THE WHO, WHERE, HOW, WHY AND WHEN OF MODULAR AND INCREMENTAL CERTIFICATION

J L Fenn*, R D Hawkins†, P J Williams‡, T P Kelly, M G Banner††, Y Oakshott‡‡**

Representing the Industrial Avionics Working Group

*BAE SYSTEMS, UK, jane.fenn@baesystems.com
† BAE SYSTEMS, UK, richard.hawkins@baesystems.com
‡General Dynamics (United Kingdom) Limited, UK, phil.williams@generaldynamics.uk.com
**University of York, UK, tim.kelly@cs.york.ac.uk
††BAE SYSTEMS, UK, michael.banner@baesystems.com
‡‡AgustaWestland, UK, OAKSHOTY@AgustaWestland.com

Keywords: Modular, Incremental, Certification, IMA, GSN

Abstract

The Defence Technology Strategy identifies modular and incremental certification as a key enabler to 'Through-Life Capability Management' as a means of reducing the impact and hence cost of re-certification of changes to systems. The Ministry of Defence has funded the Industrial Avionics Working Group, an industrial research consortium, to undertake a 'hot research' project investigating the production of a modular safety case (SC) for an aerospace software system currently under development. This paper provides feedback and lessons learned from this project.

1 Introduction

The Defence Technology Strategy (DTS) highlights rapid capability upgrade as a cornerstone for the UK in gaining military advantage and identifies affordable assurance of software, in particular, as a technology priority. The increasing life expectancy of major platforms drives a need to consider these issues in the context of Through-Life Capability Management (TLCM). Modular and incremental certification are key enabling technologies as they provide a method for considering the impact of change on the certification of a system as it is upgraded throughout its life.

The Industrial Avionics Working Group (IAWG) has been developing an approach to modular and incremental certification, including the trial deployment on an aircraft programme. This paper highlights the outcomes and lessons learned from this programme.

2 IAWG 'Hot' Research Task

IAWG is an industrial consortium made up of BAE Systems, AgustaWestland, General Dynamics (United Kingdom) Limited, GE Aviation and Selex S&AS.

The UK Ministry of Defence (MoD) funded a research task through IAWG to mature and develop modular and incremental certification techniques in a 'hot' environment – a task run in parallel with a real project. A modular software SC was developed for the mission computer of an aircraft that is currently being procured by MoD from BAE Systems, where the computer utilises an ASAAC-compatible Integrated Modular System (IMS) [8]. The modular software SC was developed in parallel with the conventional monolithic SC. This resulted in minimum risk to the project whilst providing a full-scale alternative SC that can be swapped in once the modular and incremental techniques are sufficiently mature.

The research task was limited to considering software certification only, as this was the scope of the SC which was being 'replaced'. Work is ongoing to consider the impact of extending the approach to the system, platform and enterprise level. The research task was also limited to considering modular certification, this being the initial step towards incremental certification in the IAWG method. Work is also ongoing to develop and mature incremental certification.

3 Why Modular and Incremental Certification?

Analysis within the IAWG partner companies has highlighted that, for all but the most trivial of changes, the current cost of re-certification of change is related to the size and complexity of the system being changed. For many changes, re-certification costs approach or exceed the initial certification costs. The ambition is that cost of re-certification of change is related to the size and complexity of the *change* itself, rather than that of the system. These concepts are shown in Figure 1.

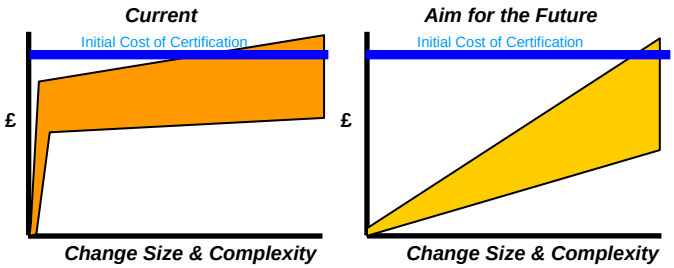

Figure 1: Certification Cost Relationships

The first step to incremental certification, and achieving these benefits, is the generation of a modular safety argument. As well as facilitating incremental certification, modular certification also brings its own benefits e.g. ease of construction (work sharing) and managing scale. The IAWG process for modular certification utilises certain boundaries within the design as the basis around which safety arguments are formed. The boundary selection is optimised by considering predicted changes to the system, and the required assurance such that these boundaries might best contain the impact of these changes upon the safety argument and evidence. Dependency Guarantee Relationships (DGRs) record the design boundary conditions of interest to the safety argument, as shown in Figure 2, where a dependency on one element in Design Module A is satisfied by the guarantee of an element in Design Module B. The SC domain is represented using the Goal Structuring Notation (GSN) as defined in [7].

