SOFTWARE VERIFICATION RESEARCH CENTRE

SCHOOL OF INFORMATION TECHNOLOGY

THE UNIVERSITY OF QUEENSLAND

Queensland 4072
Australia

TECHNICAL REPORT

No. 99-42

Improving Safety Management in Defence Acquisition

Brenton Atchison, Peter Lindsay, Tony Cant

December 1999

Phone: +61 7 3365 1003
Fax: +61 7 3365 1533

To appear in *Proceedings, 4th Australian Workshop on Safety Critical Systems and Software*, ed. M. McNichol, Australian Computer Society, 1999, pages 1-8.

Improving Safety Management in Defence Acquisition

Brenton Atchison[1], Tony Cant[2] and Peter Lindsay[1]

[1] Software Verification Research Centre
School of Information Technology
The University of Queensland
Queensland 4072, Australia

[2] Defence Science Technology Organisation

brenton@svrc.uq.edu.au, pal@svrc.uq.edu.au, Tony.Cant@dsto.defence.gov.au

Abstract

Improved management of safety during procurement of computer-based safety-critical systems is one of the key concerns of the Australian Government's Department of Defence Software Acquisition Reform program. This paper reports some lessons learnt from a task currently being undertaken by the Defence Acquisition Organisation, Defence Science Technology Organisation (DSTO) and the Software Verification Research Centre (SVRC) to study management of safety and implementation of software safety standards in a range of defence projects. Effective safety management requires the identification of potential issues, and planning for their resolution, early in the procurement process. The lessons learnt are general and may be of benefit to other organisations involved in specification and acquisition of safety-critical software systems.

1. Introduction

1.1 Defence acquisition reform and safety

As part of the Department of Defence's Software Acquisition Reform program, the SVRC and DSTO are undertaking a number of tasks in support of the acquisition of computer-based Safety-Critical Systems. The project (known as 'DefSafe') aims to improve Defence acquisition processes through a mix of research, project-based consultancy and policy guidance.

Advice to defence projects is an important part of DefSafe. Projects with which we have been involved include: a ship-launched missile-decoy system, a digital display system for fighter training aircraft, an air-combat training system and a ship command-and-control system. (This paper will not address details of specific projects). Many projects are at an early stage of procurement, while others have already undergone substantial development but a detailed safety case has not yet been formulated. Advice to projects covers the use of safety standards, the safety management process, preliminary hazard analysis and so on.

Policy advice is centred on an extensive survey of the relative strengths and weaknesses of international safety standards and a companion study of how Defence currently invokes standards in its existing contracts. The DefSafe team is also providing advice to Defence on the general approach to procurement and on certification and regulation issues.

The research aspects of the DefSafe Project are focused on further development of the Australian Standard DEF (AUST) 5679 [1], written by DSTO and recently published by the Army Engineering Agency. This Standard provides requirements and guidance for the development and assessment of safety-critical computer-based systems, focusing on requirements for the system safety case. Under DefSafe, research will be carried out into a number of technical issues, including the impact of human factors on safety and the integration of Commercial Off-The-Shelf (COTS) items and other non-development items into safety-critical systems.

1.2 Lessons learnt

The work presented in this paper draws on DefSafe experiences and observation of overseas (primarily US and UK) defence practices. It reinforces our previous experiences with the implementation of safety programs across a range of different industries.

Many different technical and management issues can affect the success of a project's safety program. Perhaps the most important issue is the timely recognition of possible safety implications of the system under procurement – be they possible hazards for operators, other defence staff, allies, or even civilians. These safety

implications are not always immediately obvious. All too often, safety is considered only late in the procurement lifecycle, by which time it may be too late, or too expensive, to implement effective risk mitigations. This often results in sacrifice of system functionality or imposition of undesirable operational constraints.

This paper presents some of the issues that we have observed in the implementation of safety programs across a broad range of defence projects, at different stages of the development lifecycle. Some of the issues have well-known solutions while others are symptoms of the inherent difficulty of analysing, designing, implementing and assuring safe computer-based systems. In any case, it is hoped that other projects – in defence and other application domains – can benefit from the mistakes of the past by taking timely and appropriate management action.

