The Safety Argumentation Schools of Thought

Patrick John Graydon

NASA Langley Research Center, Hampton, VA 23681, USA
patrick.j.graydon@nasa.gov

Abstract. Safety cases have been produced and researched for decades. Definitions of 'safety case' agree on both the need to generate suitable evidence and the central role of argument. But the relevant literature seems to exhibit multiple schools of thought that are largely unrecognized and somewhat at odds with each other. This paper presents preliminary results from research to identify and characterize the safety case schools of thought so as to reduce confusion and discord in research and practice.

1 Introduction

Safety cases—the safety-specific form of *assurance cases*—have been produced, reviewed, researched, and written about for decades [42]. Definitions of safety case agree on the need for suitable evidence and, recently, the central role of argument. But the literature exhibits multiple *schools of thought* on safety cases and their value. We hypothesize that (1) several distinct but largely unidentified schools of thought exist and (2) no one form of safety case suits them equally well. If so, identifying and distinguishing the schools of thought will help stakeholders better understand, interpret, and contribute to the relevant literature.

Shared understanding of safety cases is challenged by stakeholders' diversity of domains and technical specialities. Without such an understanding, specialists in a particular field may make recommendations that seem to them well-justified yet are nevertheless detrimental. A newcomer might hear about other regulators' success with safety cases, read about a subtly different thing also called a safety case, and fall prey to an equivocation fallacy.

Differences between the schools of thought might explain differences among recommendations for practitioners. For example, researchers have argued in favor of graphical argument notations, structured prose, and symbolic logic [29,34,47]. These mutually exclusive endorsements might reflect different interpretations of limited evidence for each notation's efficacy. But they might also reflect different understandings of the purpose and scope of safety cases.

In this paper, we report on a literature survey in progress. To date, we have identified nine schools of thought and six inconsistencies in what they ask of arguments. We have not identified the complete range of thought or perfectly captured any part of it, and we are not the first to note apparent differences in the use of the term safety case [36]. But safety case research is ongoing, safety case practice is expanding—e.g., as ISO 26262 [31] is adopted—and the schools of thought remain apparently unrecognized. We hope that these early results

will both (a) promote clarity in ongoing research and practice and (b) prompt conversation that will help to identify and characterize the schools of thought.

2 Asking Better Questions

Typical definitions of 'safety case' focus on *form* and emphasize either the case's collective nature or an explicit argument. For example, the UN Nuclear Energy Agency (NEA) defines a waste repository safety case as "a formal compilation of evidence, analyses and arguments that quantify and substantiate a claim that the repository will be safe" [40]. The Goal Structuring Notation (GSN) standard defines an assurance case as an "argument, supported by a body of evidence, that a system, service or organisation will operate as intended for a defined application in a defined environment" [3]. Though useful, these definitions do not suggest testable efficacy hypotheses or the relative merits of, e.g., argument notations. To clarify efficacy hypotheses found in the literature, we asked:

1. *What value does a safety argument deliver?*
2. *To whom does a safety argument deliver each kind of value?*
3. *How does a safety argument produce that value?*
4. *How well does a safety argument produce each kind of value?*

Evidence answering the last question would underpin decisions about safety case practice. Such evidence is badly needed: one might deploy an untested tool or technique in a low-consequence application with little justification, but increasing consequences demand correspondingly stronger evidence of efficacy.

3 Schools of Thought

In 2016, Rinehart et al. conducted interviews and a literature survey to assess what it means for assurance cases to 'work' [42]. While insightful, that work did not fully address the questions presented in Sect. 2. We are reassessing both the assurance-case-related works they surveyed—and works published since—to better characterize the safety case schools of thought. This section presents, in arbitrary order, nine schools of thought we have identified to date.

The schools of thought are not mutually exclusive: while each identifies a distinguishable stream of thought, single sources often exhibit several at once. Moreover, the literature survey is not systematic. It need not be: our aim is not to assess either the balance of evidence for and against a proposition or the prevalence of any school of thought. We aim instead to identify and characterize as many schools of thought as practicable. While our literature survey remains incomplete presenting these preliminary findings will help us to identify more schools of thought and refine our current descriptions.

3.1 A Safety Case (Report) Is (the Product of) a Risk Assessment

This school of thought defines a safety case as a (residual) risk assessment, possibly summarized in a safety case report. An explicit, overarching safety argument is either omitted or less central than risk assessment and management.

