CONTRACTING FOR ASSURANCE OF MILITARY AVIATION SOFTWARE SYSTEMS

Squadron Leader D.W. Reinhardt
Royal Australian Air Force
Research Student
University of York
derek.reinhardt@defence.gov.au

Professor J.A. McDermid OBE FREng
Head of Department of Computer Science
University of York
United Kingdom
john.mcdermid@cs.york.ac.uk

Abstract

Contracts are instruments which provide a legally binding agreement for the purchase/exchange of goods or services. While both civilian and military aviation software systems are acquired by contract, in the military circumstance the contract has an additional regulatory and safety assurance role.

Military contracts typically achieve the regulatory and safety assurance outcome by ensuring that relevant contract clauses reference applicable regulations and safety standards. However, industrial practice suggests several key factors that influence the effectiveness of the contracting approach to achieving safety. For military aviation software systems, these factors seem to be particularly prevalent.

The paradigm of the standard (i.e. goal-based, prescriptive or combinations thereof) is a factor as it influences the perspectives and behaviours of suppliers and acquirers with respect to evidence provision to the regulatory authority. Another prevalent factor is the extent to which the standard guides the effective establishment and execution of a contract through providing certainty in both product and evidence delivery. Standards may also have a substantial impact on achieved product safety.

This paper examines these factors and aims to assess their effect on military aviation software system contracts. The paper sets out a framework for relating evidence to safety objectives. The framework also provides an approach for identifying, analysing and evaluating the tolerability of limitations (e.g. incompleteness) in evidence for assuring safety. A fictional example is presented to demonstrate application of the framework to the contracting process. Observations on evaluation of the framework are presented to provide support to their validity in industrial practice.

Keywords: Architecture, Assurance, Aviation Systems, Contracts, Fault Tolerance, Safety, Software Assurance, Software Safety, Tender.

1 Introduction

Contracts are instruments which provide a legally binding agreement for the purchase/exchange of goods or services. A contract normally consists of terms and conditions, and is supported by technical annexes to define the requirements for goods/services and scope of work. For aviation systems, contracts are used for the acquisition and/or modification of these systems between the developer/manufacturer (i.e. supplier) and the owner or operator (i.e. acquirer). While both civilian and military aviation systems are acquired by contract, there are key differences in the role of contracts between the military circumstance and the civilian circumstance. Specifically in the military circumstance, the achievement of regulatory and safety assurance functions has to be enabled through the contract. This is because the regulations and safety requirements established by military regulators are not legally enforceable onto a supplier unless the contract enables this. This is very different to the civil case (e.g. the Federal Aviation Administration (FAA) or Civil Aviation Safety Authority (CASA)) where the responsibility to promulgate and enforce regulations on suppliers is enshrined in law.

Military contracts typically achieve the desired regulatory and safety assurance outcome by ensuring that relevant contract clauses reference the applicable regulations and safety standards. However, on its own, this may be insufficient. The authors' practical experience suggests several key factors that influence the effectiveness of the contracting approach to achieving safety regulation. For example, the clarity within the nominated standard of the requirements for evidence provision from supplier to regulator seems to be a major factor. For military aviation software systems, these factors seem to be particularly prevalent. This paper is focussed on military software systems, however due to the unavoidable coupling between software and its system, where relevant this paper may take the perspective of the system, software or software system. For ease of discussion, we assume that certification is based on the delivery of (safety) arguments and supporting evidence to the acquirer.

1.1 Standards Paradigm: Goals-based or Prescriptive.

The paradigm of the standard (i.e. goal-based, prescriptive or combinations thereof) is a crucial factor for achieving regulation through contracts as it influences

the perspectives and behaviours of suppliers and acquirers regarding the provision of evidence to the regulatory authority. For example, a goal based standard might set high level safety objectives and permit substantial flexibility for designs, which gives benefit in defining effective products. However it may have limitations with respect to establishing contractually enforceable benchmarks for evidence provision; and this will impact suitability and sufficiency of both evidence and argument. Similarly, resolution within the contract, of evidence and argument shortfalls might be equally limited, depending on the supplier's attitude and perspective.

On the other hand, a prescriptive standard may set clear benchmarks for evidence and activity completion that are straightforward to enforce through contractual mechanisms, but have limitations in relevance to achievement of product safety objectives. This means that, depending on the supplier and acquirer's bias in worldviews (see [McR12]), the paradigm choice will affect behaviours, and these behaviours will ultimately affect the level of safety (not just evidence provision) achieved through the contract.

The question of paradigm is further complicated for complex aviation systems involving technologies (e.g. software) where failures are (predominantly) the consequences of systematic faults. This is because, across academia and industry, there is still limited consensus (refer to [JTM07], [McD07], [McK06], [NTS06], and [Wea03]) as to how to provide assurance that these faults do not lead to unacceptable aircraft failure conditions. All that can be concluded from this lack of consensus is that current approaches to providing safety assurance of software in military aviation systems have limitations. Thus, as neither paradigm is without its limitations in this context, it is likely that the more effective approach may be a compromise between both paradigms. This raises the question, what combination of goal-based and prescriptive standards elements is necessary to minimise these limitations and enable effective safety regulation via contracts?

1.2 Integrating the Standard's Lifecycle with the Tender/Contract Lifecycle

Another important factor is the way the standard integrates with the contractual lifecycle. Ideally the standard should assist in reducing uncertainty about the delivered product, argument and evidence prior to the establishment of a contract. This is important because both acquirer and supplier will be seeking confidence that the contract will be successful prior to entering into the contract. Similarly, the standard should assist during contract execution. Should safety issues emerge during the contract, then timely and cost effective resolution will be a goal for both supplier and acquirer. The contract and standard should support the resolution of safety issues, and not hinder it by contributing to dispute.

An inspection of contemporary safety standards reveals that integration between the standard lifecycle and contract lifecycle varies significantly between standards. For example ARP4754 and RTCA/DO-178B make no mention of integration with contracts as the means of evidence provision. However, they effectively achieve some potential contract integration through certification authority liaison and artefact requirements within these standards. UK Defence Standard 00-56 Issue 4 makes numerous mentions of contracts and requirements on contractors, but doesn't provide requirements for contracts relating to the provision of arguments or evidence across the contracting process. Whereas, MIL-STD-882C and D deals explicitly with contract integration, include specific references to contract clauses, tender processes and data requirements.

It is evident that the requirements of the standards have a substantial effect for the integration of the standard across the tender/contract lifecycle. This raises the question, what elements of standards, and their implementation in contracts provides appropriate certainty (regarding product and assurance evidence) for acquirers and suppliers? Is it possible to define requirements for safety and assurance standards to achieve effective contract process integration?

1.3 What Does This Mean for Standards and Contracts?

Ultimately, it is vital that the regulatory and safety assurance paradigm used be compatible with the contracts used for military acquisitions, without impairing or detracting from the achievement of system safety. Success is dependent on perspective and worldview. Contracts which provide cost and schedule certainty are preferred by both suppliers and acquirers. Suppliers will also have a vested interest in profitability and acquirers in value for money. Suppliers will generally strive to achieve safety, and the acquirer's regulatory authorities will strive for achievement of an acceptable level of safety (or risk) without significant out of scope rework to treat risks, or without the retention of intolerable risks. How to do this for the assurance of military aviation software systems is still very much a challenge.

