industry review of safety case practices", Safety Science, 84, pp. 181–189, 2016, http://doi.org/10.1016/j.ssci.2015.12.021.

[15] K. Beckers, C. Isabelle, T. Frese, D. Hatebur, and M. H. "Systematic Derivation of Functional Safety Requirements for Automotive Systems", Lecture Notes in Computer Science, vol. 8666, pp. 65–80, 2014.

[16] J. Wu, T. Yue, S. Ali, and H. Zhang. "Ensuring Safety of Avionics Software at the Architecture Design Level : An Industrial Case Study". In 13th International Conference on Quality Software, pp. 55–64, 2013. http://doi.org/10.1109/QSIC.2013.41

[17] A. Abdulkhaleq, S. Wagner, and N. Leveson. "A comprehensive safety engineering approach for software-intensive systems based on STPA", Procedia Engineering, 128, pp. 2–11, 2015. http://doi.org/10.1016/j.proeng.2015.11.498

[18] R. K. Panesar-Walawege, M. Sabetzadeh, L. Briand, and T Coq. "Characterizing the Chain of Evidence for Software Safety Cases: A Conceptual Model Based on the IEC 61508 Standard", In Third International Conference on Software Testing, Verification and Validation, pp. 335–344, 2010. http://doi.org/10.1109/ICST.2010.12

[19] C. Wohlin, P. Ruseson, M. Host, C. Ohlson, B. Regnell, and A. Wesslén, Experimentation in Software Engineering: An Introduction. Kluwer Academic, 2000.

[20] P. Braun, M. Broy, F. Houdek, M. Kirchmayr, M. Müller, B. Penzenstadler, K. Pohl, and T. Weyer. "Guiding requirements engineering for software-intensive embedded systems in the automotive industry: The REMsES approach", Computer Science Research and Development, vol. 29, no. 1, pp. 21–43, 2014. http://doi.org/10.1007/s00450-010-0136-y

[21] P. Runeson and M. Host. "Guidelines for conducting and reporting case study research in software engineering", Empirical Software Engineering, 14:131-164, April 2009.

[22] A. Ruiz, G. Juez, H. Espinosa, J. L. de la Vara, and X. Larrucea. "Reuse of safety certification artefacts across standards and domains: A systematic approach", Reliability Engineering and System Safety, vol. 158, pp. 153-171, 2017. http://doi.org/10.1016/j.ress.2016.08.017

[23] T. Gorschek, E. Tempero, and L. Angelis. "On the use of software design models in software development practice: an empirical investigation", The Journal of Systems & Software, vol. 95, pp. 176-193, 2014.

[24] A. Ferrari, P. Spoletini, and S. Gnesi. "Ambiguity Cues in Requirements Elicitation Interviews", In IEEE International Requirements Engineering Conference, pp. 56-65, 2016. DOI: 10.1109/RE.2016.25

[25] G. Popov and B. K Lyon, Risk Assessment: A Practical Guide to Assessing Operational Risks. Wiley, 2016.

[26] R. J. Mikulak, R. Mcdermott, and M. Beauregard, The Basics of FMEA. 2nd Edition, CRC Press, 2008.

[27] U.S. Nuclear Regulatory Commission, Fault Tree Handbook. 2014.

[28] H. Soares and R. S. Moura. "A Methodology to Guide Writing Software Requirements Specification Document", Latin American Computing Conference (CLEI), pp 1-11, 2015. http://DOI:10.1109/CLEI.2015.7360001

[29] M. Komssi, M. Kauppinen, m. Pyhajarvi, J. Talvio, and T. Mannisto. "Persuading Software Development Teams to Document Inspections: Success Factors and Challenges in Practice", In IEEE International Requirements Engineering Conference, pp. 283-288, 2010. DOI: 10.1109/RE.2010.40

[30] J. Zhou, Y. Lu, K. Lundqvist, H. Lonn, D. Karlsson, and B. Liwang. "Towards Feature-Oriented Requirements Validation for Automotive Systems", In IEEE International Requirements Engineering Conference, pp. 428-436, 2014. DOI: 10.1109/RE.2014.6912294

[31] S. Fricker, T. Gorschek, and M. Glinz. "Goal-Oriented Requirements Communication in New Product Development", In Second International Workshop on Software Product Management, pp. 27-34, 2008. DOI: 10.1109/IWSPM.2008.2

[32] DoD USA, Military Standard Mil-Std-882C, System Safety Program Requirements, 1993.