Kind of argument. Safety case documentation might include explicit argument, e.g., to justify predictions about the efficacy of a safety management system or new mitigations. Such arguments are likely to be informal and loosely structured.

Values, beneficiaries, and mechanisms. First, by generating insight into application-specific risks, "doing a safety case" is said to yield more effective risk mitigation than, e.g., implementing prescribed safety features. Second, safety cases are said to better inform decision-makers by communicating risk assessments more effectively than, e.g., unstructured collections of safety artifacts.

Examples. This school of thought is often exemplified by discussions of safety case development as an approach to safety assurance. For example, one textbook notes that "developing a safety case through analysis of [residual] risk is one way of finding out details about the level of risk that people or things are being exposed to" and introduces HAZOP, fault trees, event trees, ZHA, FMEA, etc. as "techniques and tools for safety cases" [37]. One inquiry meant to "examine the arrangements for assuring the airworthiness and safe operation" of an aircraft famously labeled that aircraft's safety case "a lamentable job" owing mainly to shortcomings in its risk assessment [23]. A paper about nuclear waste repositories notes that "in recent years, the scope of the safety assessment has broadened to include the collation of a broad range of evidence and arguments that complement and support the reliability of the results of quantitative analyses and the broader term 'safety case' is used to refer to these extended studies" [5].

3.2 A Safety Case Documents the Story of a System's Safety

In this school of thought, a safety argument or narrative tells diverse stakeholders what it means for a system or service to be safe and how it achieves safety.

Kind of argument. The bulk of the argument traces the most significant hazards to safety requirements and assessment results. A writer might argue over specific software contributions to hazards—rather than about such contributions as a class—thereby communicating the writer's contention that the specified contributions are what matter in the application at hand.

Values, beneficiaries, and mechanisms. By clearly communicating the safety story, the safety argument facilitates the work of each stakeholder. For example, a developer who is aware of the safety story might be less likely to inadvertently undermine safety than one who is not. An assessor who is aware of the story might be better able to judge the appropriateness of a particular type of evidence. It is the accessibility, clarity, conciseness, and detail level of the tale—properties that might be at odds in some cases—that determine the argument's value.

Examples. This school of thought often appears in discussion about how safety cases benefit a large and diverse readership. For example, Alexander et al. write that "whenever we honestly claim that a system is acceptably safe or secure, ...

we have some mental model behind that claim that we could probably describe if asked. By making an explicit assurance case, however, we open up that mental model to review and criticism by others, and we record our reasoning so that others can learn from it (for example, if they want to make a change to the system they can use the assurance case to assess some of the impact)" [2]. The *GSN Community Standard* notes that the key benefit of explicitly documenting a safety argument is that doing so "can improve comprehension amongst the key stakeholders (e.g., system developers, engineers, independent assessors and certification authorities)" [3]. This in turn "improves the quality of the debate and the time taken to reach agreement on the [safety] approaches being adopted." An NEA report notes that safety arguments for waste repositories are meant to describe, among other things, the safety management strategy, siting and design strategy, safety assessment strategy, system concept, and repository design to be used [40].

3.3 A Software Safety Argument Shows Requirements Refinement

Adherents to this school of thought maintain that a safety argument shows how safety requirements are correctly refined through tiers of design detail level. In doing so, the argument replaces alternatives such as traceability matrices.

Kind of argument. This school of thought is often applied to software safety arguments that show how each system safety requirement allocated to software is refined by high-level software requirements, low-level software requirements, and the source code (which is itself a form of specification) [47]. The argument cites software-level testing, integration testing, unit testing, and other verification and validation evidence at appropriate levels.

Values, beneficiaries, and mechanisms. First, an argument tracing requirements down to evidence might benefit (a) engineers planning verification and validation and (b) assessors reviewing those plans by more clearly communicating the relevance of each evidence item than alternatives such as RTCA DO-178C conformance documentation [45]. Second, an argument tracing requirements through levels of design detail might better communicate to programmers each module's contribution to system safety than alternatives such as traceability matrices. Third, such an argument might facilitate better detection of requirements decomposition errors than reviews and analyses using only alternatives such as traceability matrices. Some researchers suggest machine analysis of arguments for this purpose, but such approaches remain without justification [15].