This paper further examines the different standards and contracting paradigms, and aims to assess their effect on military aviation software system acquisition. This paper articulates more generic principles learned as a result of defining a framework [ReM11] by which to contract for architectural assurance [ReM10], and to provide claims and evidence assurance [RMc10] for aviation systems.

2 Why Military System Acquisition Contracts are Different

In civil aviation the regulator responsible for airworthiness is a government agency (e.g. the FAA). The regulator is a legally recognisable independent entity from the supplier and acquirer of aircraft and aviation systems. Regulations established by the regulator are indoctrinated in law and are legally enforceable. However, in the military aviation domain, the regulator is typically part of the same high level organisation as the acquirer. For example in the Australian Defence Force, both the Directorate General Technical Airworthiness (regulator) and the Defence Materiel Organisation (acquirer) are part of the Commonwealth of Australia – a single legal entity in the eyes of the law. In the United Kingdom, the Military Airworthiness Authority (regulator) and the UK Ministry of Defence (acquirer) report to the Secretary of

State for Defence and are part of the Crown – again a single legal entity in the eyes of the law. The same can also be said for the relationship between military regulators and acquirers in the United States of America. This relationship between the acquirer and regulator roles has several implications for the way airworthiness is regulated; of which one significant factor highlighted in Section 1 is the impact on contracts between suppliers and acquirers. Regulatory enforcement is enabled by the contract rather than via laws for the military circumstance. The following subsections elaborate several impacts for contracts.

2.1 Enforcement of Design Requirements

In civil aviation, the supplier is required to supply aircraft and aviation systems that meet the applicable airworthiness design requirements promulgated by the regulations. For example, the civil airworthiness regulations (e.g. [14CFR25]), and their supporting guidance in the form of advisory circulars, orders and notices, define a substantial set of design requirements for their applicable aircraft category. These are usually supplemented by additional design requirements agreed between the supplier and regulator throughout the certification process. Design requirements are typically in the form of product requirements and assurance requirements (which includes evidence, verification, etc.). However, in military aviation, the airworthiness design requirements (or requirement to establish and agree them) must be included in the contract if they are to apply to the development. This means that the contract Statement of Requirement (SOR) should include or reference applicable airworthiness design requirements, including safety assurance requirements, and that the Statement of Work (SOW) must include activities to ensure elicitation and agreement of any additional airworthiness or design requirements relevant to the design. This is no simple task, as the set of potentially applicable airworthiness design requirements may be large and complex. In the context of military aviation software systems, the subset of applicable design requirements includes assurance requirements, in addition to a range of 'product' design requirements, depending on the system application; these assurance requirements are the main focus of this paper.

2.2 Obtaining Assurance Evidence

In civil aviation, the regulator obtains evidence required for certification from the supplier as required by the regulations. The regulations will require the supplier to provide the regulator with plans, artefacts (inspection, analysis and test documentation), access for the purposes of audit, access for the purposes of witnessing / participation / conduct of tests, etc. However, in military aviation, the regulator (as part of the acquirer) obtains these types of evidence required for certification from the supplier via the contract. This means that the contract SOW must include applicable activities for the generation of relevant certification evidence, including assurance evidence. Delivery versus access to evidence is usually dictated by intellectual property considerations, and will be evident from the artefacts listed in the Contract Data Requirements List (CDRL), and supporting Data Item Descriptions (DIDs).

2.3 Resolving Shortfalls in Assurance Evidence

In civil aviation, if there are shortfalls in the supplier provision of evidence to the regulator for certification, then the onus is on the supplier to resolve the shortfalls. If the supplier doesn't resolve the issue then they don't achieve certification, and they can't sell their product. However, in military aviation, resolving the shortfall in evidence will very much depend on whether it is in or out of scope of the contract. In many respects the acquirer can be considered to have already purchased the product once the contract is signed. If the issue is within scope, then the onus is on the supplier, but if there is any ambiguity regarding scope of the contract pertaining to the issue, then the onus for resolution is shared by the acquirer. If the supplier and acquirer can't agree that it is wholly within the scope of the contract, then the issue may be the subject of contractual dispute. Ramifications of a contractual dispute can include cost and schedule implications, a requirement to elevate beyond project staff, a requirement to negotiate over contractual interpretation and compliance, etc. These issues potentially have the impact of degrading the effectiveness of safety regulation achieved through the contract, particularly where projects must seek additional funding from Government (an onerous process) to resolve the safety shortfalls via contract change proposals.

3 Impact of Uncertainty at Contract Signature

Section 2 has identified several responsibilities of contracts if safety regulation is going to be effective via the contract. Uncertainty in any of these may increase the risk of the contract being unsuccessful. Signing a contract, in some respects, involves a gamble. It is a wager for both supplier and acquirer that the supplier can provide a system that the meets the acquirer's requirements within the cost and schedule dictated by the contract. The odds (for or against) depend on the uncertainty in factors important to either supplier or acquirer. Therefore, any sensible gambler (and one that abides by causality) will acknowledge that the contract success risk is a function of the uncertainty at contract signature. Lots of uncertainty, and the odds could be dramatically against success; lesser uncertainty, and the odds might favour success. Fortunately the normal processes for getting to contract signature such as project definition and tender phases provide the contract authority with a means of seeking important information prior to contract signature. This information, if sought and used effectively, can reduce uncertainty, and thus reduce potential contract risks.

How to seek the right information and effectively evaluate it with respect to safety for military aviation software systems is still very much a challenge. Furthermore the existing standards and contracting approaches offer limited guidance on how this might be achieved effectively. Industrial experience involving project overruns and cancellations due to safety assurance concerns suggests that the current approaches are also insufficient, although mostly the evidence is anecdotal.

To further understand the implications of uncertainty at contract signature for safety it is necessary to establish where this uncertainty might exist. To elicit this, consider

the factors outlined in Sections 2.1 through 2.3 with respect to a military aviation software system and safety. In this context, uncertainty might exist with respect to the following:

- Will the design requirements proposed by the acquirer be adequate to achieve the safety objectives? Specifically, from a safety assurance perspective, will:
  - the software and system architecture, including the use of redundancy, diversity, and fault avoidance/tolerance likely permit achievement of the safety objectives?
  - the architecture provide adequate protection against systematic faults and failures?
- Will compliance with the design requirements and safety objectives be compelling based on the evidence provided? Specifically, will:
  - the behaviours of the system and its software be sufficiently understood and valid under both normal and failure circumstances?
  - these behaviours be appropriate with respect to safety?
  - the evidence support the safety assurance claims made by the supplier about these behaviours?
  - any limitations in evidence be tolerable?
- Will limitations in evidence be resolvable within the scope of the contract? Specifically, what is:
  - within scope?
  - out of scope, requiring a contract change?

Whenever there is uncertainty with respect to these questions throughout the contract lifecycle, then the contract risks relate to the following issues. The first is that the uncertainty might undermine the acquirer's aspiration to establish if the software system will likely be acceptably safe (if this supplier were to be chosen to contract with). Thus the supplier might be eliminated during the tender evaluation based on perceived uncertainty in suitability. The second, and ultimately more serious, issue is that if this design solution is contracted for, and it turns out the design has unsuitable behaviours; in this case there is risk that the acquirer may not be able to complete safety certification within the scope of the contract. Worse still, it may require the acquirer to retain risks, due to uncertainty, and these risks prove to be intolerable in practice.

