This is a post-peer-review, pre-copyedit version of DepCoS-RELCOMEX 2019 conference paper published in " Engineering in Dependability of Computer Systems and Networks", vol. 987 of the series Advances in Intelligent Systems and Computing, pp. 245-255, Springer International Publishing, the definitive authenticated version is available at: https://doi.org/10.1007/978-3-030-19501-4_24

Representing Process Characteristics to Increase Confidence in Assurance Case Arguments

Aleksander Jarzębowicz [0000-0003-3181-4210] and Szymon Markiewicz

Department of Software Engineering,
Faculty of Electronics, Telecommunications and Informatics,
Gdańsk University of Technology, Narutowicza 11/12, 80-233 Gdańsk, Poland
olek@eti.pg.edu.pl

Abstract. An assurance case is a structured, evidence-based argument demonstrating that a safety or other quality objective of a high integrity system is assured. Assurance cases are required or recommended in many industry domains as a means to convince the regulatory bodies to allow commissioning of such system. To be convincing, an argument should address all potential doubts and thus cover numerous additional issues, including the processes that led to development of the considered system. It is however not obvious, which elements of processes (and which characteristics of them) should be documented and how to include them in the argument without making it too large and complex. In this paper we provide description structures for essential process elements. The structures were developed on the basis of literature search and reviews of publicly available assurance cases. We also show how to include such information within the overall assurance case in a way that reduces the complexity and allows to distinguish process-related elements from the primary argument.

Keywords: Assurance Case, Safety Case, Confidence Argument, Defeater.

1 Introduction

Assurance cases are developed for systems considered as safety-critical or expected to demonstrate other high integrity attribute (e.g. security, reliability). An assurance case is defined as "A reasoned and compelling argument, supported by a body of evidence, that a system (...) will operate as intended for a defined application in a defined environment" [1].

Assurance case is a structured, tree-like argument which starts with high-level claims about considered system's attribute(s) like safety or security. Then a supporting argument for each such claim is provided. The supporting argument will include more detailed, lower-level claims, which in turn need to be supported. This process

continues iteratively, until claims need no further decomposition, but can be addressed by providing facts and evidence demonstrating their validity. A simplified example depicting the frequently used schema of arguing safety by addressing particular hazards (situations that can potentially lead to an accident) is shown in Fig. 1. The example does not conform to any particular assurance case notation, to avoid the need to introduce it here.

Assurance cases have been adopted in several industry domains, which require development of an assurance case before the system can be commissioned and used (e.g. railway [2], flight control [3], automotive [4]). Also, a cross-domain ISO/IEC standard dedicated to assurance cases was published [5]. Moreover, for several other standards or guidelines, which do not explicitly encourage nor mention using an assurance case, it was shown that development of such argument can help to demonstrate conformance to standard's requirements – examples include ISO 61508 [6][7], ISO 15408 [8] or safety and quality management of healthcare services [9][10].

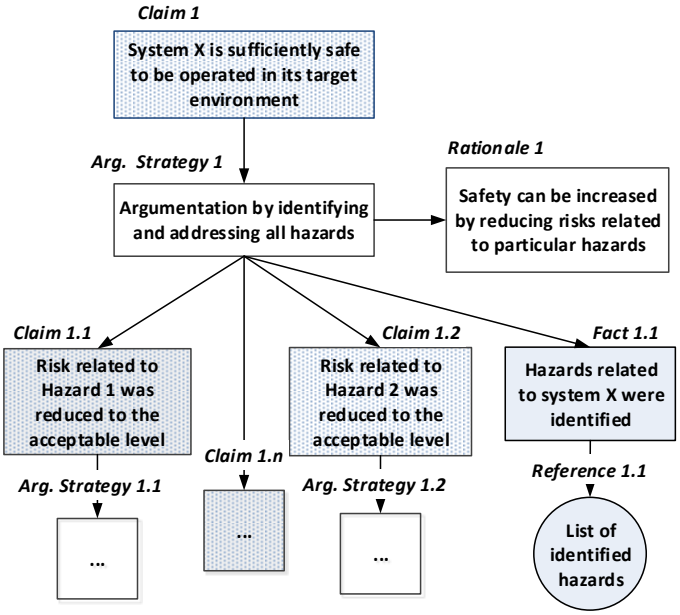

Fig. 1. A simplified assurance case argument example.