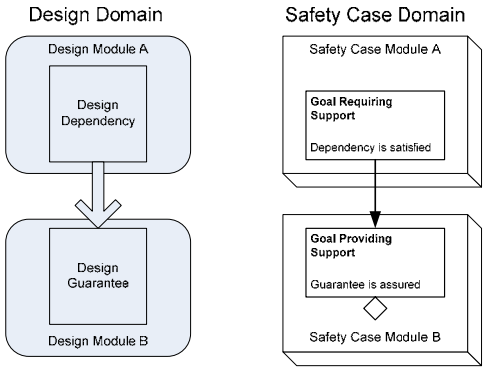

Figure 2: Dependencies and Guarantees

The IAWG incremental certification process, which is still under development, provides a systematic means of assessing the impact of change by identifying which Guarantees or Dependencies are affected by the change and reducing the re-certification effort to only those affected. Our assertion is that focussing on only affected areas will reduce the overall cost of certification of change and result in the change cost becoming more closely aligned with the change size and complexity. This should achieve a reduction in lifecycle costs, reduced time to market and the need to batch system changes.

4 How is Modular and Incremental Certification Achieved?

There are a number of stages to the modular and incremental certification process. Given that the process for incremental certification is still immature, this section focuses on the modular certification process. The production of a modular safety argument for the system under consideration is a precondition for achieving incremental certification under the IAWG method. The main steps in the IAWG modular certification method are described in the following sub-sections.

4.1 Identifying Change Scenarios

There are several reasons why it is important to analyse a system for expected change scenarios over its projected lifetime. Firstly, it will help assess the potential benefits that may be achieved through incremental certification. If as a result of the analysis there are no changes expected, then the full benefits of modular certification may not be realised, and it may therefore be decided not to adopt a modular approach. (However as discussed in section 3 there are other reasons why a modular approach may still be adopted). Where such changes are identified, then the analysis will be key context for the optimisation of system design and SC architecture when creating a Modular SC for the system.

The trial deployment identified a number of different types of change scenario. These included new and changed functional requirements originating from the customers (MoD); users (RAF air or ground crew); and regulatory bodies. Other change scenarios arose from changes in operational usage, the need to fix outstanding problems, and provision to fix those problems that are as yet unknown. Another change scenario was the management of hardware obsolescence (both in ‘new build’ and maintenance). Finally there was recognition that there are secondary effects from other changes e.g. the obsolescence of tools required in producing or maintaining the system.

The effect on the system due to the changes may be functional (relate to system behaviour) or operational (usage related). The consequences of these changes with respect to the safety of the system may be summarised as:

- New safety requirement,
- Changed safety requirement,
- No change to associated safety requirements,
- No obvious safety relationship,
- Change to assurance levels relating to the change.

Given the above information, it is apparent that the number of possible changes to a system, over its life, is huge, with complex combinations. In order to make the change scenarios manageable, a subset of the total number of changes needs to be considered. This subset is derived by categorising and filtering on:

- Likelihood of change,
- Size of change,
- Frequency,
- Complexity,
- Relationship to safety,
- Any required grouping of changes.

The aim in making this subset of changes should be to reduce to a distilled set of change scenarios for a particular system.

4.2 Defining and Optimising the Safety Case Architecture

Kelly, in [6], described how many of the following concepts from system architecture and object-object oriented design can be applied to the partitioning of a safety case into an architecture of well-defined safety case modules:

- **High cohesion** – where the responsibilities of the SC module are well-focussed to assuring, for example, the argument relating to the subject design module
- **Low coupling** – where the reliance of the SC module upon other SC modules is low
- **Well-defined interfaces** – where any collaborations between SC modules only occur via well-defined module interfaces
- **Information Hiding** – to ensure the impact of change can be determined, only the minimum necessary information should be 'exposed' at the public interface of the SC module and all information not used at the interface should be kept private to the SC module

The SC architecture for a system provides a high level view of the interconnections between the SC modules. An example SC architecture is shown in Figure 3.