If we extrapolate these factors alone, then the result is easy: have the supplier provide full disclosure to the acquirer during the tender process. However, the realities of the commercial business environment quickly show the impracticality of this aspiration. In domains where developmental and novel systems are more common-place, it is uneconomical to require suppliers to complete their development lifecycle to the point that answers to the above questions become entirely certain during the tender process. As only a small percentage of tender responses are actually successful, and tenderers already invest substantial resources in preparing them, the acquirer must be cognisant of the need to avoid deterring potentially suitable tenderers due to the level of effort required to tender. Therefore, in establishing the level of detail required in the tender response the solution must provide for sufficient disclosure and understanding, but while ensuring the minimum imposition on tenderers. This is a difficult balance.

Acquirers and suppliers enter into the tender and contracting activities with a set of motivations, aspirations and perspectives which are a unique dynamic contrast between goals for specific project success, mixed with broader commercial goals and commercial restrictions. Each of these will vary between every acquirer, supplier and circumstance. The most obvious motivations for the acquirer and supplier with respect to safety are that the solution will achieve the safety objectives, and that the evidence will show this. But it is the additional motivations that vary the perspective on achievement of this between supplier and acquirers. Acquirer motivators include:

- credibility of supplier cost and schedule forecasting,
- satisfying capability requirements,
- avoiding contract changes,
- costs of solutions falling within notional budgets, and
- delivery within capability scheduling requirements.

Supplier motivators include:

- providing a competitive tender cost/schedule,
- preservation of profit margins within the contract price,
- avoidance of contract penalties,
- ensuring that out of scope work requires a contract change (to protect the profit margin with the contract), and
- delivery of a broadly satisfactory product with minimal application of resources.

These motivators are intrinsically linked because cost and schedule are required to produce evidence, and evidence is required to show the provided solution meets safety objectives (and capability requirements). Because of this dependency, some of these motivators will work against each other, and this will cause divergence in supplier and acquirer motivations, and thus behaviours. Emergent (commercial) behaviours when issues arise that expose the polarisation between these motivators very much depends on the relationship between supplier and acquirer, the seriousness of the safety concerns or cost impacts, and the supplier's and acquirer's worldviews regarding assurance.

Given these contracting motivators, and assuming that any serious incompatibility between them for a given contract will result in limitations in successful outcomes for the contract: how might a framework be established to ensure that uncertainty at the time of contract signature can be bounded? I.e. what is the compromise between these motivators that enables the appropriate design solution to be identified during tender processes, and this solution to be achieved during contract execution?

The remainder of this paper examines how an approach might be established. Illustration of the benefits of the approach will be via an artificial but realistic example.

Consider an upgrade of an analogue flight control system to a digital flight control system for a military helicopter. The flight control system provides automatic flight functions and stability augmentation, and is mixed to the existing mechanical control system between pilot controls and control actuators. The objective of the acquirer is to achieve this upgrade, including the safety regulatory functions on behalf of the acquirer's regulatory authority, through a contract. The following sections examine how this can be effectively achieved.

4 Bounding Uncertainty Prior to Contract Signature – Successfully Using the Tender Process

It has already been mentioned that the tender phase provides a means for the acquirer to seek important information prior to contract signature. This information, if sought and used effectively, can reduce uncertainty, and thus reduce potential contract risks. How much the uncertainty has to be reduced is an important question, and this introduces the concept of bounding uncertainty.

Firstly, it is important to elaborate what is meant by bounded uncertainty, in this context. Put in engineering terms, it is establishing limits (upper bounds) on the cost of producing a safe product and an acceptable safety case. Bounds can be narrowed by the provision of information to the acquirer from the supplier during pre-contract phases (e.g. tender phase) balancing the motivators identified in the previous section. The limiting factor on information provision will be the affordability, for a tenderer, of conceptual and preliminary phases of requirements and design lifecycle phases within the resources that are commercially viable given the gamble of winning the tender.

In Section 3 a set of questions were introduced based on the three identified roles for contracts with respect to safety regulation: enforcement of design requirements, obtaining assurance evidence, and resolving shortfalls in assurance evidence. These questions were further refined into the context of military aviation software systems to seek information the regulator would require to be informed about safety assurance. These questions were holistically centred on three main topics: architecture, behavioural arguments and evidence provision/suitability.

Therefore, an approach to breaking this problem down further would be to examine how to bound uncertainty across each of these three topics. I.e. to effectively determine how much the regulator should know about each of these topics during the tender phase to be satisfied of a likely positive outcome, should the project go to contract.

Returning to the artificial flight control system example, let's assume that the contract authority for this project has determined that an open tender is the most suitable form of acquisition strategy for this project. The aircraft original equipment manufacturer has no off-the-shelf solution available, and various contractors have expressed interest in developing a solution.

The remaining sections of this paper will now describe how this tender may be prepared and evaluated, the most suitable option identified, and a contract established and executed for this option. Section 5 of this paper will consider the architectural topic, what information is required to inform acquirers about architectural suitability and how this information can be elicited in the pre-contract signature phases. Section 6 of this paper will consider the behavioural arguments and evidence topics, what information is required to inform about sufficiency, and how this information can be elicited by the pre-contract phases. Section 7 will then examine how issues arising as a result of the remaining uncertainty are identified and resolved post contract signature.

The example being used within this paper assumes a single phase tender process. However, this process may not always be the most suitable. Where the acquisition or modification is of substantial complexity, then the single phase tendering process may not incentivise suppliers to invest a level of effort to develop their solution to a level that permits effective evaluation. This may particularly be the case for an entire aircraft development. In these cases a two-phase tender may be more suitable. The first phase would identify holistic solutions that accord with the safety objectives of the program and use a normal tender construct. The second would be a partially funded tender phase, where funding is provided to a restricted set of tenderers to further develop the tender artefacts supporting evaluation against the framework. The second phase would be more synonymous with a Restricted Tender, but include provision for funding so that tenderers can invest a level of effort which they are compensated for. Such options are available where the acquirer is not satisfied that the tenderer is incentivised to offer competitive solutions, or to resolve the uncertainty to a level consistent with the constraints on acquirer funding. These multi-phase tender processes won't be directly addressed in the example used in this paper, but the concepts illustrated herein can be applied to those circumstances also.

5 Obtaining Solution Architectural Certainty

Obtaining architectural certainty from the tender phases and prior to entering into a contract is important as it enables early insight into potential architectural shortfalls. It also forces supplier consideration of architectural suitability including fault avoidance and fault tolerance; this is important as there is evidence in industrial practice that this is sometimes overlooked. A four step process is proposed for obtaining solution architectural certainty, as follows:

1. Set measurable benchmarks for architectural suitability
2. Inform architectural suitability using the tender process
3. Evaluate architectural suitability during the tender evaluation, and
4. Provide architectural assurance during contract execution.

The following sub-sections elaborate the four step approach to achieving this for the flight control system example and outline some of the benefits.