Examples. This school of thought is often evident in software safety argument patterns. For example, the *Software Contribution Safety Argument Pattern* proposed by Hawkins and Kelly explicitly traces software safety requirements through tiers of design decomposition [25]. Proposed patterns for the safety arguments that ISO 26262 requires for electrical or electronic systems in road vehicles trace safety goals to functional safety concepts, technical safety concepts, and hardware and software safety requirements [7,31].

3.4 A Safety Case Establishes Confidence in Safety Claims

In this school of thought, safety cases justify an assessable degree of confidence in claims about safety or its contributing factors. The degree of confidence depends on several factors, including the nature and quality of evidence cited [35].

Kind of argument. Various kinds of argument have been proposed to facilitate assessment of argument confidence [19]. For example, a *confidence argument* might present known *argument defeaters* and their resolution, if any, to facilitate qualitative judgments of argument sufficiency [13,27]. Other researchers propose breaking arguments down into specific forms of reasoning steps—e.g., *complementary* or *sufficient condition list* arguments—to facilitate using formulae to compute confidence in claims from judgments of evidence strength [10,19].

Values, beneficiaries, and mechanisms. First, regulators might use confidence assessment results as the basis for a decision to certificate an aircraft type or compel a recall or remediation. Second, the review and analysis needed to assess confidence might show developers where it is being lost, thus helping to more effectively target resources spent on improving the overall safety assessment.

Examples. This school of thought is evident in proposals to evaluate argument confidence. For example, Ayoub, Górski, and others propose using *Dempster–Shafer Theory* to evaluate the confidence an argument should inspire [4,19,50]. Others have proposed using *Bayesian Belief Networks* for this purpose [19]. Hawkins et al. propose identifying potential argument *defeaters* in a *confidence argument* to facilitate holistic confidence judgments [27]. Goodenough et al. propose documenting these defeaters in a *confidence map* and counting them as a measure of confidence [13,16]. It also appears in a more general form in the proposition that the function of an argument is to inspire confidence in a reader. For example, security arguments have been said to "convince a reader that a system can satisfy the security requirements laid upon it" [24].

Note. It is sometimes posited that preparing a safety case readies developers to argue in a law court, after a mishap, to have met their safety-related legal obligations [36]. While the claim in question and the target audience are different, this claimed value also turns on safety cases establishing confidence.

3.5 A Safety Case Promotes and Focuses Safety Thinking

In this school of thought, creating a safety case forces developers to think about the context-specific meaning of safety and how to achieve and demonstrate it.

Kind of argument. The scope and depth of the safety argument is critical to this school of thought: a writer that does not address a topic or issue in detail will not have been forced by the writing process to think deeply about that topic or issue and thus will not derive the related benefit.

Values, beneficiaries, and mechanisms. The core claim of this school of thought is that the act of writing a safety argument forces developers to ponder critical questions [49]. Doing so produces more insight into (a) hazards, (b) the way they are mitigated or managed, and (c) the means of verifying and validating the system implementation than following a prescribed process would have.

Examples. The value derived from critical reflection is often cited as a reason to adopt an assurance case process. For example, a healthcare industry report proposing the adoption of safety cases in that industry observes that "safety cases were explicitly introduced in a number of domains to encourage systematic and holistic thinking about safety issues" [9]. Others have asserted that safety cases are "a vehicle to stimulate critical thinking" [46]. In its most extreme form, this school of thought manifests in the claim that safety arguments can be used to perform what is effectively a form of hazard analysis or causal analysis. For example, Eagles and Wu assert that argumentation is a "tool that can facilitate both top-down and bottom-up analyses," including "identification of hazardous situations, causes, or subcauses, including low-level causes" [12]. Kang and Jackson "describe a technique for analyzing a dependability argument—the argument that a trusted base is alone sufficient to establish a requirement; that is, it is not missing a component that is also necessary for the requirement to hold" [32].

3.6 A Safety Argument Explains and Integrates Safety Evidence

In this vision, a safety argument *explains* and *integrates* its safety evidence [34].

Kind of argument. In this school of thought, safety arguments link evidence types to claim types. Safety arguments might argue over requirements individually or as part of categories such as 'software functional requirements' (as in arguments representing standards [30]). Evidence might be as specific as named test cases or as general as 'black-box software functional testing.' The argument might include details relevant to assessing the evidence's strength or validity [27].