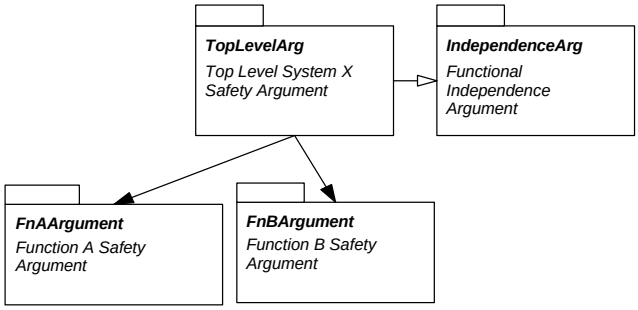

Figure 3: Safety Case Architecture

Dowding [1] considered a selection process for change scenarios as a means of optimising the modularisation within the SC. A ranking system is proposed, based on importance and likelihood, to prioritise which sub-set of identified change scenarios should be utilised for this optimisation. IAWG found that many change scenarios result in trade-offs, either between the recommended SC architecture optimisation for each scenario, or between design architecture and SC architecture recommendations. To address this issue an assessment framework has been developed [4]. This framework recommends that, for changes during either the design lifecycle or for projected in-service changes, design approaches and architectures are assessed alongside the impact assessment on the SC to ensure mutual optimisation and to guide design selection. The IAWG Modular Software SC process definition document [3] recommends an iterative approach to finalising the SC architecture and provides an example SC architecture for an ASAAC-compatible IMS.

4.3 Identifying DGRs

Dependency-Guarantee Relationships (DGRs) are created for the software design elements relating to SC modules in order to support the safety argument. DGRs are used to capture the important guaranteed properties of a software component (the *Guarantees*), and define the properties on which that component is dependent in order to uphold its guarantee (the *Dependencies*). The IAWG has developed a process for generating DGRs. This process was applied to the software as part of the case study. DGRs are currently generated using software design information, though alternative methods are being considered.

Dependencies from one software element may be satisfied by the Guarantees provided by other elements. This relationship may be captured in a *Dependency-Guarantee Contract (DGC)* between elements. Creating DGCs leads to the creation of a 'daisy chain' as the Dependency in one element is supported by the Guarantee in another element, whose associated Dependencies are supported by further Guarantees, and so on. This process is illustrated in Figure 4. Element A has a DGR defined which states that Guarantee G1 is provided if Dependency D1 is met. Element B has a DGR which states that G2 is provided if D2 and D3 are met. G1 from element A will meet the dependency D2 of element B. This relationship can be captured as a DGC. Note that all the corresponding dependencies must be satisfied before a Guarantee can be assured.

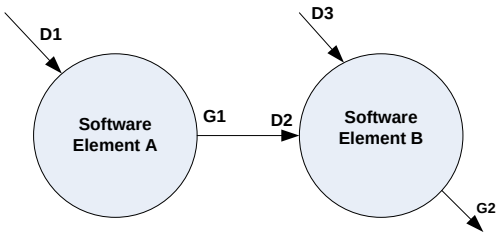

Figure 4: Linking elements using DGRs

The DGRs and DGCs (where defined) provide context to the safety argument, as discussed in the next section. For the safety argument to be valid, it is important to have confidence that all of the dependencies for each module have been correctly identified. There are a number of techniques which can be used in order to identify dependencies. The IAWG trial deployment used a manual analysis technique, which was considered to provide adequate confidence for the level of assurance that was required from the argument. The level of confidence required in the completeness of the set of dependencies will vary with the assurance requirements placed on the system. For systems with which there is a high risk associated, higher assurance is required in the identification of dependencies. For such systems, a more formal approach to dependency identification would be required. The IAWG has undertaken some initial research into formally validating DGRs and developing a rigorous notation for expressing DGRs.

4.4 Generate Safety Argument

Once the SC architecture has been defined it is possible to begin to generate the safety argument for each module. The argument presented for a module will often make use of the DGRs defined for that module to make claims about assuring the guarantees of that module. The claims relating to the guarantees of one module can be used to support the dependencies of another module. In this way, the individual safety argument modules can be composed together in order to form a coherent integrated argument for the entire system.