5.1 Setting Benchmarks for Architectural Suitability

The first step to obtaining architectural certainty is to set some benchmarks for solution architectural suitability. The benchmarks should not be specifying solutions so they do not stifle novelty or limit flexibility; they should instead set measurable criteria against which different solutions can be evaluated. In this way, benchmarks allow the acquirer a way of measuring solutions against each other from a safety perspective. Benchmarks also provide a way of specifying to a supplier what attributes their software system design should have.

A review of the literature reveals that there is very little published guidance on explicit benchmarks for architectural suitability, particularly with regards to systematic faults and failures. Some standards permit assurance levels on specific system components to be reduced based on architecture, but this is not a measure of the overall architectural adequacy. Therefore, new approaches are required to achieve this if architectures are to be effectively evaluated during tender evaluations. One possible approach has been developed by the authors that introduce the concept of an Architectural Safety Assurance Level and Layered Fault Tolerance Requirements [ReM10]. The core idea with the Architectural Safety Assurance Level is that it provides a measure of how many layers of defence an architecture provides against systematic faults. The layering defences against faults concept is synonymous with the 'defence in depth' principle often referred to in security manuals. It also derives from the 'Fail Safe Design Criteria' from [AC25.1309]. The 'layers of defence' concept is a useful measure because it is independent of specific solutions, emphasises architectural handling of faults between architectural components, and provides a notional level of confidence based on the number of layers of defence against each fault type.

To set the benchmark for the supplier, clauses could be developed for both the tender and contract SOR to communicate these benchmarks. The clauses should communicate the solution properties regarding the requisite number of layers of fault tolerance and avoidance/detection and handling requirements. The following is an example of a generic SOR clause to achieve this:

The [System Name] architecture and mechanisms for achieving fault avoidance and fault tolerance, against each type of credible systematic fault, shall meet the requirements for layers of fault avoidance and fault tolerance, where the number of layers is commensurate with the worst credible failure condition, as specified at {reference a Table in the SOR detailing the benchmark numbers of layers for each failure condition severity}

A specific instantiation of this clause for the Architectural Safety Assurance Level approach is described at [ReM11].

5.2 Informing Architectural Suitability

To reduce architectural uncertainty at the time of contract signature, the tender phase requires a mechanism to be informed of the architecture. This implies that a tender deliverable needs to include information about the suitability of the proposed architecture. Since the information will be used by the acquirer to evaluate the suitability of the architecture against the benchmarks, it is useful to ensure the information directly addresses the benchmarks set in Section 5.1.

One possible approach would be to require the tenderer, through the tender SOW, to provide a *Conceptual System and Software Architecture Suitability Documen*t. The document would describe how the system's architecture and mechanisms for achieving fault avoidance and fault tolerance against systematic faults would meet the benchmarks established above. The intent is to provide a description of the architecture at a level of fidelity that the acquirer can evaluate against the benchmark, without forcing the supplier to completely design and implement the system before contract signature. For a largely mature design, the document can focus on what already exists, and whether or not it requires supplementation; for a developmental design it provides a framework for the supplier to cost the architectural elements of their system with improved accuracy. The following is an example of the generic Tender SOW clauses to achieve this:

Total Layers of Defence. The [Tenderer] shall prepare a [Conceptual System and Software Architecture Suitability Document] per TDRL XX to describe how the [System Name] architecture and mechanisms for achieving fault avoidance and fault tolerance, against each type of credible systematic fault, is proposed to meet the {reference to SOR's requirements for number of layers of fault avoidance and fault tolerance to systematic faults}.

Adequate Constraints. The [Tenderer] shall prepare a [Conceptual System and Software Architecture Suitability Document] per TDRL XX to describe how each proposed constraint (i.e. absence/detection and handling mechanism) is proposed to achieve the architecturally layered fault tolerance requirements as defined by the SOR {reference the SOR requirement}.

A specific instantiation of these clauses for the Architectural Safety Assurance Level approach is described at [ReM11].

For the flight control system example, let's assume that each of the proposed options provides a *Conceptual System and Software Architecture Suitability Document*, for which the proposed architecture is briefly summarised as follows:

- Option A
  - Quad redundant digital flight control system incorporating two flight control computers with two independent channels per computer.
  - Dual sensors including air data systems, attitude/heading reference systems and triplex actuators and actuator sensors.
  - Incorporation of software fault tolerance within each computer.
- Option B
  - Quadruplex digital flight control computers incorporating a single channel per computer.
  - Incorporation of software fault tolerance within each computer.

- Option C
  - Quad redundant digital flight control system incorporating two flight control computers with two independent channels per computer.
  - Sensors include a single air data system, dual attitude/heading reference systems and dual actuators and actuator sensors.
  - Design is based upon a flight control system from a fixed wing military aircraft, and adapted for this application.
- Option D
  - Simplex digital control system, single control panel, and single sensors including air data system, attitude and heading references, and actuator position sensors.

Note that these architectural descriptions are deliberately brief. They are intended to be illustrative for the purposes of making a point about how contracting processes can be used to inform their suitability. A more detailed example, which includes a more thorough architectural analysis, is to be documented within the first author's PhD thesis.

5.3 Evaluating Architectural Suitability

The purpose of the tender requesting this information is to permit evaluation of the extent to which the holistic safety and software architecture requirements are costed into the tender response. The retrospective incorporation of constraints to treat systematic failure modes is rarely straightforward, particularly when architectural change is required. Therefore, it is in the acquirer's interests to establish the extent to which the contractor has determined an architecture based on the types of constraints required to meet safety objectives. While it is recognised that many sub-system architectures may not be well defined for large system acquisitions, the absence of this information in a tenderer's response will permit the acquirer to adjust the contractor's proposed costing by a risk figure based on the amount of uncertainty (or extent of suitability) in the tenderer's proposed architecture to provide a normalised evaluation of tenderer's responses that do include the relevant information.

As can be seen from the differing architectures proposed by Options A through D, the complexity of each solution differs notably. Using the benchmarks set for the architecture, each option is evaluated. The evaluation results are summarised as follows:

- Options A and B – Treatments to all classes of systematic fault use layers of fault avoidance and fault tolerance mechanisms. Architecture is suitable.
- Option C – Treatments relating to omission and value failures of the air data system sensor rely on fault avoidance via absence arguments only. There is limited software fault tolerance proposed for these failures. Therefore the architecture is deemed to contain weaknesses against these systematic faults and thus would require changes to adequately treat. Architecture is potentially unsuitable, and is flagged for further consideration once evidence provision is evaluated.
- Option D – Treatments relating to omission and value failures of sensors and flight control computers rely on fault avoidance from absence arguments only. This is assessed to provide grossly inadequate defences against these classes of systematic failures. Architecture is deemed unsuitable, and option is eliminated from the tender.

5.4 Providing Architectural Assurance

Once the preferred tenderer has been identified, and any uncertainties regarding the architectural assurances are tolerable (assuming in this case that it will end up being either Option A or B because of their superior architectural suitability), then it is possible to develop a contract between the supplier and acquirer.

