Safety and Assurance Cases: Past, Present and Possible Future – an Adelard Perspective

Robin Bloomfield and Peter Bishop

Adelard LLP

London, UK

Abstract This paper focuses on the approaches used in safety cases for software based systems. We outline the history of approaches for assuring the safety of software-based systems, the current uptake of safety and assurance cases and the current practice on structured safety cases. Directions for further development are discussed.

1 History of Computer System Safety and Related Standards

The nuclear industry has had a major influence on the development of approaches to safety related computer system development and assurance. From the late 1970s the European Working Group on Industrial Computer Systems (EWICS), a cross-sector pre-standardisation working group, developed a series of guidelines and books that documented best practices. The guidance was subsequently incorporated into the IEC 880 standard on software for nuclear systems (IEC 1986). The experience of EDF and Merlin-Gerin with the first generation of reactor protection systems, SPIN, was fed into the committee. The software engineering approach in the EWICS guidelines (Redmill 1988, 1989) and their book on safety techniques (Bishop 1990) represented the then state of the art.

In the UK there were a number of policy initiatives. The ACARD report (ACARD 1986) and subsequent IEE/BCS and HSE studies (IEE 1989, HSE 1987) set the scene and in 1988 the Interdepartmental Committee on Software Engineering (ICSE) established its Safety-Related Software (SRS) Working Group to coordinate the Government's approach to this important issue. Members were drawn from a wide range of departments and agencies: CAA, CEGB, DES, DoE, DTI, DoH, DoT, HSE, MoD, RSRE and SERC. The work was motivated 'not by recognition of particular present dangers; rather by a desire to anticipate and forestall hazards which may arise with the very rapid pace of technical change'.

The UK Health and Safety Executive (HSE) were active in taking the lead in ICSE and this, with support from DTI, led to a consultation document known as

C. Dale, T. Anderson (eds.), *Making Systems Safer*, DOI 10.1007/978-1-84996-086-1_4,

SafeIT (Bloomfield 1990) and an associated standards framework (Bloomfield and Brazendale 1990). HSE also published awareness documents on the safety of programmable electronic systems (PES), including the Out of Control report (HSE 1993), and some earlier studies that looked at the feasibility of providing a validated framework for selecting software engineering techniques. SafeIT identified four main areas of activity requiring a coordinated approach: standards and certification; research and development; technology transfer; education and training.

The UK MoD were, as one might expect, pioneers in the use of critical software and the development of static analysis tools to analyse the code (Malpas) as well as forays into formally proven hardware designs. In the light of finding defects in certain operational systems, dramatic changes to the supply chain as well as reductions in MoD scientific personnel, they responded in 1989 with the publication of a new draft interim standard 00-55 (MoD 1989). This used expertise from the nuclear and aerospace industry, MoD and elsewhere to develop a market leading standard around the requirements for mathematically formally verified software and statistical testing.

It was soon realised – in part because of the attempt to classify systems as non-safety critical and outside the remit of 00-55 – that a wider system standard was needed. This led to Def Stan 00-56 (MoD 1991). There was considerable work to take into account strong industry and trade association comments (led by the DTI that developed a detailed trace from all comments to the final issue of the standard).

In parallel with the development in the defence sector, the HSE led the production of the IEC generic standards that became known as IEC 61508 (IEC 1998). Draft publications (IEC 1993) emerged in the early 1990s sharing much in common with the defence standards but addressing a wide range of systems and safety criticalities. During their prolonged drafting they developed detail, consistency and international recognition. However the technical basis of their software aspects remained fixed. The software techniques guidance in IEC 61508 and its software engineering approach was essentially just an extension and internationalisation of EWICS guidance on techniques (Bishop 1990). There are still a number of technical difficulties in IEC 61508 (e.g. how SILs are used) and it lacks a requirement for a safety case.

Around 1993 the limitations to the claims that could be justified by testing were investigated by NASA (Butler and Finelli 1993), and similar results, involving testing and other evidence, were published by Littlewood and Strigini (1993). The 10^{-4} limit was one set by pragmatics of testing technology, but did not include the assumption doubt that we might now make explicit (Bloomfield and Littlewood 2007).

In 1997 the 1991 Interim MoD Def Stan 00-55 was revised to become a full standard and became the first standard to explicitly require a software safety case. This was a radical departure from previous standards but offered some flexibility in the justification of the software, important in view of industry comments on the interim standard. The key features of the revised standard were:

- Deterministic reasoning and proof
- Statistical testing
- Importance of a range of attributes (not just correctness)
- Multi-legged arguments and associated metaphors (belt and braces rather than a wing and a prayer)
- Safety cases and reports
- Sound process to provide trustworthy evidence
- Systematic approach and clarity of roles and responsibilities and other recommendations to reduce project risks
- Evidence preferences: *deterministic* evidence is usually to be preferred to statistical; *quantitative* evidence is usually to be preferred to qualitative; *direct* evidence is usually to be preferred to indirect

The nuclear expertise was influential in Def Stan 00-55. As with many standards and guidelines 00-55 grappled with how to treat software of lower criticalities: at one extreme everything is required and at the other a minimum set of good practices. Populating the regions in between has been problematic and largely a product of the standards process rather than the scientific one. More recently 00-55 has become part of a reissued 00-56 (MoD 2004) and no longer contains software integrity levels.