Despite the presence of several graphical notations dedicated to assurance case representation, all of them are based on the underlying argument model by Toulmin [11]. This model defines how to express so-called defeasible reasoning, which, in contrary to deductive reasoning (known from e.g. formal logic), cannot be proved with absolute certainty. Defeasible arguments are probably most common in real-world discussions (including e.g. law or politics). Assurance cases are defeasible arguments as well, perhaps with some exceptions when formal proofs are used to e.g. demonstrate consistency between referenced models. The confidence in a defeasible argument can be decreased by pointing out its weaknesses like questionable reasoning or insuffi-

ciently reliable evidence. Such weaknesses are represented in the original Toulmin's model as "rebuttals" [11], also alternative terms "deficit" [12] and "defeater" [13] are used in the literature dedicated to assurance cases. In this paper we will henceforth use the term "defeater".

In our example from Fig. 1, one could easily identify defeaters, both related to the reasoning (Argumentation Strategy 1 + Rationale 1) and the evidence (Fact 1.1 + Reference 1.1) e.g.:

- Reasoning – Is handling each hazard separately sufficient? What about the possibility of two or more hazards occurring simultaneously?
- Evidence – What confidence can we have that the list of identified hazards is complete? What hazard identification methods were used and why should they be considered adequate? Were the people responsible for hazard identification task sufficiently qualified and experienced?

Both kinds of defeaters are important, but in this paper we will focus on the second kind (i.e. defeaters related to evidence) only.

To address potential defeaters, additional elements are added to the assurance case. In our example, attempts to strengthen the argument with respect to the evidence could be made by providing claims (together with their supporting arguments) and facts documenting the process characteristics of hazard identification, including its participants, activities, methods and tools used. Another reason for inclusion of such elements in an assurance case is the fact that some standards and regulations demand process-related evidence concerning activities, artefacts, roles (part 1, p.35 of [5], p.20 of [3]).

Including such additional process-related elements in the assurance case has its consequences. The positive result is that it would (hopefully) lead to the increase of argument's confidence and cover requirements of some standards. There are however negative consequences as well: the assurance case becomes larger, more complex and harder to understand, as some parts argue that the main objective like system safety is achieved (assurance argument), while others focus on dealing with defeaters and increasing confidence (confidence arguments) [12]. Similarly, the set of evidence grows and includes two kinds of evidence items: those referred to by the main assurance argument (direct evidence) and those used to show that some other evidence is reliable (backing evidence) [3].

Despite such drawbacks, it is still necessary to include confidence arguments and process-based backing evidence in order to make assurance case a "compelling argument", as its definition states. It however raises the following questions:

1. Which process elements should be considered as evidence items and what characteristics should be specified for each of them?
2. How to structure the assurance case to include confidence arguments and process-based evidence but to distinguish them from the main assurance argument?

In this paper we attempt to provide answers to these questions. Hence, its main contributions are:

1. A tabular description structure specifying essential process elements and their characteristics, that are expected to be required to include in an assurance case. It can be used as a checklist when developing an assurance case and documenting evidence. It is not the first such proposal published, but it is more comprehensive, as it is based on the previous proposals as well as on assurance case reviews.
2. A distinction between generic and context-specific attributes of process elements. The former are closely associated with the evidence item, while the latter should be provided within a context of a particular confidence argument.
3. A way of representing (1) and (2) in the assurance case argument structure.

The remainder of the paper is structured as follows. In Section 2 we outline the related work. Section 3 presents our proposal of process elements to be used in assurance cases. Section 4 explains how to represent process-based evidence in confidence arguments. Finally, the paper is concluded in Section 5.