The paper is organised as follows. Section 2 discusses issues that arise in the choice of safety standards to be applied to a project. Section 3 describes issues for the safety management process, and discusses the regulatory and certification context. Section 4 discusses safety management issues, including procurement processes and capabilities. Section 5 describes technical factors that are known to cause difficulties in safety management. Section 6 presents a checklist of issues for consideration in early phases of the procurement lifecycle. Finally, Section 7 presents a summary and conclusions.

2. Choice of safety standards

In a sense, existing Safety standards embody 'best practice' in safety management, yet the range of available safety standards is potentially very confusing, and choosing the best standard for a particular project can be difficult. Here we present some relevant background, then some observations about the choice of safety standards for safety-critical computer-based Defence systems.

2.1 International trends

The SVRC has carried out an extensive survey of international safety standards [2], drawing on existing surveys [3], [4], covering the following standards:

- DEF (AUST) 5679 [1]
- MIL-STD-882C [5]
- NATO StanAgs 4404 [6]and 4452 [7]
- UK Defence Standards 00-54 [8], 00-55 [9] and 00-56 [10]
- ARP Standards 4754 [11] and 4761 [12]
- RTCA/DO-178B [13]
- IEC 61508 [14]

There are some clear trends emerging in these standards.

The *system* nature of safety is clearly recognised. Moreover, standards are now addressing the whole system lifecycle, from concept through to decommissioning. In many cases (such as MIL-STD-882C), there is a broadening of definition of safety so that it covers not simply "life and limb" but includes also equipment damage and threats to the environment.

The concept of a *Safety Case* is central in many safety standards. A Safety Case documents the evidence providing assurance that the system will be safe to operate, and assumptions on which the assurance is based [15], [16].

Emphasis in the past has often been on documenting the quality of the engineering process, but certifiers increasingly require also "product" assurance – details of safety features of the design and evidence of their effectiveness. For example, the UK Ministry of Defence's flight certification authorities focus entirely on product evidence. At the very least, the safety case needs to contain sufficient details of design and identification of safety mechanisms for an independent safety assessment to be made ("transferable assurance").

Standards are increasingly recognising the inherent difficulty of assuring the safety of software-based systems, and encourage system engineers to consider other (simpler or more reliable) means of implementing safety-critical functionality. Standards that address software safety are now deliberately moving away from the use of quantitative (probabilistic) risk assessment in favour of qualitative analysis and evidence of good design practices.

2.2 Defence use of standards

A number of safety standards are used in Defence projects: projects that we are aware of have invoked one or more of the following standards: MIL STDs 882B and 882C, UK Defence Standards 00-55 and 00-56, RTCA DO-178B, DEF (AUST) 5679, and StanAgs 4404 and 4452. Publication of the use of standards is limited [17]. In some cases, standards have been invoked that do not provide sufficient coverage of safety aspects, while in other projects no specific safety standards are used.

Clearly, the approach to procurement of Defence systems within Australia is not a uniform one as far as safety standards are concerned. Special problems can arise where multiple standards are invoked, because of the difficulty in reconciling what can be vastly different approaches to safety management.

3. Safety management framework

Standards alone are not adequate to provide assurance of safety: there needs to be in place an appropriate cross-organisational management framework for safety [18]. We make various observations here, by way of context for the rest of the paper.

3.1 Certification and regulatory authorities

Various organisations exist with the responsibility of certifying safety-critical systems. Within Australia, the Australian Ordnance Council (AOC) provides advice on the safety and suitability for service for ordnance aspects of weapons systems in Defence. The Royal Australian Airforce (RAAF) has a Directorate of Technical Airworthiness, which makes recommendations on airworthiness of piloted aircraft. In the USA, the US Navy's Weapons Systems Explosive Safety Review Board (WSESRB) reviews the safety programs for weapons systems.