Under the contract, the acquirer will need to achieve two things. The first is that they will need to maintain the benchmarks for product suitability by inclusion of SOR clauses similar to those defined in Section 5.1, but for the contract. Further the acquirer will require means to establish if the final 'as-delivered' architecture meets the prescribed benchmarks. This can be achieved by requiring the contractor to deliver (via appropriate SOW contract clause) a *System and Software Architectural Assurance Document*. The document would describe how the system's architecture and mechanisms for achieving fault tolerance against systematic faults actually achieves the benchmarks established above. The following is an example of the generic Contract SOW clauses to achieve this:

Total Layers of Defence. The [Contractor] shall prepare a [System and Software Architectural Assurance Document] per CDRL XX to describe how the [System Name] architecture and mechanisms for achieving fault avoidance and fault tolerance, against each type of credible systematic fault, meets the {reference to SOR's requirements for the number of layers of fault avoidance and fault tolerance to systematic faults}.

Adequate Constraints. The [Contractor] shall prepare a [System and Software Architectural Assurance Document] per CDRL XX to describe how each proposed constraint (i.e. absence/detection and handling mechanism) achieves the architecturally layered fault tolerance requirements as defined by the SOR {reference the SOR requirement}.

A specific instantiation of these clauses for the Architectural Safety Assurance Level approach is described at [ReM11].

The Contract Data Requirements List (CDRL) should require that various iterations of the document be delivered at relevant system engineering milestones to permit the acquirer to monitor the evolution of the architecture under the contract. This monitoring is important because it allows the acquirer to measure the progression of the architecture throughout the contract lifecycle, and to respond early if there are divergences to acquirer understanding and assumptions from the tender evaluation.

Obviously Data Item Descriptions (DIDs) will be required for all the deliverables listed in the CDRL (or TDRL mentioned in the previous section). DIDs are generally structural, and could be developed to provide a specific heading framework to support provision of the relevant information. However the SOR clauses setting benchmarks for the product, and the SOW clauses requiring provision of the information are the means by which the adequacy of the architecture is enforced. DID

compliance is only a means of ensuring potentially relevant information has been provided in a structure that is understood by the acquirer.

6 Obtaining Argument and Evidence Certainty

Obtaining argument and evidence certainty from the tender phases and prior to entering into a contract is important because it enables early insight into potential argument and evidence shortfalls. It also forces explicit context specific agreement between acquirer and supplier on the measures of argument and evidence sufficiency for which there is no agreed universal approach. A four step process is proposed for obtaining argument and evidence certainty, as follows:

1. Set benchmarks for argument and evidence suitability
2. Proposal of argument and evidence using the tender process
3. Evaluate argument and evidence suitability during the tender evaluation, and
4. Provide argument and evidence assurance during contract execution.

The following sub-sections elaborate the four step approach to achieving this for the flight control system example and outline some of the benefits.

6.1 Setting Benchmarks for Argument and Evidence

The first step to obtaining argument and evidence certainty is to establish how to set benchmarks for argument and evidence sufficiency. In keeping with the notion of a compromise between goal-based and prescriptive standards, the benchmarks should not specify specific techniques or methods for evidence generation, but instead provide a coherent framework for how evidence will be related to safety properties, and provide a set of criteria for establishing when evidence generation is completed. In this way, benchmarks allow the acquirer a way of measuring evidence sufficiency from a safety perspective.

A review of the literature reveals that there is very little literature in the public domain that sets explicit benchmarks for measuring argument and evidence sufficiency. The generalised goal-based approaches provide flexible argument structures [Kel98], and the development of patterns and anti-patterns has provided some reusable argument structures that might provide the basis for argument agreement [KeM01]. Argument assurance [Wea03] and assurance deficit approaches [SSEI09] provide an approach, but they lack detail on evidence sufficiency benchmarks sufficient to reach a consensus before contract signature. Less generalised goal-based (or objective-based) approaches such as RTCA/DO-178B provide a detailed framework of sub-objectives that would form part of an argument structure, but unfortunately stray into prescription in some limited areas [Rei08]. In contrast, prescriptive standards provide very clear measures of evidence completion, but are lacking in justification for evidence sufficiency in a given context. Therefore, new approaches are required to achieve this if arguments and evidence are to be effectively evaluated during tender evaluations.

6.1.1 Benchmarks for Argument

First, we address the question of argument. Having an entirely flexible argument is useful in that it does not constrain design solutions, the claims that can be made about them, and does not limit novel approaches to arguing safety. Further, this approach means that the argument has the flexibility to present evidence that is important to the argument, rather than producing evidence because the standard requires it (as with the prescriptive standards). But the drawback is that it is very difficult to communicate acquirer expectations to the supplier if the overall approach doesn't provide a way for the supplier to measure the suitability of their design solution and argument. It should also be apparent when inappropriate design solutions are proposed and inappropriate claims used to defend them.

To bound the uncertainty such that the acquirer can be confident in the supplier's intended argument approach, a means is required to convey the attributes of acceptable arguments to the supplier through the tender and contract documents. The purist goal-based approach doesn't achieve this. On the other hand an entirely prescriptive standard provides a set of evidence that the supplier should produce, but the argument relating the evidence to the behaviours of the product and the safety claims may be either implicit, missing in part or missing entirely. Thus a move to activity and technique prescription doesn't address the need of contracts either. So how can these approaches be combined without undermining their advantages, while ensuring their usefulness as a contracting benchmark?

Let's consider any argument as consisting of some holistic claim about a property of a product with respect to safety, and a strategy for showing the credibility of this claim. This emphasises two key points: the claim and the strategy. This could be considered analogous to the relationship between the Goal and its Strategy in Goal Structuring Notation (GSN) as described by [Kel98].

Consider the claim first. At the architectural level, architectural assurance is based upon the presence of layers of fault avoidance or fault tolerance, such as detection/handling mechanisms. Let's call the requirements that define the specific fault avoidance or fault tolerance behaviour at the relevant layer a 'constraint', as a generalised term. Therefore it follows that an argument is required for the suitability of each 'constraint' and that each 'constraint' needs to be assured commensurate with its impact on safety. The architectural suitability elements of Section 5 provide a means for establishing the collective suitability of 'constraints', and how their behaviours combine to provide the requisite architectural defences against systematic faults. Thus we are left with providing evidence that each individual 'constraint' is assured, and we need to turn our attention to the strategy to achieve this. Note that the 'avoidance' constraint amounts to correctness, e.g. of control algorithms).

Consider this; what if the general evidence types used to support claims about the 'constraint' were categorised in with respect to software lifecycle products for which there

is consensus. For example, current standards almost universally agree that there should always be:

- requirements at the system level,
- one or more design decompositions and refinements of these requirements (e.g. high level software requirements, abstract software requirements, low level software requirements),
- source code, and
- executable object code.

These are real software lifecycle products, and they exist as some form of physical document or electronically for virtually all developments. When they are lacking, it is not because they are inappropriate, it is because there is a gap in evidence. Further, since they appear in all of the existing software assurance standards, we can utilise the consensus this provides. There is some dispute that contemporary methods such as model-based software engineering undermine these general categories. However, consider this perspective. Model-based software engineering simply changes the sources of evidence for these products from human centric processes to tools. The evidence still exists; it just takes a different form depending on the construct of the tool. Such product benchmarks also provide a rationale for the types of evidence model-based software engineering tools should produce as their output, and this may help with establishing criteria for the qualification of such tools. Hence this paper argues that the categories of life-cycle products should still exist; it's just the source of evidence that changes (i.e. human to tool).