Adelard had an important role in the development of the defence standards and drafted the safety case requirements. The origins of the work go back to the individuals' involvement (Bloomfield, Bishop and Froome) in the days of the Public Inquiry into the Sizewell B Primary Protection System (CEGB 1982). The work is similar to the approach used by Toulmin (1958) although developed somewhat independently. The concepts were first documented in the EU SHIP project and the work was taken up within a UK nuclear research programme. This led to the first software safety case publication and, in 1998, to ASCAD (Bloomfield et al. 1998, Bishop and Bloomfield 1998), a safety case development manual (still the only one). ASCAD provided the now customary definition of a case as 'a documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment'. In addition to the Adelard work there was research being done at York University (McDermid 1994) that later led to the Goal Structuring Notation described in Kelly's PhD (Kelly 1998).

The ASCAD manual incorporated, with permission, considerable work from the UK nuclear research programme:

- On long-term and safety case maintenance
- How to address specific design issues, even the work on reversible computing
- The work on worst case reliability bounds
- Field experience collected from a range of projects and also used in the SOCS report (ACSNI 1997)
- On argument architecture based on analogies and analysis of PWR pressure vessel cases (Hunns and Wainwright 1991, CEGB 1982).

It also made use of nuclear work on safety culture and work from REAIMS on organisational learning and human factors (Bloomfield et al. 1998).

In 1995 the Advisory Committee on the Safety of Nuclear Installations (ACSNI)[1] set up the Study Group on the Safety of Operational Computer Systems with the following terms of reference:

- to review the current and potential uses of computer systems in safety-critical applications;
- to consider the implications for the nuclear industry;
- in this context, to consider developments in the design and safety assessment of such computer-based systems, including other aspects of control systems; and
- to advise ACSNI where further research is necessary.

The report from this group (ACSNI 1998) addressed the broad principles upon which the evidence and reasoning of an acceptable safety case for a computer-based, safety-critical system should be based. It also discussed, but did not attempt to cover in detail, the extent to which the UK nuclear industry already accepts these principles in theory, and the extent to which they act on them in practice. It made a number of recommendations on regulatory practice, safety cases, computer system design and software engineering, standards, and research.

2 Current Practice in Software Safety and Regulation

The justification that a system is fit for purpose (and continues to be fit for purpose as the environment, use and implementation change) is a complex socio-technical process. In safety regulation in general there has been a widespread adoption of safety case regimes. The Robens Report (Robens 1972) and the Cullen Inquiry (Cullen 1990) were major drivers behind the UK regulatory agencies exploring the benefits of introducing goal-based regulations. The reports noted several shortcomings with prescriptive safety regulations: that is regulations that provide a strict definition of how to achieve the desired outcome.

Firstly, with prescriptive regulations, the service provider is required only to carry out the mandated actions to discharge his legal responsibilities. If these actions then prove to be insufficient to prevent a subsequent accident, it is the regulations and those that set them that are seen to be deficient. Thus safety can be viewed as the responsibility of the regulator and not the service provider whose responsibility, in law, it actually is.

Secondly, prescriptive regulations tend to be a distillation of past experience and, as such, can prove to be inappropriate or at worst to create unnecessary dangers in industries that are technically innovative.

Thirdly, prescriptive regulations encode the best engineering practice at the time that they were written and rapidly become deficient where best practice is

[1] Became the Nuclear Safety Advisory Committee (NuSAC) and then disbanded.

changing e.g. with evolving technologies. In fact it is quite probable that prescriptive regulations eventually prevent the service provider from adopting current best practice.

Another driver for adopting goal-based regulation, from a legal viewpoint, is that overly-restrictive regulation may be viewed as a barrier to open markets. Various international agreements, EC Directives and Regulations are intended to promote open markets and equivalent safety across nations. Whilst it is necessary to prescribe interoperability requirements and minimum levels of safety, prescription in other areas would defeat the aim of facilitating open markets and competition.

Finally, from a commercial viewpoint, prescriptive regulations could affect the cost and technical quality of available solutions provided by commercial suppliers. So there can be clear benefits in adopting a goal-based approach as it gives greater freedom in developing technical solutions and accommodating different standards.