In many cases, safety assessment is required prior to acceptance into service. However, for Australian Defence systems there are safety aspects for which no identified authority takes responsibility (for example, for systems outside the scope of the AOC but which are nevertheless safety-critical). Another concern is that, where authorities' jurisdictions overlap, inconsistencies between approaches may result in conflicting requirements.

3.2 Through-lifecycle support

The safety program should not conclude with the delivery of the safety case prior to system operation. Training of system operators, maintenance, "upgrades" and interfacing with new and evolving systems all have possible safety implications.

Sometimes, the procured system will form part of a wider operational system, for example upgrades to an existing aircraft. Typical current practice is simply to assess the "deltas", based on an Engineering Change Proposal. Unfortunately, this often results in an incomplete safety case that becomes harder to maintain – and less credible – the longer the system continues to evolve.

As noted earlier, safety management must extend beyond the procurement lifecycle, into operation, maintenance and modification. These phases typically have more Defence involvement, perhaps with some Developer support in maintenance and modification activities. In the future, Defence faces the challenge of maintaining a continuous safety program throughout these post-procurement phases.

The UK Ministry of Defence is moving towards the creation of a single safety authority covering all forces and both in-acquisition and in-service aspects of safety. However, currently there is no corresponding single authority in the Australian Department of Defence (ADF). Despite the unique requirements of different ADF sectors, Australia would benefit from a more coordinated approach to safety management.

4. Project safety management

4.1 Safety Management Group

Safety management is a responsibility shared by multiple stakeholders, including the Client, Australian Defence Organisation, Developer and Subcontractors, Evaluators, Certifiers and End-users. Safety-critical decisions require ratification from all parties, particularly where trade-offs between cost and risk are made. Effective safety management also requires direct access to information and representation of diverse expertise. Accordingly, most standards call for the establishment of a Safety Management Group involving representatives of all stakeholders.

Some balance must be struck between adequate representation and a workable size. In practice, it is likely that a core group will make the decisions and will invite specialist representation as appropriate.

4.2 Procurement processes and capabilities

The Client's role in the development and maintenance of the safety case is often downplayed or overlooked. In the past, much of the responsibility for safety engineering has rested with the Developer. However, it is increasingly recognised that Defence has a role to play in safety management.

There may be parts of the safety case that only the Client can provide. Client information is particularly important in the risk analysis and assessment activities, which typically require extensive knowledge of the system environment and operational profile, external risk mitigations and rationale for safety targets.

In fact, many standards require the Client to determine risk targets and acceptability of risk. Typically this means defining severity and likelihood categories for system failures, and defining the levels of authorisation required for accepting risk or approving engineering changes. DEF (AUST) 5679 provides pre-determined risk criteria, but other standards (e.g. MIL-STD-882) need "tailoring" by the Client to set risk targets appropriate to the application.

Defence thus needs to have the capability to assess safety throughout development, certainly at Preliminary Design Review and Critical Design Review, and also prior

to authorisation of field-testing. This capability also needs to be transferred into operational support teams. Due to lack of established expertise and experience, on-the-job training is usually required.

4.3 Developing safety assurance

We have observed a number of different models for the development of safety assurance.

In some (typically older) models, a separate team (sometimes even a separate organisation) performs safety analysis, by analogy with Independent Verification and Validation (IVandV). This can be frustrating for the safety analysis team, since their analysis will usually lag behind design, and there may be little opportunity to influence design. Conversely, designers may resent what they see as interference, and may even take a proprietorial position and hinder access to information by analysts.

For these reasons, there are often significant overheads, and considerable risk of schedule delays and cost overruns (or threat to safety integrity). A substantial problem is the transfer of knowledge between teams, with safety analysts missing critical design rationale and explanations, while designers fail to benefit from the design insight gained by analysis.

A related issue is configuration management where design baselines stable enough for external review appear too late in the evolutionary design process to be efficiently modified by analysis.