It should be noted that the modules of argument relating to physical entities in the system will often not make claims relating to safety, instead a number of claims relating to a set of defined guarantees will be provided. It is only once integration of the argument modules has occurred that the safety argument for the system, as a whole, is formed. It is for this reason that linking together the modules is so important for modular certification.

The modular GSN notation (defined originally by Kelly in [6]) has been used by IAWG to represent the modular arguments. One extension to GSN defined to support modular SCs is the *Away Goal.* An away goal references a goal (claim) defined within another module.

Away goals effectively represent 'hard-wired' links to other modules. This means that, for example, Module A must specify up-front the goal from Module B which is required to support it. The drawback of this approach is that if a change is made to the argument in Module B, Module A will correspondingly have to change, such that it now links to the updated argument in Module B. The effect of this is that it is necessary to change Module A, as a result of any change made to Module B. This does not support one of the stated aims of the modular and incremental certification process that is to limit the impact of changes to the system.

An alternative approach for linking argument modules together is to use *safety case contracts*, a concept first introduced by Kelly in [6]. Kelly proposes that where a successful match can be made between two or more modules, a contract should be recorded of the agreed relationship between the modules. The advantage of using SC contracts to link argument modules, rather than away goals is that the modules no longer link *directly* to the goals providing support in other modules. Instead a module references the SC contract, it is then the SC contract which identifies the appropriate goal in another module to provide support. In this case, if a change is made to the argument in Module B, Module A no longer needs to be changed to link to the updated Module B. Module A continues to link to the SC contract and instead it is the SC contract which is updated to link to the updated version of Module B. Using this *indirection* the argument modules can be isolated more effectively from changes in other modules.

In Kelly's original work on SC contracts, the contracts themselves are captured in tabular form. The IAWG approach instead proposes the use of GSN to capture the SC contracts that exist between modules. There are a number of advantages to using GSN for this purpose. Firstly, the contract becomes part of the GSN argument structure itself. It is then possible to see the complete argument represented in GSN, it is not necessary to consult tables which exist as separate entities from the rest of the argument to obtain an overall view. Secondly, the full expressiveness of GSN notation can be used to reason about the relationship between the goals. It was found in practice to be difficult to capture all of the rationale for why the contract between the modules was valid using the tabular form. GSN allows the use of strategies and justifications where necessary to make the rationale explicit. Finally, the SC contract is captured as its own GSN argument module. This makes it possible to provide an explicit link in the argument to the contract module itself. More detail on the IAWG approach to using contracts to compose modular SCs can be found in [2].

The IAWG trial deployment found that as the number of modules in the SC architecture increases, it is easy for the structure of the argument to become too complicated distracting the reader from the fundamental structure of overall argument. The IAWG showed that it is often unnecessary for all modules in the architecture to be 'visible' to all others. It can aid clarity of the argument to limit the visibility of some of the modules. The concept of *module containment* was proposed by IAWG to address these issues.

The basic concepts of module containment are that every module created must have one, and only one, containing module declared for it. The containing module defines the *scope* of the module (only modules declared to have the same containing module share the same scope). A module cannot be referenced from outside the containing module (i.e. it is only available to modules of the same scope). This means that, for example, an away goal reference cannot be made to a goal provided by a module with a different scope.

The use of module containment was found to be an effective way of managing complexity within large-scale modular safety arguments. Further details on modular containment can be found in [3].

5 Where Should Modular and Incremental Certification be Utilised?

The IAWG research addresses the assessment of a product to determine receptiveness to the approach. This research addressed the following question: 'What key criteria make the application of modular and incremental certification beneficial?' The five key criteria identified are outlined in section 5.1. A process has been developed to assess receptiveness of a system to modular certification based on these criteria [5], which has been trialled on a rotary wing case study. Having established that a programme is suitable , a number of external factors that may impact application are identified in section 5.2. Finally section 5.3 addresses the adoption argument for a demonstrably receptive programme.

5.1 The 5 Key Criteria

5.1.1 Criteria 1 - Distilled Set of Change Scenarios

The potential impact of the distilled set of change scenarios (see section 4.1) on the modular safety argument can be assessed. This can be compared with the potential impact of those change scenarios on a traditional monolithic safety case. Where the impact of the changes is reduced for a modular safety argument, this return on investment can be off-set against the overheads (e.g. additional up-front detailed analysis (see section 4.3)) of modularising the safety argument. Provided provision for the anticipated change is

built into the optimised design and SC architectures the full benefits of modular certification should be realised over time.