Values, beneficiaries, and mechanisms. The argument illustrates how the cited evidence collectively covers a proposition to be shown [17,42]. First, developers arguing this is thought to help developers create more efficient or effective development plans than had they simply describes those plans as, e.g., as a plan for software aspects of certification (PSAC) [45]. Second, assessors are thought to better judge the sufficiency of evidence or identify gaps or potential improvements than had they been presented with an alternate form of documentation. Third, this might help reviewers analyzing a standard to identify gaps and flaws in its assurance requirements [20]. Fourth, these arguments might help researchers to identify hypotheses about the efficacy of selected techniques.

Examples. This school of thought is often evident in papers about certification. As one report observes, "all evidence provides only partial support for a given high-level claim" [2]. As a Health Foundation report observes,

> a diversity of evidence sources and types are required to demonstrate system safety – such as trials, human factors analysis, testing and operational experience. However, this diversity and amount of evidence can create difficulties. It can be difficult to judge completeness. Is the evidence set comprehensive? Does it cover all the issues? It can also be difficult to understand the distinct role and purpose served by each form of evidence. Safety cases help in this regard, by presenting the argument that explains how the overall safety objectives can be seen to be addressed through the assembled items of evidence [9].

Ruiz et al., paraphrasing Eagles and Wu, observe that medical safety cases "ensur[e] the completeness of risk identification and risk controls" [12,46]. Similar observations have been made for security arguments [39]. For example, Tippenhauer et al. describe security arguments as "mak[ing] explicit the underlying interdependecies of different pieces of security-related information" [48].

3.7 A Safety Argument Focuses Systematic Regulatory Inquiry

This school of thought is concerned with how regulators assess systems and make decisions such as whether to certify a system or require redress of an issue.

Kind of argument. This school of thought is not limited to any form of argument.

Values, beneficiaries, and mechanisms. Regulators systematically assess a submitted argument, investigating other artifacts more deeply as needed and in light of their purpose as expressed in the argument [33,35]. Such inquiries are thought to reveal more issues in an inspector's limited time than alternatives such as audits focused on issues implicated by recent or notable incidents or accidents.

Examples. This school of thought is evident in descriptions of how safety cases are used in regulation. For example, Eagles and Wu write that "the structured documentation provided by a safety assurance case can help an independent reviewer evaluate the rationale and evidence for safety efficiently and effectively, without requiring the same level of familiarity with the device as a member of the development team" [12]. This school of thought is also evident in proposed techniques for reviewing of safety arguments [21,33].

3.8 A Safety Argument Facilitates Choosing Among Potential Mitigations and Lifecycle Activities

In this school of thought, developers use safety arguments as whiteboards to propose and assess designs and plans for development, verification, and validation.

Kind of argument. This school of thought requires the safety argument (1) to cover the hazards to be mitigated; (2) to give the goals to be met by lifecycle activities; and (3) to be updated after each development decision is made. Some techniques might require the use of specialized (e.g., formal) argument notations.

Values, beneficiaries, and mechanisms. By creating and assessing variants to explore potential decisions, developers gain insight into the application-specific effect those decisions might have. This might "allow targeting of resources and efforts, thus avoiding spending wildly varying and disproportionate amounts of effort on risk management" (in comparison with using standard best practices regardless of circumstance) [9].

Examples. This school of thought is evident in exhortations to "integrat[e] the safety case into the design and development process" and "design for assessment" [8]. As one tool developer put it, "a safety case can ... act as a focal point for project management since material in the safety case provides a lot of material upon which management decisions are made" [1]. The same developer also observed that "development of arguments for the safety cases of complex systems frequently requires extensive and protracted manipulation of arguments to facilitate the exploration of alternative lines of argumentation." The NEA report quoted earlier observes that safety cases are often

> compiled and presented at certain stages of a stepwise repository development programme with an aim to inform decision makers about whether adequate information is available to that decisions to proceed to the next step can be made. ... As a repository development programme continues to advance, the safety case provides an important basis for repository development activities including research and development (R&D) [40].

This school of thought finds its purest expression in detailed proposals for safety-argument-centric development methodologies such as *Assurance Based Development* and Despotou's systems-of-systems assurance framework [11,14,35].

3.9 A Safety Argument Guides Accident Investigation

This school of thought holds that, where available, safety arguments should guide accident and incident investigation and response.

Kind of argument. The argument must cover each mitigation for each hazard (rather than argue over them as a class in the manner of a process argument).