We have observed that a better, more effective approach is for the two teams to work closely together (or better still, to have a single team) to produce a safety case that is then evaluated by an independent third party. One model is to distribute responsibility of safety analysis and assurance across design teams, with the "safety management team" simply responsible for coordination and quality control.

4.4 Independent safety assessment

For complex or innovative system designs, certification and regulatory authorities generally rely on an independent technical assessment of the safety case. Indeed, many standards explicitly require this, for example the *Evaluator* role in DEF (AUST) 5679. Depending on the standard used, the requirement for independent assessment may range from a peer review within the design team to a review by a separate organisation.

Criteria for independence include financial and management autonomy. This may present problems in a relatively small software system safety community such as exists in Australia. There is frequently a need to put "Chinese walls" in place and to ensure possible conflicts of interest are declared.

The argument for independence is that an independent Evaluator may discover anomalies overlooked by those more familiar with the problem, and that their opinions are less likely to be distorted by other project goals. This means the Evaluator should have limited involvement in decision-making or design but should instead review and assess the result of such decisions.

The extent of assessment effort may range from an audit of processes to a full technical review of program outputs. In some instances, it may also include elements of the safety analysis itself.

5. Technical issues

This paper largely focuses on management issues in procurement of safety-critical systems, but certain technical issues can severely impact a safety program. In part, the issues reflect a general lack of consensus on certain aspects of safety assurance. However, through early identification of possible pitfalls, it may be possible to minimise the possible disruption that these technical issues have on the safety program.

5.1 Process versus product assurance

Many safety standards, following recent trends in quality assurance, prescribe processes to follow in the construction of a safety-critical system in the belief that executing the processes will generate a high integrity product. While there is a broad consensus about the correlation between process and product quality, safety-critical systems additionally require product-based evidence of safety. However, all too often inadequate or inappropriately detailed design information is made available to assessors, resulting in lower assurance than appropriate.

It is necessary to plan a technical argument to present in the safety case based on product assurance. Such an argument will also assist the planning and coordination of the safety program by allowing the results of assurance processes to be focussed on well-understood goals.

5.2 Use of COTS components and reuse

Safety integrity of "Commercial Off the Shelf" (COTS) components is a particularly difficult issue. Sometimes COTS products can be acquired along with evidence of assurance. However, this evidence must be shown to be applicable under the demands of the new operational environment and the regulatory expectations. If appropriate assurance cannot be acquired, the costs involved in "reverse engineering" a safety case for COTS,

and in maintaining the safety case in the face of component upgrades, can outweigh the potential savings of off-the-shelf procurement [19]. A similar consideration applies to reuse of components or software platforms. At present, there are very few cost-effective methods of providing assurance of COTs in safety-critical systems.

5.3 Safety Integrity Levels as a sticking point

Most safety standards employ a notion of *Safety Integrity Levels (SILs)* – sometimes called Development Assurance Levels – to categorise the level of safety assurance required for a system, subsystem or component according to their criticality to safety. The level of assurance is typically defined by the degree of rigour and independence applied during analysis, design, implementation and testing.

Although SILs are an intuitively appealing notion, they are surprisingly difficult to make precise and different standards use different definitions. This can present significant difficulties in combining the use of different standards, or in integrating systems whose components were developed against different standards [20].

The use of SILs also complicates the process of software development planning and cost estimation. One difficulty is that required integrity levels are not firmly known until the system design and safety analysis is in place. However, software development must be costed and planned long before this time. The general solution to this problem we have observed is to plan software development based on assumptions of required integrity levels. Such assumptions may even be formalised in the contract. Subsequent design effort may then reduce the required software integrity to feasible levels. If such decisions are not made prior to the contract, it may become necessary to seek consensus from all stakeholders early (including the Evaluator) to "make the best guess and proceed".

Even if the required integrity level is known, the cost of achieving ultra-high integrity is not currently possible to predict and, in some cases, may be prohibitively high. The assurance techniques required will generally challenge Developer's capabilities and may require substantial modifications to familiar development processes.