5.1.2 Criteria 2 – Re-use

As software and systems engineering processes mature, greater emphasis is being placed on the benefits that can be gained from reuse. There may be an existing requirement for reuse or a modular certification approach may actually facilitate reuse.

Benefits may be gained in terms of both cost and schedule in re-using already proven entities, provided they are used in a compatible context. The issue of context compatibility is often complex and additional costs associated with creating and maintaining SC modules / interfaces that are sufficiently generic to support reuse should be considered. Trading off the benefits against these additional costs can establish whether reuse of components within the safety case would be viable.

5.1.3 Criteria 3 - Modularity

The research looked at the impact of modularity in the design solution, i.e. the architecture and applications, on receptiveness. An investigation into modularising SC evidence obtained from the design domain is ongoing.

The following questions must be considered: Is it feasible to construct stand-alone arguments about the modular elements?; Can the interactions between these elements be isolated and argued about?; Can an argument about non-interference between elements be constructed?; and Can all this be achieved to the required level of assurance? The suitability of any software architecture is ultimately dependent on having confidence that attributes of the system actually exist to address the primary considerations.

The trial deployment established that a system based on an ASAAC-compatible IMS is receptive. It is anticipated that other system architectures that support modularisation, are likely to be similarly amenable.

The level of modularity within the applications also impacts receptiveness. The greatest potential payback is to be gained for a system that has a high degree of freedom in placing the SC boundaries. This allows increased granularity in the SC modules to be focussed to where it is most required. A receptive modular design is one that adheres to the principles of low coupling/high cohesion and well-defined interfaces, so allowing an optimal solution to be reached.

The trade-off in exploiting the modularity in the design is ultimately between the complexity (and cost) associated with many SC modules versus the payback from containment of areas of high assurance and future change into confined areas.

5.1.4 Criteria 4 – Use of COTS and Vendor Co-operation

Multi-vendor involvement brings with it the benefits of domain expertise, but also the added complications of managing contractual boundaries and the limited availability of suitable supporting evidence. Modular and incremental certification is still subject to these considerations but also increases the need for a well-defined set of boundaries and contracts, in the technical and commercial domains, early in any programme.

Section 7 discusses the role of component suppliers. Some degree of vendor co-operation is required, whether the COTS vendor supplies a SC module for the guaranteed properties of their component, as advocated, or just provides supporting evidence for the integrator's safety arguments.

When assessing receptiveness in respect of COTS the following questions should be considered, whilst recognising that some of the issues raised will need to be addressed regardless of the certification approach employed.

- What is the benefit of using the COTS component?
- What is being provided to support the SC?
- What are the interfacing issues?
- Are all or a sub-set of the features of interest?
- What needs to be incorporated into the overall SC?

Sufficient visibility is required to ascertain that the COTS system is suitable to meet the integrators requirements in terms of assurance of the required safety features, within a compatible context. If this position is defensible, then the product is receptive, as it should be possible for the vendor and/or integrator to construct a compatible SC.

5.1.5 Criteria 5 – System Size and Complexity

The potential benefits to be gained from modular and incremental certification are likely to be much greater for large complex system. If a change to the design can be isolated by modular boundaries, the total cost of a change could even be prohibitive without the option of incremental certification. Conversely for a small system, the total costs of a change (and so maximum payback) may be insufficient to warrant considering a modular certification approach.

5.2 External Factors

Having made the decision that a programme is suitable, it is necessary to consider the impact of the following external factors: supportiveness of customer(s), adequacy of tool support, and the availability of trained practitioners.

5.3 Adoption Argument

A product is deemed to be receptive to the application of modular certification technologies if the benefits that may be reaped (see section 3) can be shown to outweigh the technical and commercial risks. To this end the argument for adoption should focus on quantifiable benefits (technical and commercial), the identification of risks for the adoption, the risk mitigation strategy, and establishing that the residual risks are demonstrably tolerable.

6 When Should Modular and Incremental Certification be Utilised?

The trial deployment undertaken by IAWG derived DGRs from pre-existing design information, hence this activity was retrospective. The IAWG asserts that greater benefits would

be gained by defining a SC architecture and identifying DGRs early in the design lifecycle. This would maximise opportunities to mutually optimise the design approach and corresponding SC. An ongoing research task will provide additional opportunities to assess the potential for influencing the early life-cycle phases.