Values, beneficiaries, and mechanisms. A safety argument presents claims about the safety-related behaviors and features of a system or service. If the conclusion of adequate safety is false, one or more of those claims, or the logic connecting them, must be in error. By systematically testing the argument's claims, an investigator might gain insight into the causes of an accident more effectively or efficiently than by using other accident investigation methods.

Examples. The Pandora approach to analyzing digital system failures uses the safety argument (1) to suggest hypotheses about causal factors to be investigated and (2) to assess proposed mitigations [22]. Security researchers have suggested similar uses for security arguments [44].

3.10 Related Schools of Thought: Safety Argument Analogues

There are schools of thought surrounding arguments that are analogous to assurance arguments focused on safety or security. We identify two here.

Modeling Standards. An analyst might model the argument implicit in a safety or security standard [20,30]. The purposes for doing so might include:

- To be critically analyzed in order to assess the standard, answering questions such as, *Should the standard call for additional evidence?*
- To facilitate decisions about evidence substitutions, e.g., whether to accept *alternative methods of compliance* with DO-178C [45].
- Pedagogical purposes, i.e., teaching the core logic of the standard that must be appreciated if readers are to correctly interpret its requirements.

Analyzing or Teaching Safety Tools, and Techniques. Analysts might model the arguments surrounding tools and techniques [17]. The analysis might promote thinking about the tool's role in achieving or demonstrating safety, thus helping the analyst to identify critical factors that determine how well the tool or technique supports a given kind of assurance claim. Such factors comprise potential defeaters for safety arguments that rely on those tools, techniques, or mitigations [13,16,27,43]. The resulting argument might be used to teach developers about those defeaters.

4 Some Tension Between the Schools of Thought

In Sect. 3, we identified nine distinct but overlapping schools of thought. Having multiple schools of thought can be problematic if they are unrecognized and no single form of safety case suits all schools equally well. In this section, we describe six issues of contention for safety case practice and, for each, identify schools of thought that would favor different resolutions to the issue.

This list of tensions is incomplete. Moreover, the existence of tensions need not preclude the joint application of schools of thought to a safety case provided stakeholders are aware of the tension and appropriate tradeoffs are made.

4.1 The Scope and Completeness of Safety Arguments

The complete safety case for a system or service must address each hazard and safety requirement. But an argument could explicitly represent all, none, or some of these (in the latter two cases using categorical arguments and detail relegated to linked documents). These options vary in desirability according to school of thought. If the argument is meant to show requirements refinement (Sect. 3.3), it must trace all requirements through all design levels. But if such detail obscures the story of a system's safety (Sect. 3.2), argument writers might focus on key requirements, relegating complete details to linked documents.

A complete safety case must also discuss both *process* of producing artifacts and evidence and safety-relevant features of the *product* produced. The balance of attention the explicit argument pays to each of these might vary. Arguments focused on process adequacy might better demonstrate confidence (Sect. 3.4) and facilitate regulatory inquiry (Sect. 3.7) than arguments that illustrate the decomposition of all requirements. Human reviewers do not have unlimited time; the more their attention is directed to important matters, the more productive their auditing can be expected to be. But if confidence is computed, it might be necessary for the argument to cover all hazards and safety requirements.

4.2 The Depth of Safety Arguments

Top-down argument creation methods raise a question: when should arguers stop decomposing claims? There is no universal test for whether a claim is basic enough to be supported directly by evidence. Indeed, while most professionals will agree on whether a given thing is evidence in a given case, neither the safety argumentation literature nor the literature on its philosophical foundations offers a useful, concrete definition of 'evidence' [18]. This raises many questions of the basic form *Should we argue* over *or* through *X?* For instance, consider a claim that software will cause a computer system to exhibit a desired behavior. A writer might support this claim by appealing to a corresponding software safety requirement, an appropriate integrity level assignment, and software conformity review results. But a writer might instead argue through software design levels, documenting requirements traceability and the role of integration testing along the way. If an argument is to show requirements refinement (Sect. 3.3), the writer must take the deeper course. But arguments that focus on a few key requirements might better document the story of safety (Sect. 3.2). Moreover, if adequate evidence of requirements refinement has already been provided, auditors' time might be better used by an argument that invites them to question the proposed mitigations and integrity level allocations (Sect. 3.7).