To some extent, this is a deliberate act by Certifiers in order to make Developers think twice before implementing critical functions in software. In practice, we can expect to see arguments for reducing SILs, but validity of arguments needs careful examination.

5.4 Balance between testing and analysis

Although most standards highlight the need to generate safety assurance through both analysis and testing, traditional verification processes are often focussed on test activities. There are a number of dangers with this approach.

First and foremost, in many cases the attention paid to identification of safety requirements and safety integrity levels is not sufficiently thorough, or is performed late in development. Careful analysis is needed early in development, so that safety can be designed into the system.

For safety-critical systems it is necessary to validate their safe operation even in the presence of failures. This means being able to inject faults and simulate failure scenarios that may be difficult or even impossible by testing alone. System modelling and analysis can be used to explore scenarios that are too difficult or expensive to construct in the test lab [18].

A balance of analysis and testing may also be necessary for cost optimisation. Implementing test criteria for safety-critical software can be very expensive, for example the Modified Condition/Multiple Decision testing of RTCA/DO-178B. Exhaustive testing is also prohibitive and is not generally achievable in practice, for all but the simplest of components. Software safety analysis can reduce the cost of rigorous testing by identifying critical software components, which demand the most attention, and presenting scenarios as the basis of more intelligent test cases.

Software is typically changing substantially throughout development due to the evolution of requirements and design solutions, and large amounts of testing are also very expensive to repeat in the face of change. Safety analysis can assist the regression testing process by identifying the potential scope of the assurance through knowledge of causal relationships between software elements.

5.5 Technical infrastructure

The following processes are essential for safety management, and should be established early in development. Ideally, solutions will be determined prior to contract signature.

- **Configuration Management**, not just of software and hardware, but also of designs and documentation, tools and test rigs, and all the other artefacts involved in the construction and maintenance of the safety case. The configuration management system should allow for the

identification of safety-critical system elements and their relationship.

- **Safety Incident Reporting** and integration into Engineering Change Management processes. A means of determining the safety implications of engineering changes and modifying safety assurance should be identified.

- **Hazard Log and Safety Data.** One difficulty in management of a safety program is coordination of the information gathered. The Hazard Log forms an index into the Safety Case, in essence summarising the controllable hazard conditions, their risks and mitigations. However, a significant amount of other safety data is generated, through various analyses. This information is generally related and it often needs to be checked for consistency and completeness [15].
  The information system required to manage the hazard log and other safety data needs forethought to get the structuring right. Not only does all information need to be collected concisely, but also relationships between information should be captured. This is especially important where development of the safety case is distributed across multiple organisations. It would also be useful to integrate the safety data with other project data, such as the requirements database.

6. Starting a project safety program

Many of the activities already mentioned, such as choosing a safety standard, constituting a Safety Management Group, defining risk tolerability and development processes, require early attention in the project life cycle. We have observed significant problems arising in development and assessment due to scant attention being paid to these activities in early stages of procurement.

We present a checklist to consider in the project initiation phases to manage the risk of safety programs. The checklist is designed to create design decision-making structures, establish a safety culture, plan an effective safety management process and reduce uncertainty about program scope and cost.

The suggestions are largely independent of the standards selected, although details will change depending on the adopted approach.

6.1 Request For Tender (RFT)

The Client has the ultimate responsibility for procuring a safe system so must be largely responsible for establishing a safety program. This is achieved by setting an appropriate priority on the safety program in the tendering process.

There are a number of activities that should be performed during RFT preparation:

- Appoint a Client Safety Representative.
- Determine legal and certification requirements.
- Construct a Preliminary Hazard List to gauge criticality of the system. If necessary and possible, revise the system concept to eliminate hazards.
- Consider the safety program scope outside of the system to be acquired. In particular, establish the relationship with integrated systems and support requirements.
- If necessary, engage specialist technical support to prepare or review the RFT content.

