Vara, J. L., Jiménez, G., Mendieta, R. y Parra, E. (2019). Assessment of the Quality of Safety Cases: A Research Preview. In: Knauss E., Goedicke M. (eds) *Requirements Engineering: Foundation for Software Quality. REFSQ 2019.* Lecture Notes in Computer Science, 11412, pp. 124-131. Springer, Cham.

DOI: https://doi.org/10.1007/978-3-030-15538-4_9

Assessment of the Quality of Safety Cases: A Research Preview

Jose Luis de la Vara[1], Gabriel Jiménez[1], Roy Mendieta[2], and Eugenio Parra[1]

[1] Departamento de Informática, Universidad Carlos III de Madrid, Leganes, Spain
jvara@inf.uc3m.es, {gabriel.jimenez, eparra}@kr.inf.uc3m.es

[2] The REUSE Company, Leganes, Spain
roy.mendieta@reusecompany.com

Abstract. [Context and motivation] Safety-critical systems in application domains such as aerospace, automotive, healthcare, and railway are subject to assurance processes to provide confidence that the systems do not pose undue risks to people, property, or the environment. The development of safety cases is usually part of these processes to justify that a system satisfies its safety requirements and thus is dependable. **[Question/problem]** Although safety cases have been used in industry for over two decades, their management still requires improvement. Important weaknesses have been identified and means to assess the quality of safety cases are limited. **[Principal ideas/results]** This paper presents a research preview on the assessment of the quality of safety cases. We explain how the area should develop and present our preliminary work towards enabling the assessment with Verification Studio, an industrial tool for system artefact quality analysis. **[Contribution]** The insights provided allow researchers and practitioners to gain an understanding of why safety case quality requires further investigation, what aspects must be considered, and how quality assessment could be performed in practice.

Keywords: safety case, quality, quality assessment, system assurance, safety-critical system, Verification Studio.

1 Introduction

Safety-critical systems are those whose failure can harm people, property, or the environment [17], e.g. systems in aerospace, automotive, healthcare, and railway. These systems are subject to rigorous, systematic, and planned assurance processes to provide confidence that the systems satisfy given requirements. These requirements can be system requirements (i.e. about the properties of a system, including safety requirements) or be indicated in standards with which a system must comply. Among the artefacts to manage for systems assurance, safety cases are arguably the main ones.

A safety case is a structured argument, supported by a body of evidence, that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given environment [16]. Safety cases have been used in industry for over two decades, first in application domains such as defence and energy and more

recently in domains such as automotive and healthcare. Many researchers have worked on the specification and management of structured safety cases [17], e.g. with GSN (Goal Structuring Notation). The notion of safety case has also evolved towards the more general concept of assurance case, to justify system dependability, and other specific cases such as security case. Although the term safety case is not used in some applications domains and standards, the concept of artefact to justify system safety and dependability exists in all safety-critical contexts.

Despite the importance and wide use of safety cases, certain aspects of their development require improvement to ensure that the quality of a safety case is sufficient and thus system safety has been acceptably justified. Among the authors that have studied safety case quality, Nancy Leveson is one of the most well-known experts that has doubted the quality and effectiveness of safety cases. For example, she argues that confirmation bias can easily appear in a safety case and has reviewed issues in past safety cases such as obscure language and compliance-only exercises [12]. Greenwell et al. [8] found several types of fallacies in the arguments of existing safety cases, e.g. using wrong reasons, drawing wrong conclusions, and omission of key evidence.

Even researchers and practitioners that strongly support the use of safety cases have acknowledged the risk of developing low-quality safety cases. Tim Kelly [10] has referred to issues such as the "apologetic safety case", the document-centric view, the approximation to the truth, the prescriptive safety case, and the illusion of pictures, and Bloomfield and Bishop [3] argue that improvements are needed in safety case structure and confidence. In a seminal paper on software safety certification [9], Hatcliff et al. refer to the weakness that there are many possible forms of an assurance case, some good and some bad, and to the lack of guidance to produce effective assurance cases, among other issues. Langari and Maibaum [11] review challenges for safety cases, including size and complexity, readability, checking soundness, and checking completeness, and Wassyng et al. [22] discuss weaknesses about argumentation.