A system safety case is now a requirement in many safety standards and regulations. Explicit safety cases are required for military systems, the off shore oil industry, rail transport and the nuclear industry. For example, in the UK a nuclear safety case must demonstrate, by one or other means, the achievement of ALARP. In the Health and Safety Commission's submission to the Government's 'Nuclear Review'[2] a Safety Case is defined as 'a suite of documents providing a written demonstration that risks have been reduced to ALARP. It is intended to be a living dossier which underpins every safety-related decision made by the licensee.'

The system safety case of course varies from sector to sector. The core of a nuclear system safety case is (i) a deterministic analysis of the hazards and faults which could arise and cause injury, disability or loss of life from the plant either on or off the site, and (ii) a demonstration of the sufficiency and adequacy of the provisions (engineering and procedural) for ensuring that the combined frequencies of such events will be acceptably low. Safety systems will feature amongst the risk reducing provisions comprised in this demonstration, which will thus include qualitative substantiations of compliance with appropriate safety engineering standards supplemented (where practicable) by probabilistic analyses of their reliabilities. Other techniques which may be used for structuring the safety case include fault and event tree analysis, failure mode and effects analysis (FMEA) and hazard and operability studies (HAZOPS).

The safety case, particularly for computer based systems, traditionally contains diverse arguments that support its claims. These arguments are sometimes called the 'legs' of the safety case and are based on different evidence. Just as there is defence in depth in employing diversity at system architecture level, so we see an analogous approach within the safety case itself. Another important feature of the safety case process is independent assessment. The objective of independent assessment is to ensure that more than one person or team sees the evidence so as to overcome possible conflicts of interest and blinkered views that may arise from a single assessment. The existence of an independent assessor can also motivate the

[2] The review of the future of nuclear power in the UK's electricity supply industry.

assessed organisation. The relationship between independent assessment and 'legs' can however be complex.

Safety cases are important not only to minimise safety risks but also to reduce commercial and project risks. In industries such as nuclear, the need to demonstrate safety to a regulator can be a major commercial risk.

So to sum up, the motivation for a safety case is to:

- provide an assurance viewpoint that demonstrates that safety properties are satisfied and risks have been satisfactorily mitigated
- provide a mechanism for efficient review and the involvement of all stakeholders
- provide a focus and rationale for safety activities
- demonstrate discharge duty to public and shareholders
- allow interworking between different standards and support innovation.

So in a safety case the emphasis should be on the behaviour of product not just the process used to develop it: a useful slogan is 'What has been achieved not how hard you have tried'.

3 Uptake and Development of the Safety Case Approach

The incorporation of software safety case requirements in the defence standards drove interest in safety cases, and other forms of assurance case. A generalisation of the safety case concept also appears in Def Stan 00-42 Part 3 (MoD 2008), on the reliability and maintainability case, and in Part 2 (MoD 1997), which deals with the software reliability case. Similar requirements appear in equivalent NATO standards. Adelard has developed and marketed a supporting tool for safety cases (ASCE) and published a supporting methodology in the ASCAD manual. The University of York was also active in developing the safety case approach, such as the use of contracts to modularise safety cases (Fan and Kelly 2004) and safety case patterns (Kelly and McDermid 1997). Much of the work on safety cases and the supporting research is not published and this is becoming increasingly an issue. By their nature safety cases are sensitive for a variety of reasons (security, confidentiality, sensitivities) and not many are available in the public domain. Some safety case work has been published by the University of York (e.g. Chinneck 2004), some anonymised cases are available from Adelard, and John Knight maintains a list of some public cases (Virginia 2009). There is also useful briefing material at (Bloomfield et al. 2002) on safety and (Lipson 2008) on assurance cases. There are also some safety case templates available for UK defence projects.

Goal-based software safety cases have seen take up and interest shown from other sectors. In 1998 the UK CAA Safety Regulation Group drafted a goal based approach to the regulation of air traffic management systems and its proposals are contained in CAP 670 SW01, *'Regulatory Objectives for Software Safety Assur-*

ance in Safety Related ATS Equipment' (CAA 2009). This has gone through a number of iterations. Proposals from Eurocontrol (Eurocontrol 2003) incorporate similar top level goals to CAP 670 SW01 and there is a guidance document from Eurocontrol on safety cases along with some examples and an introduction to GSN (Eurocontrol 2006).

The idea of a case has also applied in areas outside the safety arena. In the medical domain there is considerable work on trust cases (Gorski 2004) for IT systems and there is an International Working Group on Assurance Cases (for Security) (Bloomfield et al. 2006). In terms of security, the uptake by the US DHS of Assurance Cases is significant (Lipson 2008) as is their sponsorship of the draft international standard ISO/IEC 50126. The whole issue of evidence based approaches is receiving considerable international interest as indicated by the US NAS study (Jackson et al. 2007).