2 Related Work

The main areas of related work include: (1) representing defeaters and confidence arguments in assurance cases; and (2) defining process elements to be included in assurance case argument.

2.1 Defeaters and Confidence Arguments

As mentioned, defeaters (under a different name) were already included in the original Toulmin's model [11]. The idea and categorization of defeaters was further researched both for general arguments [14][15] and specifically for assurance case arguments [16][17], also a concept of additional confidence argument was introduced to cope with them.

Hawkins et al. [12] developed a proposal of dividing an assurance case into two separated but interrelated parts: safety argument (which can be generalized into assurance argument) and confidence argument. The resulting confidence argument is a large structure gathering arguments addressing various unrelated defeaters. Goodenough et al. [13] introduced so called confidence maps, which extends the known notation with additional diagram to represent defeaters and confidence arguments addressing them. Ayoub et al. [18] describe the approach which includes identifying potential defeaters and developing a separate ("contrapositive") confidence argument. Jarzębowicz and Wardziński [19] proposed that a given step of the main assurance argument should be associated with a corresponding "local" confidence argument. Such confidence argument can be attached to the rationale/justification element that explains the overall validity of the reasoning used in such step.

We consider our work as the further step in the research on representing defeaters and confidence arguments, based on two observations: (1) confidence arguments should be compact and focused on addressing particular defeaters, instead of being enumeration of all good practices used; (2) the existing, Toulmin-based notations are

sufficient to represent confidence aspects without the need to introduce additional elements or diagrams.

2.2 Process Elements in Assurance Cases

Graydon et al. [20] analyzed a software assurance argument and pointed out that software dependability can be compromised by several process-based causes including fallible humans, immature tools and unsuitable techniques. Ayoub et al. [18] proposed a list of process elements used in assurance cases and a pattern of a confidence argument addressing such elements. An alternative list of process-based elements and their characteristics (contributing to trustworthiness and appropriateness) as well as associated confidence argument pattern was given by Nair et al. [21]. Hawkins et al. [22] went one step further, by using process models (documented in software tools) to automate instantiation of confidence argument patterns.

The lists of process elements and their characteristics given by particular sources differ. As several recent papers express the need for better representations and ways to include confidence aspects of process-based elements [23][24] and for evidence [25], we believe that it is useful to aggregate the existing body knowledge and to create a comprehensive list on such basis.

3 Process Elements and Their Characteristics

Our aim was to identify process elements and their characteristics that are relevant to increase confidence in an assurance case argument. We intended to do it in a thorough manner, so we used diversified sources and divided this task into several steps:

1. Studying related papers that included lists or models covering process-related defeaters and process elements which should be addressed in confidence arguments. We used Common Characteristics Map from [18], the list of factors influencing assessor confidence from [21] and defeater checklist from [19].
2. Reviewing publicly available assurance cases and related documents from various domains: medical devices [26][27], airspace control [28] and railway [29] to identify processes and resources used by them and referred to in assurance arguments.
3. Analyzing the experience-based knowledge available in GSN standard [1] about assurance case reviews and problems commonly found during them.
4. Combining the inputs from steps 1-3 into a consistent list, removing potential duplicates, unifying the language and designing the structure to represent the results.

Our proposal is based on the observation, that the key, central entity is the artefact used as an evidence to support assurance case claims. Other process elements (like activities, people, tools or input resources) appear in the confidence argument only as a means to increase the confidence in that evidence item. The artefacts include e.g.:

- Reports of hazard and risk analyses;

- Requirements including safety/security requirements;
- Technical documentation describing system's architecture, implementation, components used, interlocks applied etc.;
- Test specifications and reports;
- Formal proofs of correctness;
- Field reports;
- Procedures addressing operation and maintenance.