Recent studies about the state of the practice [5, 18] report that practitioners face challenges to effectively create and structure safety cases, that tool support for safety cases is basic, and that safety case evolution does not seem to be properly addressed. How safety case quality is managed, including its evolution, can be improved.

In summary, and as further discussed below, the current practices and tools to ensure and assess the quality of safety cases seem to be insufficient and further research is needed. We are working towards filling the gaps in the state of the art, and in this paper we present a research preview about (1) the main needs to take into account for effective assessment of the quality of safety cases in practice, and (2) our current results on the development of a solution to assess safety case quality with Verification Studio [21], an industrial tool for system artefact quality analysis. We have been able to successfully use Verification Studio to analyse the quality of safety cases specified with ASCE (Assurance and Safety Case Environment) [1]. The quality analysis is partial and several important aspects have not been addressed yet, but the results represent a promising initial step towards the assessment of the quality of safety cases in industry.

This paper distinguishes from prior work by focusing on how the quality of safety cases should be assessed and proposing a solution linked to quality analysis in practice. The insights provided can help industry and academia gain a better understanding of what factors can influence safety case quality, why the topic requires further research, what aspects should be considered, and how quality assessment could be performed.

The rest of the paper is organised as follows. Section 2 introduces the main needs for assessing the quality of safety cases. Section 3 presents our current results and Section 4 our next steps. Finally, Section 5 summarises our conclusions.

2 Needs for Assessing the Quality of Safety Cases

This section presents the six main needs that, in our opinion, must be addressed to enable the effective assessment of the quality of safety case in practice.

1) The information about safety case quality is scattered. There exists guidance about the quality properties that a safety case should have; e.g. the GSN standard [7] presents errors to avoid. However, this information is in many different sources [20]: standards, research literature, tool documentation… It is necessary to create a unifying framework for safety case quality and that the framework gathers information from different sources, harmonising the guidance from different application domains.

2) Quality metrics for safety cases are limited. As a follow-up need, it is not clear how safety case quality could be objectively and suitably measured. Some metrics can be found in the literature, e.g. the number of unsupported claims, but the metrics (1) have not been developed in the scope of a sound quality framework and (2) usually deal with simple attributes. Most of the tool support for measurement of safety case quality further corresponds to research prototypes [14]. More mature tools, e.g. AdvoCATE [6], provide very limited and narrow sets of metrics. In addition, most metrics defined for safety-related assessments (e.g. [4]) do not apply to the specific quality needs of safety cases, but the metrics should be adapted or re-defined. Once the framework from the previous need is developed, metrics and measurement procedures must be defined and implemented to be able to quantitatively asses the quality of safety cases.

3) Safety case quality goes beyond safety case structure and syntax. Most work on safety case quality has focused on structural and syntactical aspects [20], e.g. the language used to specify a claim or how to assess the confidence in a claim. However, safety case quality is also based on e.g. the semantics of the elements and how well the argumentation is formed. These aspects indeed relate to some of the main criticisms that safety cases have received. It is necessary to pay further attention to them.

4) Safety cases are most often managed as textual documents. This is arguably the need that has been most widely disregarded by the research community. Prior work has focused on analysing graphical structured safety cases [17], but the reality in industry is that safety cases are most often managed as textual documents. These documents might include graphical arguments created with e.g. GSN, but the diagrams would correspond to only a part of the safety case document. It is necessary to think of how the textual descriptions could be analysed to assess the quality.

5) Safety case quality depends on the quality of many other system artefacts. Safety cases relate to other artefact types [5], e.g. safety analysis results and V&V results. Hundreds of references to other artefacts can be found in the safety case of a complex system, and the quality of the safety case depends on these artefacts. The relationship with other artefacts and their influence must be characterised from a quality perspective, also considering that the influence might vary among artefact types.