There is also work in validating simulation by the use of 'cases' – that is whether one can trust the results of a simulation for a new system. This is led by SE Validation, a small UK company.

4 Current Practice in Safety Cases

Our early definition of a safety case (Bloomfield et al. 1998) was

> 'a documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a given environment'

More recent definitions (e.g. in the revised Def Stan 00-56) make explicit the concept of structured argumentation

> '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'

Current safety case practice makes use of the basic approach that can be related to the approach developed by Toulmin (1958) where claims are supported by evidence and a 'warrant' that links the evidence to the claim. There are variants of this basic approach that present the claim structure graphically such as Goal Structuring Notation (GSN) (Kelly and Weaver 2004) or Claims-Argument-Evidence (CAE) (Bloomfield et al. 1998). GSN is the dominant approach in the UK defence sector. These notations can be supported by tools (McDermid 1994, Emmett and Cleland 2002) that can help to create and modify the claim structure and also assist in the tracking of evidence status, propagation of changes through the case, and handling of automatic links to other requirements and management tools. However the actual claim decomposition and structuring is normally very informal and argumentation is seldom explicit. In practice, the emphasis is on communication and knowledge management of the case, with little guidance on what claim or claim decomposition should be performed.

Toulmin's scheme addresses all types of reasoning whether scientific, legal, aesthetic or management. The CAE style is much more like Toulmin's where we articulate and elaborate textually and yet retain the overall structure. The philosophical approach is that context and assumptions are often rich and complicated and best captured in narrative. A purely graphical rendering would be simplistic and verbose and would certainly go against the spirit of Toulmin in that reasoning can rarely be reduced to just a flow chart or logic network.

The 'case' and associated supporting tools can be seen as having two main roles:

Reasoning and argumentation. As an over-arching argumentation framework that allows us to reason as formally as necessary about all the claims being made. Here there are two very different viewpoints: the one that sees argumentation as primarily a narrative and the other where we seek to model judgements in a formal framework. There are some hybrid approaches where the case can be seen to integrate and communicate a selection of formal analyses and evidence, e.g. it would not seek to reason formally about the timing of a component but leave that to a separate analysis. The balance between these two approaches should be part of ongoing research.

Negotiation, communication, trust. As a boundary objective between the different stakeholders who have to agree (or not) the claims being made about the system. To this end it has to be detailed and rigorous enough to effectively communicate the case and allow challenges and the subsequent deepening of the case.

4.1 Safety Case Structures

One approach that we used in Adelard is to explain safety assurance in terms of a 'triangle' comprising:

- The use of accepted standards and guidelines.
- Justification via a set of claims/goals about the system's safety behaviour.
- An investigation of known potential vulnerabilities of the system.

This is illustrated in Figure 1 below.

The first approach is based on demonstrating compliance to a known safety standard. This is a common strategy, for example the Emphasis tool (Smith and Stockham 2007) produced by the nuclear industry supports an initial assessment of compliance with IEC61508.

The second approach is goal-based – where specific safety goals for the systems are supported by arguments and evidence at progressively more detailed levels. This would typically be implemented using Claims-Argument-Evidence (CAE) or goals-structuring notation (GSN) notations.

The final approach is a vulnerability-based argument, where it is demonstrated that potential vulnerabilities within a system do not constitute a problem. This is

essentially a 'bottom-up' approach as opposed to the 'top-down' approach used in goal-based methods.

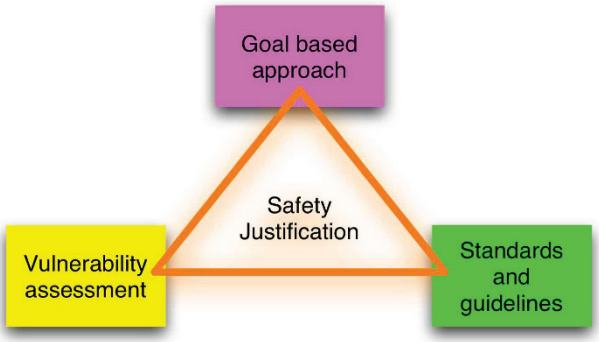

Fig. 1. The Safety Case Triangle

These approaches are not mutually exclusive, and a combination can be used to support a safety justification, especially where the system consists of both off-the-shelf (OTS) components and application-specific elements.

On behalf of the UK nuclear industry we have also been developing a more rigorous approach to claim decomposition. While the details have yet to be published this has involved an empirical study of a large number of safety cases available to Adelard (and reflecting the take up of the cases approach) allowing an informal empirical study of what is needed for a claims decomposition language.