4.3 Safety Argument Syntax and Structure

Definitions of argument notations describe basic syntactic limits. For example, GSN permits goals to support other goals either directly or through a *strategy* element [3]. Some argument construction methods place few further restrictions on such argument steps [3]. Some techniques are more restrictive. For example, one confidence quantification proposal requires arguments to be constructed of argument steps that each can be classified as a *complementary argument, alternative argument, sufficient condition list argument,* etc. [10]. One might argue that a "system is acceptably safe to operate" because of n premises of the form "hazard i mitigated" [27]. Or one might first argue that "all significant hazards have been identified" and "identified hazards have been acceptably mitigated," only then supporting the latter claim with the same premises. One could, in principle, expand any complex argument step into an equivalent combination of more basic steps. Doing so might facilitate confidence assessment (Sect. 3.4),

either by satisfying the structural requirements of a quantification technique or by making the argument step easier for human reviewers to critically analyze. But doing so would likely enlarge the argument structure, which might make it harder for readers to grasp the big picture of the safety story (Sect. 3.2).

4.4 The Use of Argument Patterns

Safety argument patterns have long been proposed as a means of reusing successful fragments of argument [34]. But their use might have effects beyond the ease of writing. For example, a pattern might function as a kind of framing device, helping readers to grasp the big picture quickly. If so, arguments comprising patterns might better document the story of a system's safety than entirely bespoke arguments (Sect. 3.2) [34]. A conscientious writer might thoroughly examine whether a pattern reflects the system or service in question without oversimplifying or omitting relevant detail. But if writers use patterns less conscientiously, the argument writing process might fail to promote safety thinking (Sect. 3.5). Worse, it is possible that a pattern would be instantiated by a writer who does not fully understand it, thus giving the reader the impression of greater understanding than the writer actually has. If so, the use of patterns might tend to frustrate regulatory inquiry (Sect. 3.7).

4.5 Automatic Argument Generation

Some writers have proposed automatically generating safety arguments, e.g., from automatically constructed proofs or argument fragments associated with architectural model elements [6,26]. Others have proposed construction assistance that falls short of automatic generation, e.g., generating graphical argument structures from patterns and provided parameters [38]. Automatically generated arguments might provide a sound, low-cost basis for establishing confidence (Sect. 3.4). Moreover, arguments generated from formal proofs of requirements refinement might illustrate that refinement more effectively than, e.g., automatically constructed proofs (Sect. 3.3). But such automatic generation might not promote safety thinking (Sect. 3.5). Moreover, automatically generated arguments are likely to be less readable than bespoke arguments and thus to be less well suited to documenting the story of a system's safety (Sect. 3.2).

4.6 Argument Form and Notation

Safety arguments have been written in many forms, including structured and free-form prose, tables, informal logic diagrams such as GSN, and even symbolic logic [3,8,15,28,29,41,42,47]. Some writers draft graphical arguments to fit the dimensions of a printed page, while others use tools that are geared toward on-line viewing of arguments [1]. Some guidance allows writers to omit *warrants* where the connection between claims and premises is self-evident, while other writers insist this should always be stated [15]. Formal notations might have specific

features that facilitate automated analyses which might help to show requirements refinement (Sect. 3.3) or assess confidence in safety claims (Sect. 3.4). But formal languages often require training and expertise to read and understand. As a result, arguments written in them might be less effective at communicating the story of a system's safety to a broad audience (Sect. 3.2) than arguments in informal notations.

5 Conclusion

In this paper, we described nine distinct schools of safety case thought identified in an ongoing literature survey. We also presented six examples of conflicts among these schools of thought. While the schools of thought go largely unidentified in the literature, tensions between them might explain disagreements, e.g., about the merits of formalizing or automatically generating arguments. Lack of uniformity across the literature might also explain the difficulty in getting started that novice safety argument writers sometimes report having [42].

There are more schools of thought than we report here, and the descriptions we give of each are mere sketches of impressions. For example, we could have discussed other functions of safety cases such as developers' demonstration of due diligence [42]. The literature survey that produced these preliminary results is ongoing. Moreover, we are planning collaborative activities to further elaborate the schools of thought, gauge developer and regulator interest in each of these, and characterize potential conflicts between them.