Not all such artefacts have to be authored by the organization responsible for the development of the high-integrity system under consideration and its assurance case, as for example, when a third party solution is used, a relevant artefact describing this solution could be included as evidence. We however exclude from our area of interest items like: regulations, norms and standards, guidelines used in a particular industry domain etc. Such artefacts are referenced by assurance cases, but as a rationale for argument decomposition, not as evidence. Besides, they are considered trustworthy – assurance case developer is not expected to identify and address weaknesses of an international safety standard. Of course, one can challenge the decision of using a given standard or guideline as the basis for the development of an assurance case or its part, but it would be a defeater against reasoning, not against evidence.

Our research study described above resulted in the identification of the process elements listed below. The references given for each element indicate the sources that contributed to developing its description structure.

1. Artefact – the basic information (necessary e.g. for the auditors) that allows to identify the artefact and its version, summarize its contents and retrieve its source [27-29]
2. Author – the characteristics of people responsible for the development of the artefact, their background and involvement. [1][21]
3. Process – description of the process, method or technique used as part of artefact development [18][21]
4. Tool – the characteristics of the tool used in the process of artefact development e.g. compiler, proof checker, automated testing tool. [21][27][28]
5. Contents – considerations of the quality and adequacy of artefact's contents, as well as the role it should play in the assurance case argument. [1][21]
6. Language – the description of the language used in the artefact, including conventions, abbreviations and symbols. It is important to prevent communication errors when the artefact is referenced from the argument. [27-29]
7. Associations – the explicit list of associated artefacts together with considerations related to their quality. Such characteristics allow to eliminate defeaters implying that an artefact is based on some other unreliable artefacts or that some hidden interdependencies exist. [18][27-29]
8. Reviews – details about conducted reviews of the artefact, including incorporation of reviewers' remarks. [18][27-29]

Each of the listed process elements is described through a number of its characteristics. We make a distinction whether a characteristic is generic and associated with the

element regardless of context or context-specific i.e. important in the context of arguing against a particular defeater. For example, a name and contact information of artefact's author are certainly generic and should always be provided together with the artefact. On the other hand, description of author's experience and technical competence can form a long resume and only a part of such information would matter in the context of a given confidence argument. Our example from Fig. 1. would likely require confidence argument that each member of the team responsible for hazard identification task is sufficiently qualified and experienced. If so, it would be important to document each person's experience in hazard identification, his/her courses, trainings, certificates etc., but the information that someone has a substantial job experience in software testing and is a certified Scrum Master would not be relevant here. Thus, in accordance to our postulate that confidence arguments should be compact and focused on addressing particular defeaters, we insist on providing only such process-based evidence that is relevant for a given context.

The example description structure including characteristics for "Author" element is shown in Table 1. The meaning of the "G/CS" column is whether a given characteristic is generic or context-specific. Due to space limitations we are not able to present any more description structures here, however a full report is available online[1].

Table 1. Description structure for „Author" element.

Characteristic	Description	G/CS
2.1. Personnel	The personal data of the person participating in the development of an artefact. In case an artefact is a product of teamwork, all team members should be listed (and characterized in the following rows). Contact info like phone number and e-mail should also be provided.	G
2.2. Domain knowledge and experience	The confidence that the author has a knowledge about the industrial domain and experience in working on similar projects. Job history and projects conducted are the most important information to be included here. In addition: reports, scientific papers or presentations given can be mentioned.	CS
2.3. Technical competence	The confidence about technical skills and competencies, based on education history (diplomas), certificates obtained, courses finished etc.	CS
2.4. Independence	The confidence that the work conducted by the author on the artefact is free from unwanted dependencies e.g. the same person develops and tests a software module or a subordinate verifies the document created by his/her superior. A description explaining the separation of task assignment and lack of conflict of interests.	CS
2.5. Team organization	In case of teamwork, team structure and responsibilities should be described. Also the leader who is responsible for the artefact should be explicitly determined.	G