The key technical concept behind the work is the idea that claims decomposition can be formalised in a rigorous way to demonstrate that the decomposition is complete, i.e. that the sub-claims actually do demonstrate the higher claim. This demonstration of completeness requires an extra 'side-condition' to the set of sub-claims that also need to be demonstrated to be correct. The soundness of the approach was established using the PVS proof tool. In practice a user need only choose the type of claim decomposition from the set of sound options. Demonstration of the 'side-condition' could be implemented informally via a checklist derived from the formal analysis, but in principle, it could be proved formally via a formal model of the system under consideration (e.g. to show that the timing of sub-components is additive).

An analysis of actual safety cases from a range of industries showed that only a limited number of decomposition strategies were used. The claim decompositions that we have identified empirically and then formalised are shown in Table 1 below.

Table 1. Formal Claim Decompositions (to be published)

Main types – keywords	Comment
architecture	splitting a component into several sub-components
functional	splitting a component into several sub-functions
set of attributes	splitting a property into several attributes
infinite set	inductive partitioning from a base case (e.g., over time)
complete	capturing the full set of values for risks, requirements, etc.
monotonic	the new system only improves on the old system
concretion	making informal statements less vague

4.2 Confidence, Challenge and Meta-Cases

The structured safety case, either in CAE or GSN notations, needs to be challenged and assessed if we are to be sure that it is fit for purpose. In some areas such as defence and nuclear there is a well defined process for such independent assessment.

The basic measure of efficacy of an argument in this work is the *confidence* that the argument engenders in a dependability claim. Informally here, a dependability case[3] is taken to be some *reasoning*, based upon *assumptions* and *evidence*, allowing certain *confidence* to be placed in a dependability *claim*. For a given claim (e.g. *pfd* is smaller than 10^{-3}), the confidence – and its complement, *doubt* – will depend upon confidence/doubt in the truth of assumptions, in correctness of reasoning, and in 'strength' of evidence.

A key notion here is the recognition that there is uncertainty involved in the assessment of system dependability: it is (almost) never possible to claim with certainty that a dependability claim is true. In the jargon this kind of uncertainty is called *epistemic* (Littlewood and Wright 2007). It concerns uncertainty in an expert's 'beliefs-about-the-world'. It contrasts with the more common *aleatory* uncertainty, which deals with 'uncertainty-in-the-world': e.g. uncertainty about when a software-based system will fail next. It is now widely accepted – even for software-based systems – that the latter is best measured using probabilities, such as probability of failure on demand.

In this work, probability is used to capture the epistemic uncertainty involved in the dependability case: it is the confidence in the truth of the dependability claim. An important part of the work investigates how effective multi-legged arguments are increasing such confidence (i.e. over and above the confidence that would arise from one of the legs alone). This work has been published in the open literature (Bloomfield and Littlewood 2003, Littlewood and Wright 2007).

[3] This is usually a *safety* case, but the ideas apply more generally to other dependability attributes such as reliability and security.

The results from this work concerning the efficacy of multi-legged arguments are, at first glance, not surprising. For example, it is shown that:

- there are benefits from the use of multi-legged arguments, compared with the single legs (the work only treats 2-legged arguments so far); and
- these benefits fall short of what could be expected if arguments 'failed' independently (e.g. if you have two argument legs, for each of which you obtain 10% doubt in the dependability claim, then you cannot expect your doubt to fall to 1% when they form the legs of a '1-out-of-2' argument).

But the work is more interesting than these bald results suggest, in two ways.

Firstly, the formal probabilistic treatment of confidence in claims is novel. It treats rigorously what is often ignored, or treated very informally, even in safety critical standards. It could be used to satisfy the recommendation arising from the SOCS report (ACSNI 1997) that an ACARP (As Confident As Reasonably Practical) principle be introduced into safety cases. So far, however, only theoretical modelling work has been done and its practicality needs to be proved on real safety cases, or realistic case studies. There also needs to be further work on how the formalism might fit into current regulatory practice (Bloomfield and Littlewood 2007).

Secondly, the detailed study of idealised safety cases, as in (Littlewood and Wright 2007), demonstrates how there can be subtle and non-intuitive interactions between the different – and usually disparate – components of a safety case. Although this example concerns a multi-legged case, the insights apply to any safety case in which confidence in a dependability claim rests upon disparate evidence (and this is, essentially, always). The BBN (Bayesian belief network) methodology used in this work – which retains a complete analytical description of the uncertainty – seems much more powerful than the usual numerical BBN treatments.

This work is not yet in a state where it could be taken up as the basis for tools to be used to help build safety cases. More work is needed in several areas – e.g. on the difficult problems of eliciting probabilistic beliefs from experts for input to the Bayesian analyses. On the other hand, it has given some novel insights and it points the way toward more rigorous ways of constructing quantitative probabilistic safety cases.

Current practice regarding confidence in often very pragmatic (e.g. 'traffic lighting' of evidence nodes in a graphical case).