We hope that identifying these schools of thought will contribute to both the research and practice of safety argumentation. For example, by helping readers to understand what writers might mean, we help them to better understand the safety case literature. By helping researchers to better understand each other, we might contribute to forming useful consensuses. And by establishing more precise definitions of what it means for safety cases to 'work,' we might facilitate the sort of experiments and studies whose results should define best practice.

References

1. Aiello, M.A., Hocking, A., Knight, J., Rowanhill, J.: SCT: A safety case toolkit. In: Proc. Int'l Wksp. on Assurance Cases for SW-Intensive Systems (ASSURE). (2014)
2. Alexander, R., Hawkins, R., Kelly, T.: Security assurance cases: Motivation and the state of the art, issue 1.1. Tech. Rpt. CESG/TR/2011/1, University of York, York, UK (2011)
3. Attwood, K., et al.: GSN Community Standard, Version 1. Origin Consulting Limited (2011)
4. Ayoub, A., Kim, B., Lee, I., Sokolsky, O.: A safety case pattern for model-based development approach. In: Proc. NASA Formal Methods. (2012)
5. Baik, M., Park, T.J., Kim, I., Jeong, J., Choi, K.: Development of a natural analogue database to support the safety case of the Korean radioactive waste disposal program. Swiss Journal of Geosciences **108**(1) (2015)

6. Basir, N., Denney, E., Fischer, B.: Deriving safety cases from automatically constructed proofs. In: Proc. IET Int'l Conf. on Systems Safety. (2009)
7. Birch, J., et al.: Safety cases and their role in ISO 26262 functional safety assessment. In: Proc. Int'l Conf. on Computer Safety, Reliability, & Security (SafeComp). (2013)
8. Bishop, P., Bloomfield, R.: A methodology for safety case development. In: Proc. Safety-Critical Systems Symp. (SSS). (1998)
9. Bloomfield, R., et al.: Using Safety Cases in Industry and Healthcare. The Health Foundation (2012)
10. Cyra, L., Gorski, J.: Supporing expert assessment of argument structures in trust cases. In: Proc. Int'l Probabilistic Safety Assessment and Management Conf. (PSAM). (2008)
11. Despotou, G.: Managing the Evolution of Dependability Cases for Systems of Systems. PhD thesis, University of York (2007)
12. Eagles, S., Wu, F.: Safety assurance cases for medical devices. Biomedical Instrumentation & Technology **48**(1) (2014)
13. Goodenough, J., Weinstock, C., Klein, A.: Eliminative argumentation: A basis for arguing confidence in system properties. Tech. Rpt. CMU/SEI-2015-TR-005, Software Engineering Institute (2015)
14. Graydon, P.: Assurance Based Development. Ph.D. thesis, University of Virginia (2010)
15. Graydon, P.: Formal assurance arguments: A solution in search of a problem? In: Proc. Dependable Systems and Networks (DSN). (2015)
16. Graydon, P.: Defining Baconian Probability for use in assurance argumentation. Tech. Memo. NASA/TM-2016-219341, NASA (2016)
17. Graydon, P., Bate, I.: Realistic safety cases for the timing of systems. The Computer Journal **57**(5) (2014)
18. Graydon, P., Holloway, C.M.: "Evidence" under a magnifying glass: Thoughts on safety argument epistemology. In: Proc. IET System Safety & Cyber Security Conf. (2015)
19. Graydon, P., Holloway, C.M.: An investigation of proposed techniques for quantifying confidence in assurance arguments. Safety Science **92** (2017)
20. Graydon, P., Kelly, T.: Using argumentation to evaluate software assurance standards. Information and SW Tech. **55**(9) (2013)
21. Graydon, P., Knight, J., Green, M.: Certification and safety cases. In: Proc. Int'l Systems Safety Conf. (ISSC). (2010)
22. Greenwell, W.: Pandora: An Approach to Analyzing Safety-Related Digital-System Failures. PhD thesis, University of Virginia (2007)
23. Haddon-Cave, C.: The Nimrod Review: An Independent Review Into The Broader Issues Surrounding The Loss Of The RAF Nimrod MR2 Aircraft XV230 In Afghanistan In 2006. The Stationery Office, London (2009)
24. Haley, C., Laney, R., Moffett, J., Nuseibeh, B.: Security requirements engineering: A framework for representation and analysis. IEEE Trans. on SW Engineering **34**(1) (2008)
25. Hawkins, R., Kelly, T.: A software safety argument pattern catalogue. Tech. Rpt. YCS-2013-482, University of York (2013)
26. Hawkins, R., Habli, I., Kolovos, D., Paige, R., Kelly, T.: Weaving an assurance case from design: A model-based approach. In: Proc. Int'l Symp. on High Assurance Systems Engineering (HASE). (2015)
27. Hawkins, R., Kelly, T., Knight, J., Graydon, P.: A new approach to creating clear safety arguments. In: Proc. Safety-Critical Systems Symp. (SSS). (2011)