[1] https://drive.google.com/open?id=1uo-PlUhPLJ2KIY1ck9kBkBfmifCl8c67

4 Representing Process-based Evidence in Confidence Arguments

In our earlier work [19], we proposed that confidence arguments should be strongly context-related and associated with a particular step of the main assurance argument. In order to avoid introducing additional concepts and notational elements, we proposed a solution where an element which explains the reasoning used in a given step of argumentation (called rationale or justification in existing notations) is the "root" of a confidence argument regarding this step. It makes sense, as such rationale should convince a reader that, given the premises (lower-level claims, facts etc.), the conclusion (higher-level claim) is valid – therefore it should also address any defeaters targeting evidence, reasoning or any other part of this argumentation step. Moreover, it is consistent with the original Toulmin's model.

In this paper we build on the mentioned previous work and extend it with the approach of handling process-based evidence. Based on the available sources and on reviews of available assurance cases, we compiled a list of essential process elements and we defined a set of characteristics for each element. We also introduced the distinction between generic and context-specific characteristics. We suggest that such distinction should be reflected in the way how evidence elements are defined and referenced from the assurance case argument.

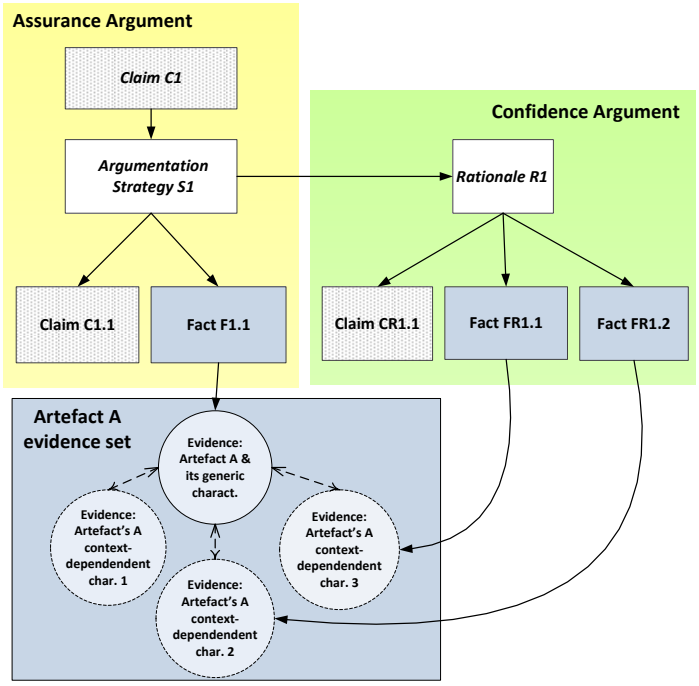

Fig. 2. Using process element characteristics as evidence.

First, let us consider the generic characteristics of process elements associated with a given artefact used as evidence. Such characteristics mostly provide the basic information about the artefact (following the earlier example: the list of identified hazards has its identifier, document version, identification of its authors, name and version of the tool it was stored in etc.), which is important whenever an artefact is referred to, regardless of the context. Such information should therefore be always stored together with the artefact itself. The artefact, together with its generic characteristics, will be referenced as direct evidence for the primary assurance argument.

Now, let us consider the context-dependent characteristics. One could store them together with the artefact, but it would result in a large portion of data that would never be used as a whole, instead only particular selected information would be needed. For example, when dealing with defeaters regarding "list of identified hazards", we would need evidence confirming team leader's experience in hazard analysis, proficiency in using FMEA method etc., while other qualifications he/she may possess (e.g. being an experienced system architect or a certified tester) would not be relevant here. On the other hand, evidence confirming such qualifications may be crucial when system architecture or test reports are referenced by some other parts of the assurance case. The evidence documenting context-dependent characteristics should be considered a backing evidence and is to be referenced from confidence arguments. It should be kept in the form of several evidence items, separately from the artefact, but linked to it.