4.3 Other Research

In addition to the work cited above, there has been a variety of other research into safety cases: work on modularity within the avionics sector (Bloomfield et al. 2002), work sponsored by HSE on assessing Software of Uncertain Pedigree (SOUP) (Jones et al. 2001, Bishop et al. 2002b, Bloomfield and Littlewood 2007, Bishop et al. 2002a, ACSNI 1997) and US work on fallibility and other issues

(Greenwell et al. 2006). A large amount of research has been sponsored by the nuclear industry, particularly in the UK.

Within the European nuclear industry, the Cemsis project – Cost Effective Modernisation of Systems Important to Safety (2001-4) – sought to *maximise safety* and *minimise costs* by developing common approaches within the EU to the development and approval of control and instrumentation systems that are regarded as 'systems important to safety' (SIS) that use modern commercial technology. The project had close contacts with the Task Force on Licensing Safety Critical Software of the Nuclear Regulator Working Group (NRWG) of the EU DG for Energy and Transport. The main results of the project are guidance documents on a proposed formal approach to safety justification of SIS (Courtois 2001), on requirements engineering for SIS and a qualification strategy for 'Commercial Off-The Shelf' (COTS) or 'pre-existing' software products. These were evaluated in a number of industrial-based case studies including a 'public domain' example that was used to explain and illustrate the guidance. The approach to new build in the UK specifically distinguishes claims, argument and evidence in the licensing requirements for the Generic Design Assessment for new reactors.

4.4 Specific Tool Support

Tool support for safety cases can be considered in three broad categories:

Decision Support and Elicitation Tools. These allow one to expose the thinking behind the argument, advise on how to construct a case, and assist in reading and review. The most commonly deployed tool specific to graphical safety cases is the Adelard ASCE tool. There is considerable research and development of alternative types of tool and integration with different environments. There is currently standardisation effort with the OMG on claims-argument-evidence and this should provide a good foundation for interoperability of tools and longevity of case documentation.

Tools to Generate Evidence. These provide the evidence that support the safety case argument. They include safety analysis tools (fault trees, FMECAs), tools for collecting and analysing field experience, static analysis , test and proof tools.

Safety Management System Infrastructure Support. In this category there are the tools for configuration management and traceability such as Requirements Engineering support tools and Hazard Logs.

5 Future Directions

Based on our review of past and current work on computer-based cases, we can identify a number of directions for the future development of cases.

5.1 Safety Case Methodology Enhancement

A lot of the current research has been focused on notation and structuring, but far less on how to develop a safety case and what arguments to deploy (Bishop and Bloomfield 1998, Eurocontrol 2006). We also have to recognise that safety cases are costly to develop, so we should seek more efficient means of construction. So there is scope for far more work in this area, including:

- Development of industry and sector specific argument 'templates' and linkage to sector standards and generic standards such as IEC 61508.
- Development of cases for specific classes of system for less critical and ultra critical systems (Littlewood 2000).
- Strategies for justifying COTS components within an overall safety justification.

5.2 Extension to Other Areas

Safety case concepts can be used in other areas that require assurance. There are a range of systems (e.g. for finance or communications) which are critical parts of the infrastructure where loss could have severe impacts on society. Assurance cases have been used to a limited degree, but may well be used more widely in future.

Also, as systems become more distributed and interconnected, there is an increasing need to include security and other attributes with the assurance case together with the incorporation of threat assumptions that include consideration of deliberate attacks as well as random events.

As part of this process, we need to extend our view of the 'system' we are seeking to assure. In the early days, the focus was on the technical system (hardware, software, sensor and actuators). In the future we need to think about the larger *socio-technical* system that includes the management, people and processes that interact with the technical system.

5.3 Safety Case Structuring

There is further work needed on structuring a case. This includes:

- more rigorous methods for claim decomposition
- modularisation of safety cases (so that safety arguments for subsystems can be re-used)
- the use of diverse arguments and evidence
- understanding and exploiting the relationship between the argument structure and the architecture of a system
- ensuring that the case is understandable by all stakeholders

We note that the use of diverse arguments and diverse evidence can help enhance confidence in a claim (discussed in the next section), but more work is needed on the integration of such evidence (like operational experience; statistical testing, formal proof, and process evidence) to support specific claims (such as reliability).

We also need to work on 'stopping rules', i.e. when to stop expanding claims. This is probably related to the degree of confidence the claim is correct without the need for any further supporting evidence.

5.4 Confidence and Challenge

Safety cases are open to challenge at a number of levels, such as the applicability of the arguments and the credibility of the evidence. Currently confidence is expressed in simplistic terms (e.g. using traffic lights) but it is not clear how a lack of confidence will propagate through to higher level claims. The means of expressing confidence in different aspects of the case requirements and modelling the impact on the top-level claims needs more formality and rigour.