The RFT should communicate safety program requirements and seek a proposed solution for safety management. In particular, it should:

- Define safety program requirements, including the use of standards.
- Define all safety program deliverables and their format.
- Require that a single contractor have primary responsibility for execution of the safety program (preferably the prime system contractor).
- Request a draft development or engineering plan.
- Request a draft safety management plan, including organisational structure, roles and responsibilities and technical processes.
- Request evidence of ability to execute safety management plan.
- Define special software program requirements for safety-critical software. Include any constraints or assumptions about the software safety integrity.

6.2 Tender response and evaluation

The tender evaluation process should consider the ability to execute a safety program to a level commensurate with the perceived safety risk. Specialist technical support may again need to be engaged to assist the evaluation.

- Ensure that the safety management plan is compliant with RFT requirements and standards.

- Ensure that the safety management plan is feasible and organisation is mature enough to execute the plan.
- Ensure that the safety management plan does not conflict with the development plan.
- Ensure the ability to apply software development and assurance processes, especially for reused software and COTS items.

6.3 Pre-contract activities

The best opportunity for cooperative resolution of safety program risks is following the selection of the preferred tenderer(s) and before contract signature. During this period, a joint Client/Tenderer team should competently identify and assess project risks.

Where the level of uncertainty remains high, a recommended approach is to engage in a funded Mutual Risk Reduction phase. This allows a Preferred Tenderer(s) to engage in paid preliminary activity to analyse project risk and have sound risk management process in place once on contract. This includes those risks attached to executing a safety program.

Since safety management is a cooperative activity, the pre-contract phase is also an excellent opportunity to establish working relationships and instil a cooperative safety culture in the project.

In particular, it is useful to engage in initial workshop-style discussions with all stakeholders to address:

- Risk assessment criteria, including levels of tolerable risk.
- Roles and responsibilities within the safety organisation and initiation of the Safety Management Group.
- Procedures for risk assessment and cooperative risk mitigation.
- The Preliminary Hazard List, including a discussion of hazardous operational scenarios.
- Safety Management Plan and use of standards.
- Role of Evaluator or certification authority.

The information gathered by the workshop can be used to assess the project risk associated with the safety program. Further risk reduction can be achieved during the Mutual Risk Reduction phase. Activities that can be performed during this phase include:

- Production of safety management plan. Additional details should include the technical solution for the safety case and the integration of the safety program with other activities.
  Particular attention paid to relationship with engineering activities. Influence of safety program in review and internal authorisation procedures and formal external reviews.
- Draft Preliminary Hazard Analysis, where the Preliminary Hazard List is systematically refined. An assessment of the risk based on accident scenarios can be performed in collaboration with the Client and Users. This will require recording of any operational assumptions.
  Where possible, the operational concept should be provided to the Contractor prior to Preliminary Hazard Analysis. Alternatively, the Contractor could be involved in the revision or development of the concept.
- Estimation of software criticality based on results of the Preliminary Hazard Analysis and knowledge of system design. Particular attention should be paid to the criticality of COTs in the design solution.
- Planning the software development strategy to mitigate any identified risks, possibly including a prototype software build to refine cost estimations.
- Submission of an evaluation plan corresponding to the Developer safety management plan.

7. Summary and conclusions

We conclude with a summary of the observations and lessons learnt in our study of processes in procurement of safety-critical systems in Defence:

1. Just following a standard will not guarantee safety. Domain expertise, experience and a clear understanding of the standard's intent are all vital.
2. A Safety Case should be required as part of system acquisition. The purpose of a Safety Case is to provide transferable assurance that the system is acceptably safe for service. Arrangements should be made for in-service maintenance of the safety case.
3. Cooperation from all stakeholders is required for effective safety management. The Client (Project Office) has primary responsibility for parts of the Safety Case, including details of the system's intended operational context.
4. Development of a Safety Case is different from IVandV, and is best integrated closely into the system development process.
5. However there should be an independent evaluation of the Safety Case.
6. The Safety Case should be focused on assurance that safety requirements have been adequately identified and addressed in design and test. In particular, the Safety Case should include a technically defensible argument, based on analysis

of system design and observation of system behaviour, that safety risks have been reduced to acceptable levels.