Examining the strategy in more detail, why not structure a set of generic sub-claims around 'attributes' of the aforementioned software lifecycle products (high level requirements, low level requirements, source code, executable object code, etc.). For example an attribute of a low level requirement might be its 'traceability' to higher more abstract level requirements. Numerous attributes (e.g. accuracy, consistency, traceability, compliance, verification coverage, etc.) can be defined which represent the extent of properties appropriate to the software lifecycle product.

Each 'attribute' would describe a distinct property of the evidence, such that collectively the properties would provide measurable knowledge in the claims made from the software lifecycle product. Further, instead of making the starting point of requirements entirely general (as is done in most software assurance and safety standards), ensure that they are examined with respect to real product behaviours that affect safety - in this case each specific 'constraint'. Effectively, we are explicitly annotating the 'attributes' of each software lifecycle product, with respect to the claims about the specific 'constraint'. This provides a generic universal approach to linking software lifecycle products (i.e. the real world evidence) with the properties of the software we are trying to make safety claims about.

One possible approach has been developed by the authors' (see [RMc10]) that introduces the concept of a Claims Safety Assurance Level (CSAL), and a set of generic arguments centred around the 'attributes' of lifecycle products of specified 'constraint' level requirements and applicable abstract level requirements, low level requirements, source code and executable object code. Since not all 'constraints' provide an equal contribution to the architectural level defences, and thus not all 'constraint' arguments are equal, a framework is also included that assists in determining the importance of satisfying each particular argument.

6.1.2 Benchmarks for Evidence Sufficiency

Turning our attention now to addressing the question of benchmarks for evidence. It has already been described that the goal-based approach allows flexibility in evidence, and that this is desirable. However the drawback is that a means of measuring and justifying the sufficiency of evidence has to be incorporated into each and every argument. This may be repetitive, and detract from the focus on the product aspects of the argument. On the other hand, the prescriptive approach lacks flexibility in evidence, and it does not help to group evidence in ways such that the 'so what?' can be answered from this evidence. However, the strength of the prescriptive approach is that it is very clear to suppliers trying to determine activity costs for inclusion in the tender response. So how can these approaches be combined without undermining their advantages, while ensuring usefulness as a contracting benchmark?

Consider this; what if the following assumptions are made:

- The set of evidence supplied is never infinite (because we don't have infinite time or money), thus the assurance it provides is never absolute; so there will always be limitations in the totality of evidence.
- The evidence produced from each method or technique will always have some limitation with it, and complementary evidence from one or more methods or techniques will usually be required to resolve the limitation.
- As there will always be limitations in the evidence; why not change the focus to determining if the limitations are tolerable in the specific context?

Further, a generic framework could be provided for determining the tolerability of the limitation in evidence for each argument that is going to be made. Since evidence is best presented at the sub-claim level, this is the best place to immediately assess the impact of tolerability of limitations of evidence. Once assessed with respect to the specific 'constraint' the (in)tolerability can then be evaluated in the context of the impact on architectural assurance, and thus provide meaningful insight into product safety risks.

The framework could take into account the generic properties of evidence (refer [Wea03]) including:

- *Relevance* of the evidence (as produced by method or technique X) to the sub-claim (e.g. compliance of the source code with the applicable low level requirements for constraint Y),
- *Trustworthiness* of the evidence based on who and how it was produced, and
- *Results* of the evidence, including where the results provide counter evidence.

This is advantageous because the supplier can be required to identify the limitations with each type of evidence proposed with respect to these properties of evidence. The supplier can also be required to identify how they will resolve any limitations through provision of additional evidence. The approach is generic because it reflects generic properties and limitations of evidence. The techniques and methods used to the produce the evidence are entirely within the supplier's control. The better the techniques and methods they propose, the fewer the limitations they will have to address; but this is a choice for the supplier. Further, the concept provides a means for the supplier to think critically about what techniques and methods they are proposing and provides a means for measuring the adequacy of each technique and method. Finally, when they've worked out their techniques and methods, they can cost these into their proposal, and thus the supplier can be confident in their proposal costing for the provision of evidence.

One possible approach that uses these principles has been developed by the authors. It introduces the concept of a Evidence Safety Assurance Level (ESAL) and 'Tolerability of Limitations' [RMc10]. The remaining sub-sections discuss how these principles can be incorporated into tenders and contracts to bound uncertainty.

6.2 Proposal of Argument and Evidence

To reduce uncertainty about the intended safety argument at the time of contract signature, the tender phase requires a mechanism to be informed of the argument. This implies that it is useful to know which generic claims are going to be applied to each architectural 'constraint'.

One possible approach would be to require the tenderer, through the tender SOW, to provide a *Software Assurance Plan* to describe which set of claims are going to be demonstrated for each 'constraint'. To ensure consistency in tenderer responses it is advantageous to align where possible the claims to the generic software lifecycle products and the generic attributes of each. The following is an example of a generic Tender SOW to achieve this:

The [Tenderer] shall prepare a [Software Assurance Plan] per TDRL XX to propose the attributes that will be assured, for each software lifecycle product, for each constraint described in the [Conceptual System and Software Architecture Suitability Document].

A specific instantiation of these clauses for the Claims Safety Assurance Level approach is described at [RMc10]. [RMc10] provides a systematically established set of attributes for each lifecycle, that provides confidence in its completeness of attributes for generic software behavioural claims.

To reduce uncertainty about the intended limitations in evidence for each of the aforementioned attributes at the time of contract signature, the tender phase also requires a mechanism to provide information on the likely scope of the body of evidence and its potential limitations.

One possible approach would be to require the tenderer, through the tender SOW, to provide two things:

- a *Software Development Plan* to describe which methods and techniques are going to be applied across the development, and
- a *Software Assurance Plan* to describe how any limitations in the evidence produced from the methods and techniques described in the software development plan are tolerable with respect to relevance, trustworthiness and results.

Software Development Plans are already routinely in use within projects; and this should be no surprise to any reader. However the key contribution this paper is proposing is a sister document (the Software Assurance Plan) that presents the analysis and justification for the adequacy of the Software Development Plan, with respect to the tolerability of limitations in evidence concept. By requiring each tenderer to explicitly justify the adequacy of their software development, then suppliers are provided a consistent set of expectations for costing their software development programs. This is important when it comes to establishing which of two or more software developments programs is most adequate with respect to evidence provision.

For the purposes of clarity the Software Assurance Plan is quite different from more conventional deliverables such as Software Verification Plans. A Software Verification Plan will usually provide the description of activities used to demonstrate requirements satisfaction. The Software Assurance Plan presents the analysis and justification for the adequacy of the Software Development Plan, by describing the claims and justifying the evidence proposed for each type of 'constraint'. Conventional plans such as verification plans, test plans, etc. are still envisaged being companion documents to the Software Assurance Plan.