6) Safety case quality evolves. It is strongly recommended that safety cases are created incrementally [10], evolving from a preliminary version at e.g. system analysis

phase to an interim version during implementation and an operational one when system development finishes. A safety case should also be maintained during system operation and can be impacted by changes in other artefacts [5]. It is necessary that the approaches to assess the quality of safety cases consider that a safety case evolves during a system's lifecycle and that what the quality of a safety case is can vary between different phases.

3 Current Results

We have already started to work to enable our vision for the assessment of the quality of safety cases. We have first dealt with technological aspects, setting the scope of how a tool-based solution could effectively support the assessment of safety case quality in practice. We have performed little work on the quality framework and the quality metrics related to the first two needs presented above. This requires a deep investigation, including systematic reviews of the literature that take both academic publications and other sources such as safety standards into account.

Fig. 1 presents an overview of our current solution. It is based on the integration of two commercial tools: ASCE [1] and Verification Studio [21]. ASCE supports the specification of structured safety cases with e.g. the GSN notation. It is arguably the main tool in industry for this purpose [5, 18]. Verification Studio supports the analysis of the quality of different system artefact types and in different formats, such as textual requirements, logical system models with UML or SysML, or physical system models with Modelica or Simulink. The analysis is based on metrics for which measurement procedures are specified and for which quality levels are defined, i.e. the quality will be assessed as high or low depending on a metric's measurement result and thresholds. The quality is analysed according to the information in a System Knowledge Repository, which is a domain representation with an ontology.

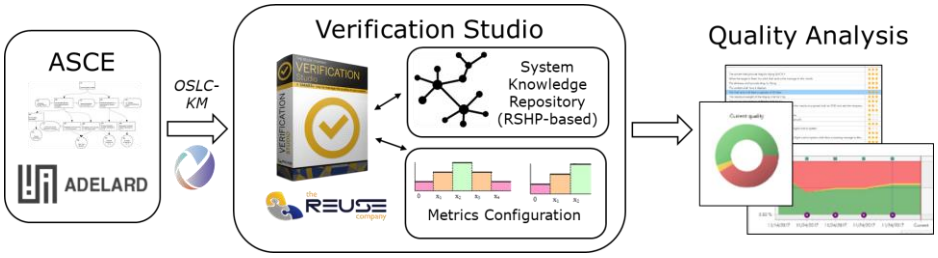

Fig. 1. Solution overview

The use of Verification Studio is suitable because it fits the needs presented above:

- Verification Studio provides default metrics to analyse artefact quality, mainly according to an ontology. The users can also define their own metrics and specify measurement procedures (need 2).
- Verification Studio supports semantics-based analyses of artefact quality, as well as analyses based on syntactical aspects and on artefact structure (need 3).
- The RSHP language [13] is used as the main basis for artefact representation in Verification Studio. It supports universal information representation via the different elements of an artefact, their relationships, and their semantics.

Artefacts in different formats (text, models, etc.) can be represented with RSHP, including safety cases specified as diagrams or as documents (need 4).

- Verification Studio supports the centralised analysis and management of the quality of different artefact types, and it is part of tool suite that also supports the management of the traceability between system artefacts (need 5).
- A recent feature of Verification Studio supports the analysis of the evolution of the quality of an artefact [19], including the use of different metrics at different moments of the lifecycle of an artefact to assess its quality (need 6).

For integration of ASCE and Verification Studio, we exploit the OSLC-KM technology [2], which provides generic means for tool interoperability. The technology allows us to transform ASCE files into data that Verification Studio can manage, i.e. data in the RSHP format. We have performed similar RSHP-targeted integrations in the past (e.g. for SysML [15]).