The proposed solution is presented in Fig. 2. Please note that evidence management is not treated here as integral part of assurance case. Evidence set concerning Artefact A includes the artefact itself together with its generic characteristics. The other, context-dependent characteristics are kept as separate evidence items within that set. Interrelationships between the artefact and related evidence items are maintained. The artefact is referenced by a fact from the primary assurance argument. The use of this artefact is addressed in the associated confidence argument – if defeaters concerning the trustworthiness of the artefact are identified, the confidence argument references appropriate evidence items describing relevant context-dependent characteristics (e.g. author's competence, integrity of tools used).

5 Conclusions

In this paper we discussed the existing approaches to represent confidence arguments and to model process elements that are used as part of assurance case argumentation. By analyzing existing published proposals and by reviewing assurance cases and related reports, we identified process elements together with their characteristics that we consider as important information for assurance case development and review. We distinguished generic and context-specific characteristics and recommended that the former should be used in the primary assurance argument and stored together with the evidence artefact, while the latter is intended for confidence arguments and should be kept in the form of several, fine-grained evidence items. Finally, we demonstrated how an assurance case using both kinds of evidence can be structured.

We are aware that the approach using explicit confidence arguments is a demanding task, but on the other hand, process-based evidence and other evidence e.g. explaining the applied reasoning are still included in assurance cases, while, according to the literature mentioned in Section 1, assurance and confidence arguments should not be mixed as it results in a number of drawbacks.

Our proposed description structures include a significant number of characteristics that would have to be documented during the project of high-integrity system development, which means additional effort of project participants. Our proposal is however of similar scale as the other proposals we described as related work in Section 2. Also, considering what is at stake – auditors refusing to accept the system or worse: the system endangering people's health and lives – it does not seem a steep price.

We conducted a preliminary validation case study, in which an existing assurance case argument was refactored. The refactoring included extracting the process-related parts of the argument to create confidence arguments and applying the approach to manage and reference evidence items described in Section 4. The case study confirmed that the proposed approach can be used, however we are aware that more sound validation should be provided in future.

A promising direction of further research is to use process elements' description structures to define confidence patterns (schemes of building confidence arguments for frequently encountered defeaters). As we are currently implementing tool support for automated patterns instantiation, it may result in a valuable assistance to assurance case developers.

References

1. The Assurance Case Working Group (ACWG): Goal Structuring Notation community standard version 2 (2018).
2. CENELEC: EN 50126. Railway Applications: The Specification and Demonstration of Reliability, Availability, Maintainability and Safety (RAMS) (1999).
3. Eurocontrol: Safety Case Development Manual (2006).
4. International Organization for Standardization (ISO): ISO/DIS 26262: Road Vehicles - Functional Safety (2011).
5. International Organization for Standardization (ISO): 15026-2:2011: Systems and Software Engineering -- Systems and Software Assurance -- Part 2: Assurance Case (2011).
6. Stensrud, E., Skramstad, T., Li, J., Xie, J.: Towards Goal-Based Software Safety Certification Based on Prescriptive Standards, in: First International Workshop on Software Certification (WoSoCER), pp. 13-18 (2011).
7. Sklyar, V., Kharchenko, V.: Assurance Case Driven Design based on the Harmonized Framework of Safety and Security Requirements. in: 13th International Conference on ICT in Education, Research and Industrial Applications (ICTERI 2017), pp. 670-685 (2017).
8. Yamamoto, S., Kaneko, T., Tanaka, H.: A proposal on security case based on Common Criteria" In Information and Communication Technology-EurAsia Conference, pp. 331-336, LNCS vol. 7804, Springer (2013).
9. Sujan, M., Spurgeon, P., Cooke, M., Weale, A., Debenham, P., Cross, S.: The development of safety cases for healthcare services: practical experiences, opportunities and challenges, Reliability Engineering & System Safety, Vol. 140, pp. 200-207 (2015).