There are also pragmatic issues of how such challenges and rebuttals are accommodated with the case. If they are included as nodes in the overall case, this can become very cluttered. It may be desirable to construct a 'meta-case' linked to the main case that includes such material, e.g. to justify the claim decomposition and the credibility of the evidence.

6 Concluding Remarks

The state of the art of safety cases for computer based systems has to be addressed within the context of regulation and system level approaches to safety. A structured approach to safety cases for computer based systems has been developed that addresses both the reasoning that safety properties are satisfied as well as pro-

viding an effective approach to communicating this reasoning. The acceptance of a case is (or should be), in the end, a social process.

The use of goal-based, structured cases is very appealing, supporting as it does innovation and flexibility but as can be seen from this review much work is needed to develop a case and put it on a convincing footing. While the basis of Toulmin's scheme is really very simple the industrialisation and application to complex systems is a significant undertaking. Our current approaches rely very heavily on the expertise and best practice of the community and the challenge and review that cases receive. The work is normally not published as there are sensitivities in most real cases and even the research that is being done is not well represented in the literature. This paper has attempted to identify public domain sources of information for those interested in the field. We hope that more will be made available during the year.

Acknowledgments The authors wish to acknowledge the support given by the UK Control and Instrumentation Nuclear Industry Forum (CINIF) research programme, the UK Health and Safety Executive research programme, the EU Environment programme (sub-theme Major Industrial Hazards) and the EU nuclear research programme who funded some of the research presented in this paper.

Disclaimer The views expressed in this paper are those of the authors and do not necessarily represent the views of the research sponsors.

References

ACARD (1986) Software: a vital key to UK competitiveness. Advisory Council on Applied Research and Development. HMSO

ACSNI (1997) The use of computers in safety-critical applications. Final Report of the Study Group on the Safety of Operational Computer Systems (SOCS) constituted by the Advisory Committee on the Safety of Nuclear Installations. HSE Books, London

Bishop PG (ed) (1990) Dependability of critical computer systems 3. Elsevier Applied Science

Bishop PG, Bloomfield RE (1995) The SHIP safety case. In: Rabe G (ed) Proc SafeComp 95, 14th IFAC Conf on Computer Safety, Reliability and Security, Belgirate, Italy

Bishop PG, Bloomfield RE (1998) A methodology for safety case development. In: Redmill F, Anderson T (eds) Industrial perspectives of safety-critical systems. Springer-Verlag

Bishop PG, Bloomfield RE, Clement TP, Guerra ASL (2002a) Software criticality analysis of COTS/SOUP. SAFECOMP 2002, Catania, Italy

Bishop PG, Bloomfield RE, Froome PKD (2002b) Justifying the use of software of uncertain pedigree (SOUP) in safety related applications. 5th Int Symp Programmable Electronic Systems in Safety Related Applications, Cologne

Bloomfield RE (1990) SafeIT, the safety of programmable electronic systems: a government consultation document on activities to promote the safety of computer-controlled systems. Department of Trade and Industry

Bloomfield RE, Brazendale J (1990) SafeIT2, standards framework. Department of Trade and Industry

Bloomfield RE, Littlewood B (2003) Multi-legged arguments: the impact of diversity upon confidence in dependability arguments. Proc DSN 2003. IEEE Computer Society

Bloomfield RE, Littlewood B (2007) Confidence: its role in dependability cases for risk assessment. Intl Conf Dependable Systems and Networks, Edinburgh, IEEE Computer Society

Bloomfield RE, Bishop PG, Jones CCM, Froome PKD (1998) ASCAD – Adelard safety case development manual. Adelard

Bloomfield RE et al (2002) Safety cases for PES. Adelard. http://www.adelard.com/web/hnav/resources/iee_pn/index.html. Accessed 17 October 2009

Bloomfield RE, Guerra S, Miller A et al (2006) International Working Group on Assurance Cases (for Security). IEEE Secur Priv 4:66-68

Butler R, Finelli G (1993) The infeasibility of quantifying the reliability of life-critical real-time software. IEEE Trans Software Engineering 19:3-12

CAA (2009) CAP 670 Air traffic services safety requirements, SW01 regulatory objectives for software safety assurance. Civil Aviation Authority Safety Regulation Group

CEGB (1982) Sizewell B preconstruction safety report. Central Electricity Generating Board

Chinneck P, Pumfrey DJ, Kelly TP (2004) Turning up the HEAT on safety case construction. In: Redmill F, Anderson T (eds) Practical elements of safety. Springer-Verlag

Courtois PJ (2001) Semantic structures and logic properties of computer-based system dependability cases. Nucl Eng Des 203:87-106

Cullen (1990) The public inquiry into the piper alpha disaster. HMSO Cm 1310

Emmet L, Cleland G (2002) Graphical notations, narratives and persuasion: a pliant systems approach to hypertext tool design. In: Proc ACM Hypertext, College Park, Maryland, USA

Eurocontrol (2003) ESARR6 Software in ATM systems.