Once the information about an ASCE diagram (claims, arguments, evidence, etc.) has been imported into Verification Studio, we can analyse the quality of the safety case. To show that this is a feasible approach, we have first analysed the quality of structured safety cases available in the literature (e.g. [10]) with a set of default metrics that Verification Studio provides to evaluate artefact correctness. The metrics selected consider the precision, concision, non-ambiguity, singularity, completeness, quantifiers, and quantification in the text of an element. For instance, the number of vague adverbs and adjectives, the use of "and/or", the presence of domain terms, the text length, and the possible subjectivity of the sentences are considered for quality assessment. We have used a default ontology with English terms but a specialised one could have been employed, i.e. with case-specific concepts. Further details about how the quality analyses have been performed are not provided due to page limitations.

Fig. 2 presents a summary of the quality analysis results for a specific safety case. The report includes a quantitative score of the individual elements of the safety case (e.g. claims) and a qualitative evaluation with stars to show whether the quality is low, medium, or high. A pie chart shows an overview.

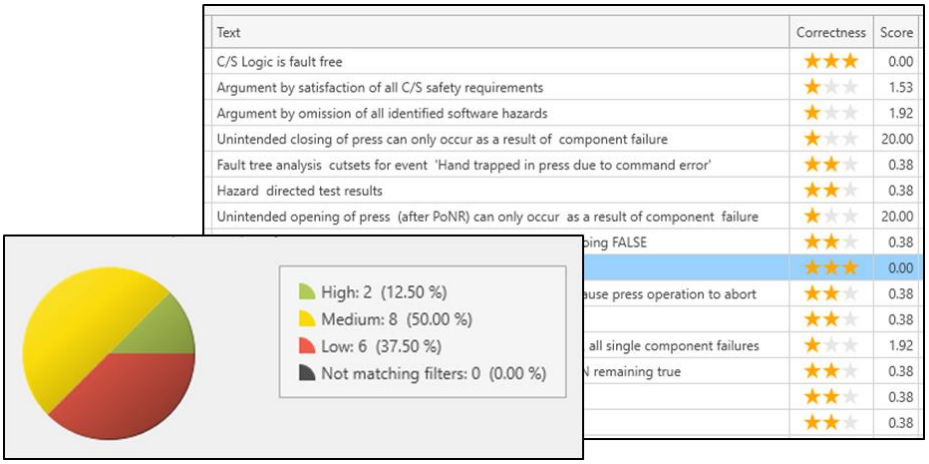

Fig. 2. Example of quality analysis results summary

4 Next Steps

In the previous sections we have presented the needs that we envision for effective assessment of the quality of safety cases and the results that we have obtained so far. In this section we present our next steps to realise our vision. Five main steps can be distinguished to complete the underlying research process.

1) Review of the current guidance for safety case quality. The goal of this step is to gather information about the practices that are used or should be used to ensure safety case quality. Different sources will be used, namely research literature, safety standards, and practitioners. For the latter, surveys and case studies could be conducted.

2) Specification of a quality framework for safety cases. This step aims at providing a framework based on which safety case quality can be assessed. The framework, which will address all the needs introduced in Section 2, will aggregate and synthesise the information collected in the previous step and will consist of different properties that could be analysed, metrics to characterise the properties, and measurement procedures for the metrics.

3) Validation of the framework. This step will confirm that the framework is suitable by comparing it against industrial practices. For example, a wide range of practitioners could be asked about the framework to identify possible missing aspects.

4) Implementation of the framework. This step refers to the enactment of the validated quality framework via tool support. The tool could correspond to a tailored usage of Verification Studio, but since the quality framework will be generic and tool-independent, it could be implemented with other tools (e.g. AdvoCATE extension).

5) Validation of the implementation of the framework. The last step will evaluate whether the framework and its implementation effectively assess safety case quality. In addition to using past safety cases, we will try to perform the validation in running projects. The safety cases will be both structured ones and documents, and we will use publicly available safety cases and safety cases provided by our industry network.

5 Conclusion

Safety cases must be managed during the lifecycle of many safety-critical systems and the quality of the safety cases must be ensured. However, weaknesses have been identified in the current practices for safety case development, affecting safety case quality and in turn the confidence in the dependability of the corresponding systems.