28. Heavner, E., Holloway, C.M.: Assurance arguments for the non-graphically-inclined: Two approaches. Tech. Memo. NASA/TM-2017-219650, NASA (2017)
29. Holloway, C.M.: Safety case notations: Alternatives for the non-graphically inclined? In: Proc. IET Int'l Conf. on System Safety (ICSS). (2008)
30. Holloway, C.M.: Explicate '78: Uncovering the implicit assurance case in DO-178C. In: Proc. Safety-Critical Systems Symp. (SSS). (2015)
31. ISO 26262-2:2011: Road vehicles — Functional safety — Part 2: Management of functional safety. International Organization for Standardization (2011)
32. Kang, E., Jackson, D.: Dependability arguments with trusted bases. In: Proc. Int'l Requirements Engineering Conf. (RE). (2010)
33. Kelly, T.: Reviewing assurance arguments: A step-by-step approach. In: Proc. Assurance Cases for Security: The Metrics Challenge. (2007)
34. Kelly, T.: Arguing Safety — A Systematic Approach to Managing Safety Cases. PhD thesis, University of York, York, UK (1998)
35. Knight, J., Rowanhill, J., Ferrell, U.: CLASS system certification. Tech. Rpt. TR-2014-4, Dependable Computing LLC (2014)
36. Leveson, N., et al.: Re: [sc] safety cases. Safety Critical Mailing List thread (2012)
37. Maguire, R.: Safety Cases and Safety Case Reports: Meaning, Motivation, and Management. Ashgate Publishing (2006)
38. Matsuno, Y., Taguchi, K.: Parameterised argument structure in GSN patterns. In: Proc. Int'l Conference on Quality SW. (2011)
39. Moore, A., Payne, C.: The RS-232 character repeater refinement and assurance argument. Tech. Memo. NRL/MR/5540--96-7872, Naval Research Laboratory (1996)
40. NEA: The nature and purpose of the post-closure safety cases for geological repositories. Tech. Rpt. NEA/RWM/R(2013)1, Nuclear Energy Agency (2013)
41. Rinehart, D., Knight, J., Rowanhill, J.: Current practices in constructing and evaluating assurance cases with applicaitons to aviation. Contractor Rpt. CR-2015-218678, NASA (2015)
42. Rinehart, D., Knight, J., Rowanhill, J.: Understanding what it means for assurance cases to "work". Contractor Rpt. NASA/CR-2017-219582, NASA (2017)
43. Rowanhill, J., Knight, J.: Domain arguments in safety critical software development. In: Proc. Int'l Symp. on Software Reliability Engineering (ISSRE). (2016)
44. Rowe, J., Levitt, K., Parsons, S., Sklar, E., Applebaum, A., Jalal, S.: Argumentation logic to assist in security administration. In: Proc. Wksp. on New Security Paradigms (NSPW). (2012)
45. RTCA DO-178C: Software Considerations in Airborne Systems and Equipment Certification. RTCA, Inc. (2011)
46. Ruiz, A., Barbosa, P., Medeiros, Y., Espinoza, H.: Safety case driven development for medical devices. In: Proc. Int'l Conf. on Computer Safety, Reliability, & Security (SafeComp). (2015)
47. Rushby, J., Xidong, X., Murali, R., Weaver, T.: Understanding and evaluating assurance cases. Tech. Rep. CR-2015-218802, NASA (2015)
48. Tippenhauer, N., Temple, W., Vu, A., Chen, B., Nicol, D., Kalbarczyk, Z., Sanders, W.: Automatic generation of security argument graphs. In: Proc. IEEE Pacific Rim Int'l Symp. on Dependable Computing (PRDC). (2014)
49. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes. Cambridge University Press (2008)
50. Zagórski, M., Górski, J.: An approach for evaluating trust in IT infrastructure. In: Proc. Int'l Conf. on Dependability of Computer Systems (DepCos-RELCOMEX). (2006)