The following is an example of a generic Tender SOW clause to achieve production of the Software Development Plan and Software Assurance Plan:

Software Development Plan. *The [Tenderer] shall prepare a [Software Development Plan] per TDRL XX to describe the methods and techniques proposed to be used throughout the software development lifecycle, including description of techniques or methods used prior to this development but for which evidence is relevant.*

Software Assurance Plan. *The [Tenderer] shall prepare a [Software Assurance Plan] per TDRL XX to describe how the evidence produced from the application of the [Tenderer] proposed methods and techniques is proposed to assure tolerability of limitations in evidence with respect to relevance, trustworthiness and results, for each attribute of each software lifecycle product, for each constraint described in the [Conceptual System and Software Architecture Suitability Document].*

A specific instantiation of these clauses for the Evidence Safety Assurance Level and Claims Safety Assurance Level approach is described at [RMc10].

For the flight control system example, let's assume that each of the proposed options provides a *Software Development Plan* and *Software Assurance Plan*, for which are briefly summarised as follows. Note that for the purposes of brevity within this paper this in only an illustrative summary without the corresponding

justification. It doesn't represent the full content of these plans.

- Option A
  - ARP4754 system safety program with software assurance to RTCA/DO-178B Level A.
- Option B
  - DefStan 00-56 Iss 4 system safety program with software assurance to DefStan 00-55 Iss 2 SIL4, including the application of formal methods.
- Option C
  - MIL-STD-882D system safety program, with new software developed to RTCA/DO-178B Level A, and reused software developed to MIL-STD-498.
- Option D
  - MIL-STD-882D safety program, with software developed to MIL-STD-498.

6.3 Evaluation of Argument and Evidence

The purpose of the tender requesting this information is to permit evaluation of the extent to which the holistic evidence requirements are costed into the tender response and to establish if they meet the acquirer's expectations. The retrospective supplementation of evidence is rarely straightforward, particularly when it results in a change to one or more of the lifecycle products such as requirements, design or code. Therefore, it is in the acquirer's interests to establish the extent to which the contractor has proposed a sufficient set of evidence. While it is recognised that the evidence would not yet exist at the time of tender, clear insight into:

- the techniques and methods proposed,
- what evidence will be produced?,
- how this evidence will combine?, and
- what limitations in the evidence might be intolerable?;

will permit the acquirer to adjust the contractors proposed costing by a risk figure based on the amount of uncertainty (or extent of suitability) in the tenderers proposed evidence set. This would provide a normalised evaluation of tenderers responses compensating for tenders that do include the relevant information.

Considering the examples proposed in the previous section, it is evident that the evidence set proposed by Options A through D varies substantially for each proposal. Using the benchmarks set for the argument and evidence, each option is evaluated. Assume, for the sake of illustration, that the evaluation results are summarised as follows:

- Option A – There appears a limitation with the extensiveness of normal and robustness verification proposed against low level requirements relating to time-dependent properties, including synchronisation, of the flight control laws in relation to fault tolerance to jitter (early and late) related effects on sensor inputs. Tenderer is requested to clarify their proposal.
- Option B – There appears a limitation of the extensiveness of the application of analytic and empirical verification of behaviours relating to fault tolerance to value failures of air data system and attitude/heading reference system sensors. This is due to fault tolerance mechanisms being incorporated into device drivers which can only be verified in the Systems Integration Laboratory but for which there is no means with the current toolset to inject these fault conditions for the purposes of verification. This limitation is flagged for clarification with the tenderer.
- Option C – Limitations in evidence for reused software are substantial with respect to low level requirements, low level requirements verification, and coverage of implementation from requirements based verification. These limitations are assessed to be intolerable.
- Option D – Already eliminated based on architectural evaluation.

Options A and B require further clarification with the Tenderers, and this will be sought. Option C is eliminated from the tender evaluation due to intolerable evidence limitations, and Option D was already eliminated based on architectural shortfalls. Clarification with Options A and B reveals the following additional information for the evaluation:

- Option B – the limitation remains as the tenderer claims that low level verification undertaken prior to integration verification will provide sufficient evidence in this regard. Therefore verification of these requirements on the target computer with credible fault conditions is via inference only. These limitations are assessed to be intolerable. Option B is eliminated from consideration.
- Option A – the extensiveness of normal and robustness verification has been adequately clarified and is acceptable.

Therefore, Option A is selected as the winning Tenderer, and negotiations are commenced to progress to contract signature.

Note that in reality there are many other selection criteria for a product, and so it is common for capability, force integration, and political factors amongst others to affect selection. Hence, these other factors may sometimes require compromise on the ideal safety solution. However, this does not invalidate the process proposed in this paper. Instead, the process in this paper enables the acquirer to be informed about the safety assurance aspects such that it is possible to make informed trade-offs between safety assurance and other selection criteria. For example, it may be possible to choose Option B, make decisions regarding risk treatment or retention, because other benefits out-weigh the impact of its limitations.

6.4 Providing Argument and Evidence Assurance

Once the preferred tenderer has been identified (in this case Option A); and any uncertainties regarding the claims and evidence assurances are tolerable, then it is possible to develop a contract between the supplier and acquirer.

Under the contract, the acquirer will require a means to establish if the final 'as-delivered' claims and evidence meets the prescribed benchmarks. This can be achieved by requiring the contractor to deliver (via appropriate

SOW contract clause) a *Software Assurance Summary Document*. The document would describe how the assurance of the 'attributes' of software lifecycle products actually achieves the benchmarks established during tender processes. The following is an example of the generic Contract SOW clauses to achieve this:

Achievement of Claims and Attributes of Software Lifecycle Products

The [Contractor] shall prepare a [Software Assurance Summary] per CDRL XX to describe the attributes that have been assured, for each software lifecycle product, for each constraint described in the [System and Software Architecture Document].

Assessing the Evidence

The [Contractor] shall prepare a [Software Assurance Summary] per CDRL XX to describe how the evidence produced from the application of the [Contractor] proposed methods and techniques has assured the tolerability of limitations in evidence with respect to relevance, trustworthiness and results, for each attribute of each software lifecycle product, for each constraint described in the [System and Software Architecture Document].

A specific instantiation of these clauses for the Architectural Safety Assurance Level approach is described at [ReM11].

7 Resolving Issues after Contract Signature

Despite best intentions, whenever there is uncertainty there is potential for it to lead to an undesirable outcome as development progresses. The sections prior to this have largely been focussed on trying to bound the uncertainty in areas that really affect the case for safety. However, once a contract is commenced, if issues do arise with respect to architecture, claims or evidence, then it is useful to establish in advance the approach for resolution of these issues.

Considering the ongoing example of Option A, and let's assume that during preliminary design review several issues are identified as follows:

- Issue 1 – Proposed treatments to value failures of air data system airspeed data are identified to be inadequate under conditions of transition to the hover. A revised treatment is proposed requiring an adaptation to flight control law transition criteria to provide an improved fault tolerance against this fault.
- Issue 2 – Verification and validation of the accuracy of the software requirements relating to discrete implementation of the legacy analogue control laws is identified to contain shortfalls relating to the reuse of modelling from the previous implementation. Additional modelling of the discrete implementation is viewed as required by the acquirer.

There are two main options for providing contract scope for the work to resolve unforeseen issues that arise: either within the original contract, or through a contract change. Both are discussed in the following sub-sections.

7.1 Resolution within Contract Scope

Resolution within the contract scope is entirely dependent on the supplier openly acknowledging the requirement to resolve the issue and perhaps do extra work. However, when profit margins are at risk, and there is risk of schedules being affected, it is not uncommon for suppliers to argue work is out of scope.