10. Górski, J., Jarzębowicz, A., Miler, J.: Validation of services supporting healthcare standards conformance. Metrology and Measurement Systems 19(2), pp. 269-284, (2012).
11. Toulmin, S.: The Uses of Argument, Updated Edition, Cambridge University Press (2003).
12. Hawkins, R., Kelly, T., Knight, J., Graydon, P.: A New Approach to Creating Clear Safety Arguments, in: Advances in Systems Safety, Springer, pp. 3–23 (2011).
13. Goodenough, J., Weinstock, C., Klein, A.: Eliminative induction: a basis for arguing system confidence, Proc. of the 2013 International Conference on Software Engineering, IEEE Press, pp. 1161-1164 (2013).
14. Pollock, J.L.: Defeasible Reasoning, Cognitive Science, Vol. 11, pp. 481-518 (1987).
15. Verheij, B.: Evaluating arguments based on Toulmin's scheme, Argumentation, Vol. 19, No. 3, pp. 347-371 (2005).
16. Kelly, T.: Reviewing Assurance Arguments - A Step-by-Step Approach, Proc. of Workshop on Assurance Cases for Security, Dependable Systems and Networks Conf. (2007).
17. Grigorova, S., Maibaum, T.: Argument Evaluation in the Context of Assurance Case Confidence Modeling, Proc. of 25th IEEE International Symposium on Software Reliability Engineering Workshops, Naples, Italy, IEEE Computer Society, pp. 485-490 (2014).
18. Ayoub, A., Kim, B., Lee, I., Sokolsky O.: A systematic approach to justifying sufficient confidence in software safety arguments, Proc. of 31st International Conference on Computer Safety, Reliability and Security, LNCS 7612, pp. 305-316 (2012).
19. Jarzębowicz, A., Wardziński, A.: Integrating Confidence and Assurance Arguments, in: 10th IET System Safety and Cyber Security Conference (2015).
20. Graydon, P., Knight, J., Yin, X.: Practical limits on software dependability: a case study, In: International Conference on Reliable Software Technologies, pp. 83-96 (2010).
21. Nair, S., Walkinshaw, N., Kelly, T., de la Vara, J.L.: An evidential reasoning approach for assessing confidence in safety evidence, in: 26th IEEE International Symposium on Software Reliability Engineering (ISSRE), IEEE, pp. 541-552 (2015).
22. Hawkins, R., Richardson, T., Kelly, T.: Using Process Models in System Assurance, in: Computer Safety, Reliability, and Security, LNCS vol. 9922, A. Skavhaug, J. Guiochet, F. Bitsch (eds.), Springer International Publishing, pp. 27–38 (2016).
23. Asplund, F., Törngren, M., Hawkins, R., McDermid, J.A.: The Need for a Confidence View of CPS Support Environments, In: 16th International Symposium on High Assurance Systems Engineering (HASE), pp. 273-274 (2015).
24. Retouniotis, A., Papadopoulos, Y., Sorokos, I., Parker, D., Matragkas, N., Sharvia, S.:, Model-connected safety cases, In International Symposium on Model-Based Safety and Assessment, LNCS vol. 10437, pp. 50-63, Springer, Cham (2017).
25. Holloway, C.M., Graydon, P.: Evidence Under a Magnifying Glass: Thoughts on Safety Argument Epistemology, In Proc. of 10th IET System Safety and Cyber Security Conference, Bristol, UK, (2015).
26. Larson, B.R., Hatcliff, J., Chalin, P.: Open source patient-controlled analgesic pump requirements documentation. In: 5th International Workshop on Software Engineering in Health Care (SEHC), pp. 28–34 (2013).
27. Larson, B.R.: Open PCA Pump Assurance Case, SAnToS Research Group, Kansas State University, http://openpcapump.santoslab.org/ (2014).
28. Civil Aviation Office, Poland & Civil Aviation Administration, Lithuania: Baltic FAB Safety Case, https://www.pansa.pl/aap/Safety_Case_2.0.pdf (2012).
29. London Underground Limited: London Underground Safety Certificate and Safety Authorisation, ver. 5.1, http://content.tfl.gov.uk/london-underground-safety-certificate-and-safety-authorisation.pdf (2017).