Eurocontrol (2006) Safety Case Development Manual. http://www.eurocontrol.int/cascade/gallery/content/public/documents/safetycasedevmanual.pdf. Accessed 17 October 2009

Fan Y, Kelly T (2004) Contract-based justification for COTS component within safety-critical applications. Proc 9th Australian workshop on safety critical systems and software, Brisbane

Gorski J (2004) Trust Case – a case for trustworthiness of it infrastructures. In Proc NATO Advanced Research Workshop on Cyberspace Security and Defence: Research Issues, Gdansk, Poland

Greenwell WS, Knight JC, Holloway CM, Pease J (2006) A taxonomy of fallacies in system safety argument. 24th International System Safety Conference, Albuequerque

HSE (1987) Programmable electronic systems in safety related applications. Health and Safety Executive

HSE (1993), Out of control – a compilation of incidents involving control systems. Health and Safety Executive (draft document)

IEC (1986) IEC 880 Software for computers in the safety systems of nuclear power stations. International Electrotechnical Commission

IEC (1993) Functional safety of electrical/electronic/programmable electronic systems: generic aspects. Part 1: General requirements. Draft standard from IEC Sub-Committee 65A: System Aspects, Working Group 10. International Electrotechnical Commission

IEC (1998) Functional safety of electrical, electronic, and programmable electronic safety related systems. IEC 61508, Parts 1 to 7, 1998 to 2000. International Electrotechnical Commission

IEE (1989) Software in safety related systems. The Institution of Electrical Engineers and the British Computer Society

Jackson D, Thomas M, Millett LI (eds) (2007) Software for dependable systems: sufficient evidence? Committee on Certifiably Dependable Software Systems, National Research Council

Jones C, Bloomfield RE, Froome PKD, Bishop PG (2001) Methods for assessing the safety integrity of safety-related software of uncertain pedigree (SOUP). HSE Contract Research Report CRR 337/2001. Health and Safety Executive

Kelly TP (1998) Arguing safety: a systematic approach to managing safety cases. PhD thesis, University of York

Kelly T, McDermid J (1997) Safety case construction and reuse using patterns. Proc 16th Conf on Computer Safety, Reliability and Security (Safecomp '97)

Kelly TP, Weaver RA (2004) The goal structuring notation – a safety argument notation. Proc Dependable Systems and Networks Workshop on Assurance Cases

Lipson H (2008) Assurance cases overview. US Department of Homeland Security. https://buildsecurityin.us-cert.gov/daisy/bsi/articles/knowledge/assurance/641-BSI.html. Accessed 17 October 2009

Littlewood B (2000) The use of proofs in diversity arguments. IEEE Trans Softw Eng 26:1022-1023

Littlewood B, Strigini L (1993) Assessment of ultra-high dependability for software-based systems. Comm ACM 36:69-80

Littlewood B, Wright D (2007) The use of multi-legged arguments to increase confidence in safety claims for software-based systems: a study based on a BBN of an idealised example. IEEE Trans Softw Eng 33:347-365

McDermid JA (1994) Support for safety cases and safety argument using SAM. Reliab Eng Syst Saf 43:111-127

MoD (1989) Draft Interim Def-Stan 00-55, the procurement of safety critical software in defence equipment. Ministry of Defence

MoD (1991) Interim Def-Stan 00-56, hazard analysis and safety classification of the computer and programmable electronic system elements of defence equipment. Ministry of Defence

MoD (1997) Def Stan 00-42 Reliability and Maintainability (R&M) assurance guide, Part 2 Software. Ministry of Defence

MoD (2004) Def Stan 00-56 Safety management requirements for defence systems. Issue 3. Ministry of Defence

MoD (2008) Def Stan 00-42 Reliability and Maintainability (R&M) assurance guide, Part 3 R&M Case. Ministry of Defence

Redmill F (ed) (1988) Dependability of critical computer systems 1. Elsevier Applied Science

Redmill F (ed) (1989) Dependability of critical computer systems 2. Elsevier Applied Science

Robens (1972) Safety and health at work. Report of the committee 1970-72. HMSO Cmnd 5034

Smith PR, Stockham R (2007) EMPHASIS – An assessment tool for smart instruments, PRfsS/Moore Industries-Europe, United Kingdom

Toulmin SE (1958) The uses of argument. Cambridge University Press

Virginia (2009) Safety cases repository. University of Virginia Dependability Research Group. http://dependability.cs.virginia.edu/info/Safety_Cases:Repository. Accessed 17 October 2009