7. The cost of developing and maintaining safety assurance for COTS components may outweigh any purported savings.
8. The project risks of developing safety-critical software should be assessed early in the procurement process. Particular attention should be paid to early estimation of software integrity targets and Developers' capability to meet the integrity assurance requirements.
9. Although both testing and analysis are essential, careful attention should be paid to the most appropriate and cost-effective balance between the two.
10. It is highly desirable to start developing safety arguments early in the project, so appropriate and effective hazard mitigations can be built into the system.
11. The safety program should be fully integrated into development processes and infrastructure, including the configuration management system, incident reporting process and product information system.
12. Risks with the safety program are best addressed in the earliest acquisition phases, before contract signature. If project risks are significant, it is useful to employ a pre-contract Mutual Risk Reduction phase, which includes a Preliminary Hazard Analysis and early consideration of possible risk mitigations.

8. Acknowledgements

We gratefully acknowledge the support, guidance and useful comments of David Marshall and Adrian Pitman (DAO-SWAR), and the contributions of DefSafe colleagues Andrew Hussey, Graeme Smith and Axel Wabenhorst.

9. References

[1] Australian Department of Defence, *DEF (AUST) 5679 The Procurement of Computer-Based Safety Critical systems*: Army Standardisation (AEA), 1999.

[2] SVRC Services, "International Standards Survey," CA38809-101, May, 1999.

[3] J. Bowen and V. Stavridou, "Safety-critical systems, formal methods and standards," *Software Engineering Journal*, vol. 4, pp. 189-209, 1993.

[4] K. A. Eastaughffe, A. Cant, and M. A. Ozols, "A Critique of Standards for Safety Critical Computer-Based Systems," Proceedings of the Fourth International Software Standards Symposium (ISESS' 99), Curitiba, Brazil, 1999.

[5] US Department of Defense, *MIL-STD-882C: Standard Practice for System Safety Program Requirements*, 1996.

[6] North Atlantic Treaty Organisation, *NATO STANAG 4404: Safety Design Requirements and Guidelines for Munition Related Safety critical Computing Systems*, 1996.

[7] North Atlantic Treaty Organisation, *NATO STANAG 4452: Safety Assessment of Munition-Related Computing Systems*, 1996.

[8] UK Ministry of Defence, *Interim Def Stan 00-54: Requirements for Safety Related Electronic Hardware in Defence Equipment*, 1999.

[9] UK Ministry of Defence, *Def Stan 00-55: Requirements for Safety Related Software in Defence Equipment*, 1997.

[10] UK Ministry of Defence, *Def Stan 00-56: Safety Management Requirements for Defence Systems*, 1996.

[11] Society of Automotive Engineers, *ARP4754: Certification considerations for highly integrated or complex aircraft systems*, 1996.

[12] Society of Automotive Engineers, *ARP47561 Guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment*, 1996.

[13] RTCA Inc., *Software considerations in airborne systems and equipment certification, RTCA/DO178B*, 1992.

[14] International Electrical Commission, *Functional safety: safety-critical systems. International Standard IEC 61508*, 1999.

[15] S. Wilson, T. P. Kelly, and J. A. McDermid, "Safety Case Development: Current Practice, Future Prospects," presented at Safety and Reliability of Software Based Systems - Twelfth Annual CSR Workshop, Bruges, Belgium, 1997.

[16] P. Bishop, R. Bloomfield, L. Emmet, C. Jones, and P. Froome, *Adelard Safety Case Development Manual*, 1998.

[17] S. Gardiner(ed.), "Testing and the Safety Case," in *Testing Safety-critical Software*: Springer, 1999, Chapter 2.

[18] N. Talbert, "Interview with John McDermid: The Cost of COTS," *IEEE Computer*, vol. 31, pp. 46-52, 1998.

[19] P. Lindsay and J. McDermid, "A systematic approach to software safety integrity levels," in Proceedings of SAFECOMP' 97, York, 1997.