[33] A. Ferrari, P. Spoletini, and S Gnesi. "Ambiguity as a resource to disclose tacit Practitioners from three companies mentioned that in somehow it is necessary to produce ", in IEEE International Requirements Engineering Conference, pp. 26-35, 2015. DOI: 10.1109/RE.2015.7320405

[34] F. Chen, N Power, and J. J. Collins. "A Stakeholder Contribution Pattern in Requirements Decision-Making: An Empirical Study in Enterprise Development", in IEEE International Requirements Engineering Conference Workshops (REW), pp. 289-295, 2016. DOI: 10.1109/REW.2016.054

[35] D. Ott. "Defects in natural language requirement specifications at Mercedes-Benz: An investigation using a combination of legacy data and expert opinion", in IEEE International Requirements Engineering Conference, pp. 291-296, 2012. DOI: 10.1109/RE.2012.6345817

[36] A. M. D. Duarte, D Duarte, and M. Thiry. "TraceBoK: Toward a Software Requirements Traceability Body of Knowledge", in IEEE International Requirements Engineering Conference, pp. 236-245, 2016. DOI: 10.1109/RE.2016.32

[37] S. Feldmann, S. Rosch, C. Legat, and B. Vogel-Heuser. "Keeping requirements and test cases consistent: Toward an ontology-based approach", in IEEE International Conference on Industrial Informatics, pp. 726-732, 2014. DOI: 10.1109/INDIN.2014.6945603

[38] J. Beatty. "Winning the hidden battle: Requirements tool selection and adoption", in IEEE International Requirements Engineering Conference, pp. 364-365, 2013. DOI: 10.1109/RE.2013.6636753

[39] M. Unterkalmsteiner. Coordinating Requirements Engineering and Software Testing. Doctoral Dissertation in Software Engineering. Blekinge Institute of Technology Doctoral Dissertation Series no. 2015:08, 2015.

[40] C. Heitmeyer. "Managing Complexity in Software Development with Formally Based Tools". *Electronic Notes in Theoretical Computer Science*, *108*, pp. 11–19, 2004. doi:10.1016/j.entcs.2004.11

[41] A. Hall. "Seven Myths of Formal Methods". *IEEE Software Magazine*, *7*(5), pp. 11–19, 1990.

[42] J. Abrial. "Formal Methods in Industry : Achievements, Problems, Future". In *Proceedings of the IEEE International Conference on Software Engineerin*, pp. 761–767, 2006.

[43] Z. Chen. "Formalizing Safety Requirements Using Controlling Automata". In Proceedings of the Second International Conference on Dependability, pp. 81–86, 2009. doi:10.1109/DEPEND.2009.18

[44] C. Robson, Real World Research. Blackwell, 2002.

[45] J. E. Hutchinson, J. Whittle, and M. Rouncefield. "Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure", Science of Computer Programming, vol. 89, part B., pp. 144-161, September 2014.

[46] X. Zheng, C. Julien, M. Kim, and S. Khurshid. "Perceptions on the State of the Art in Verification and Validation in Cyber-Physical Systems", IEEE Systems Journal, vol. 11, issue 4, pp. 2614-2627, December 2017.

[47] E. Sadraei, A. Aurum, G. Beydoun, and B. Paech. "A field study of the requirements engineering practice in Australian software industry", Requirements Engineering Journal, vol. 12, issue 3, pp. 145-162, July 2007.

[48] D. M. Fernàndez and S. Wagner. "Naming the pain in requirements engineering: A design for a global family of surveys and first results from Germany", Information and Software Technology, vol. 57, pp. 616-643, January 2015.

[49] L. Liu, T. Li, and F. Peng. "Why Requirements Engineering Fails: A Survey Report from China", in 18th IEEE International Requirements Engineering Conference, pp. 317-322, 2010.

[50] R. Wohlrab and A. Shahrokni. "SystemWeaver: Facilitating Configurable and Scalable Traceability of Systems Engineering Artifacts", in IEEE/ACM 40th International Conference on Software Engineering, Poster Exibition, 2018.

[51] A. K. Thurimella and D. Janzen. "Metadoc Feature Modeler: A Plug-in for IBM Rational DOORS", in 15th International Software Product Line Conference, pp. 313-322, 2011.

[52] Structures Software Systems Ltd. "Role and Representation of System Requirements in Systems Engineering Using Cradle", white paper, pp. 1-17, 2017.

[53] T. Gorschek, P. Garre, L. Larsson, and C. Wohlin. "A Model for Technology Transfer in Practice", IEEE Software 23(6), pp. 88-95, 2006.

Luiz Martins received the PhD degree from State University of Campinas (UNICAMP). He is a professor of software engineering and embedded systems at Federal University of São Paulo (UNIFESP). His research interests include requirements engineering, embedded systems development, safety-critical systems, model-driven software development, and technological innovation in medical systems. He has published more than 25 papers in these areas. Contact him at legmartins@unifesp.br.

Tony Gorschek is a professor of software engineering at Blekinge Institute of Technology, Sweden. He has 10y experience working with SW intensive product development in domains ranging from automotive to telecom, working as CTO, chief architect and developer. His research interests include requirements engineering, technology and product management, lean product development, quality assurance, and innovation. Gorschek has a PhD in software engineering from BTH. He's a member of the IEEE and the ACM. Contact him at tony.gorschek@bth.se or visit www.gorschek.com.