Whilst it is believed that maximum benefit can be achieved from applying the process early, the practical reality is that there are already many systems in service that have a significant service life remaining. The IAWG advocate assessing the value of applying modular certification techniques during the lifetime of legacy products. Where there is a likelihood of further changes to the product, and there is some basis for modularity in the product (either existing, or the opportunity to introduce), the technique should be considered.

7 Who is Affected by Modular and Incremental Certification?

The current monolithic approach to safety arguments requires the Prime Contractor to take an overarching view, deriving and flowing down requirements to suppliers, and seeking evidence to fill the gaps in the argument structure. The modular approach supports the delegation of argument structure to the designers of the components of the system, leaving the prime to focus on the overall structure of the argument, and on the integration of the argument modules (drawing a strong parallel to their role on the technical aspects of the product).

The supplier of a component into the product is no longer required to artificially produce a 'safety case' at some distance from the hazards created by the integrated product. Rather they are able to create a SC module for the guaranteed properties of their component that can be relied upon. This case can clearly communicate the boundary to which the argument can be taken and identify the 'dependencies' that must be addressed by a 3rd party, without the supplier ever having to know who that is.

As an argument module is produced, and dependencies arising from requirements, design and implementation decisions are elicited, there may need to be a commercial interchange with the Prime. This is nothing new, however the structured approach of the modular argument provides a framework for improved transparency of what's behind these 'changes' and hence provides an opportunity for a more open commercial method of working.

The modular approach can extend to COTS suppliers who are able to make claims about their product, and substantiate them with arguments, evidence and defined dependencies. Alternatively the impact of COTS components can be 'contained' within a wrapper argument provided by the supplier of a bespoke element that incorporates COTS elements.

The above clearly illustrates that the supply chain involved in designing a product are affected by a change to modular certification. Also affected are the other stakeholders in a traditional certification. The customer will also need to be prepared to provide support e.g. active involvement in identifying change scenarios. The regulator/customer or their agent (e.g. Independent Safety Assessor) will need to review and accept arguments presented in a different way. It is likely they will be able to see the elements of the argument being integrated in a more phased manner, allowing better distribution of review workload and a better traceability to the low level design aspects.

8 Conclusions

In order to achieve incremental certification and to realise the associated benefits, the first step is the successful development of a modular SC. This paper has discussed an approach to developing successful modular SCs based on experiences and lessons learnt from a trial deployment undertaken by the IAWG. The IAWG trial deployment has demonstrated the feasibility of adopting a modular approach for the certification of an ASAAC-compatible IMS. Ongoing research by IAWG is developing the process further and investigating the feasibility of incremental certification. The work reported here has been undertaken by the Industrial Avionics Working Group and funded by the UK Ministry of Defence.

References

[1] M. Dowding. "Maintenance of the Certification Basis for a Distributed Control System," *MSc Thesis, Department of Computer Science, University of York*, (2002).

[2] J. L. Fenn, R.D. Hawkins, T. P. Kelly, P. Williams. "Safety Case Composition Using Contracts – Refinements Based on Feedback from an Industrial Case Study", *Proceedings of 15th Safety Critical Systems Symposium (SSS '07),*edited by Felix Redmill and Tom Anderson, Springer (2007).

[3] J. L. Fenn. "Industrial Avionics Working Group Modular Software Safety Case Process", *IAWG-AJT-301,* (2007).

[4] J. L. Fenn. "Parallel Certification Study Design Architecture and Safety Case Architecture Trade-Off and Mutual Optimisation", *IAWG-AJT-703*, (2007).

[5] J. L. Fenn. "'Parallel Certification Study - Receptive Product'", *IAWG-AJT-702,* (2007).

[6] T. P. Kelly. "Concepts and Principles of Compositional Safety Cases", *(COMSA/2001/1/1) - Research Report commissioned by QinetiQ*, (2001).

[7] T. P. Kelly. "Arguing Safety – A Systematic Approach to Managing Safety Cases", *PhD Thesis, Department of Computer Science, University of York,* (1998).

[8] UK Ministry of Defence. "Interim Defence Standard 00-74 ASAAC Standards", *Defence Standard,* (2005)