This paper has presented a research preview on how to address the assessment of the quality of safety cases. This includes dealing with needs such as that the information about safety case quality is scattered, quality metrics for safety cases are limited, quality goes beyond safety case structure, safety cases are most often managed as textual documents, safety case quality depends on the quality of many other system artefacts, and safety case quality evolves. If these needs are not fulfilled, it is difficult that the quality of safety cases can be effectively assessed in practice.

As a first step to meet the needs, we have developed a preliminary solution to link safety case specification and system artefact quality analysis. It integrates ASCE (Assurance and Safety Case Environment) and Verification Studio. The solution has allowed us to assess the quality of safety cases with a set of default metrics that

Verification Studio provides and to show that the further development with Verification Studio of means for assessment of safety case quality can be a feasible approach.

We will work on meeting the needs discussed and on tool support in the future, taking the next steps presented.

Acknowledgments. The research leading to this paper has received funding from the AMASS project (H2020-ECSEL ID 692474; Spain's MINECO ref. PCIN-2015-262). We also thank REFSQ reviewers for their valuable comments to improve the paper.

References

1. Adelard: ASCE Software (online) https://www.adelard.com/asce/ (Accessed Sep 26, 2018)
2. Alvarez-Rodriguez, J.M., et al.: Enabling system artefact exchange and selection through a Linked Data layer. J. Univers. Comput. Sci. 24(11), 1536-1560 (2018)
3. Bloomfield, R., Bishop, P.: Safety and assurance cases: Past, present and possible future - an Adelard perspective. SCSS 2010
4. Cruickshank, K.C., et al.: A Validation Metrics Framework for Safety-Critical Software-Intensive Systems. SoSE 2009
5. de la Vara, J.L., et al.: An Industrial Survey on Safety Evidence Change Impact Analysis Practice. IEEE T. Softw. Eng. 42(12), 1095-1117 (2016)
6. Denney, E., Pai, G.: Tool support for assurance case development. Autom. Soft. Eng. 25, 435-499 (2018)
7. Goal Structuring Notation: GSN Community Standard Version 1 (2011)
8. Greenwell, W.S., et al.: A taxonomy of fallacies in system safety arguments. ISSC 2006
9. Hatcliff, J., et al.: Certifiably Safe Software-Dependent Systems. FOSE 2014
10. Kelly, T.: Safety Cases. In: Handbook of Safety Principles. John Wiley & Sons (2018)
11. Langari, Z., Maibaum, T.: Safety Cases: A Review of Challenges. ASSURE 2013
12. Leveson, N.: The Use of Safety Cases in Certification and Regulation. MIT (2011)
13. Llorens, J., et al.: RSHP: an information representation model based on relationships. In: Soft Computing in Software Engineering. Springer (2004)
14. Maksimov, M., et al.: Two Decades of Assurance Case Tools: A Survey. ASSURE 2018
15. Mendieta, R., et al.: Towards Effective SysML Model Reuse. MODELSWARD 2017
16. MoD: Defence Standard 00-56 Issue 4 (2007)
17. Nair, S., et al.: An extended systematic literature review on provision of evidence for safety certification. Inform. Softw. Tech. 56(7), 689-717 (2014)
18. Nair, S., et al.: Evidence management for compliance of critical systems with safety standards: A survey on the state of practice. Inform. Softw. Tech. 60, 1-15 (2015)
19. Parra, E., et al.: Analysis of requirements quality evolution. ICSE 2018
20. Rinehart, D.J., et al.: Current Practices in Constructing and Evaluating Assurance Cases With Applications to Aviation. NASA (2015)
21. The REUSE Company: Verification Studio (online) https://www.reusecompany.com/verification-studio (Accessed Sep 26, 2018)
22. Wassyng, A., et al.: Software Certification: Is There a Case against Safety Cases? Monterey Workshops 2010