Consider the two issues identified our example:

- Issue 1: This treatment is deemed in-scope of contract because it was a contractor oversight during the conceptual design proposal. Evidence is provided commensurate with previously identified attributes, lifecycle products and constraints.
- Issue 2: Acquirer and supplier enter into contractual dispute regarding the provision of additional evidence modelling the discrete implementation, because the supplier claims their limitations in the modelling are tolerable.

One way to address Issue 2 is to make absolutely explicit this requirement for limitations to be resolved to the satisfaction of the acquirer through a statement of work line item. This line item can then be costed and suppliers will be empowered to resolve such issues. An example of how this might be achieved is as follows:

Intolerable Limitations in Evidence, Claims or Architecture

Where the [Acquirer]'s certification evaluation establishes that the [Contractor] has not achieved the requirements of the {reference applicable SOR and SOW clauses relevant to architecture, argument and evidence}, or there are shortfalls in the 'Tolerability of Limitations' of evidence versus the criteria specified by this contract, then the [Contractor] shall undertake one or more of the following remediation actions to resolve the shortfalls to the satisfaction of the certification authority:

- *engineering change to architectural constraints,*
- *engineering change to implementation of architectural constraints, or*
- *additional analysis, verification and validation by further or supplementary application of methods or techniques.*

The [Contractor] shall amend all relevant deliverables per the CDRL to incorporate the engineering changes and additional evidence.

Note to Contractors

The above clause provides the means for the certification authority to address shortfalls against architecture, argument and evidence expectations. While this clause may be interpreted to result in unbounded programmatic risk for the contractor, the intent is to focus both acquirer and contractor efforts at establishing unambiguous consensus during the tender process and contract negotiations. The contractor should not sign the contract if they believe there remains substantial uncertainty regarding the provision of evidence against the framework, and instead request further clarification during contract negotiations.

The aim of this approach is to ensure that the tender phases and contract negotiation phases have systematically identified, disclosed and evaluated the intended body of evidence and that all intolerable shortfalls have been included within the contract. Thus the example clauses would only come into effect if an issue remains, and this would be less likely and less serious because the evidence planning was systematic in the first place.

The drawback to this approach is that suppliers may interpret this as a very risky statement of work line item and cost it commensurately. However there is a positive to the behaviour this generates for tender evaluation. If the acquirer evaluates the cost attribution against this line item from each tenderer, and there are notable differences in the costing, then the acquirer can use this to establish the tenderers confidence levels in their own cost estimates for achieving architectural, claims and evidence assurance. This is a very useful tool during tender evaluation, and something that is not easily gauged by other means. Even if the clause is removed during contract negotiations due to supplier concerns, its inclusion during the tender process is extremely revealing about supplier confidence in their proposals and costing.

7.2 Resolution Outside Contract Scope

Resolution of shortfalls outside the contract scope is easy from the perspective of defining the scope of work; as usually the analysis to determine that the architectural changes, design changes or evidence supplementation will be clear from the analysis done to demonstrate it is outside the original contract. If there is contingency funding to fund the contract change, then it will also be relatively straightforward for the acquirer.

However, if contingency funding is not available this is a very challenging path as it usually involves the allocation of additional funding to a project from Government. Most Government committees responsible for funding of military aviation system acquisitions are not sympathetic to issues that emerge late in the project lifecycle which were not forecast with original costing, allocated as contingencies, or articulated as program risks.

For the purposes of this paper, the approach described at Section 7.1 is preferred at least at the tender phase, so that the likelihood of additional out of scope work is well understood during the tender phase, and minimised in the contract phase.

8 Evaluation

As the concepts introduced in this paper differ substantially from existing approaches, evaluation of their effectiveness is required. However, because it is often difficult to apply novel approaches to real projects at the initial proposal of these approaches, evaluation by experiment is not straightforward. For this reason, literature [Van07] regarding the design of studies for participative research was examined to establish that preliminary evaluation of the concepts outlined in this paper was suited to survey questionnaire of suppliers, acquirers and related stakeholder agencies (e.g. regulators). A series of targeted workshops is being used to complement the survey questionnaires.

8.1 Description of Evaluation

A detailed survey questionnaire was prepared using the principles for questionnaire design from [Opp01] and [BAN86]. The questionnaire asked a mix of open and closed questions regarding the concepts and application thereof presented in this paper and the supporting literature. The questionnaire was provided to representatives of a range of supplier and acquirer agencies representing a cross section of the following:

- Military Regulatory/Certification Authorities
- Supplier Contractors
- Acquirer Agencies
  - Australian – Defence Materiel Organisation
    - Sustainment System Program Offices
    - Acquisition Projects
  - United Kingdom Ministry of Defence
  - United States Air Force
  - Defense Contract Management Agency
- Contractors to Defence (Professional Service Providers)
- Science and Technology Organisations supporting Defence Acquisition

8.2 Results of Evaluation

The evaluation is on-going; however analysis of results has been undertaken on 15 completed surveys. The surveys represent a cross-section of the above listed stakeholders from Australia, Canada, New Zealand, the United Kingdom, and the United States of America . The evaluation undertaken to date has provided the following feedback:

Framework

- Acquirers and Certification Authorities indicated that the approach may have helped to avoid several historical (and current) project issues where architectural safety shortfalls were responsible for project cancellation or significant project delays and cost increases. However, they noted that correlation in retrospect is easier than in reality.
- Some respondents were deterred by the notion of yet another assurance framework, while others noted that current approaches had limitations, and this approach seems compatible and extends some current approaches.
- Some respondents were deterred by the complexity of the inter-related assurance concepts and contracting mechanisms, although several of these indicated that the concepts were less complex than many of the systems to which they would apply. This would perhaps provide natural selection of suppliers that cope with complexity.
- Some respondents were positive about the concept of defences and ‘constraints’ although they had reservations that supporting methods as yet wouldn’t enable them to model the relationships effectively. Extension to existing methods might be required.
- Many respondents indicated that the ‘tolerability of limitations’ concept appeared useful in that it provides some inherent rules for providing and measuring supplier justifications. There was some support for developing the rules even further, and providing examples.
- The majority of respondents indicated that one or more worked examples, of both a tender costing based on the proposed tender clauses, as well as of implementing the underlying ASAL/CSAL/ESAL frameworks from [RMc10] and [ReM10], would be beneficial.

Tender Evaluation and Contract Negotiation

- Acquirers and suppliers indicated that the proposed approach does provide product and evidence focus during the tender phase that appears beneficial, although until they actually apply it, this is only speculation.
- There was positive response to knowledge of architecture during tender processes, although some suppliers were concerned about how they might progress their design processes to that point for some tenders, particularly those involving sub-vendors.

Contract Execution

- There was consensus that knowledge of architecture and knowledge of evidence at tender would reduce the difficulty of contract execution.

Risk Evaluation

- Regulators and operational representatives indicated that knowledge of product behaviours and remaining defences would help with planning operational treatments, and with developing emergency procedures.
- Regulators indicated that they were still unclear how evidence/assurance shortfalls correlated to risks, and suggested developing the framework further to address risk measurement.

Cost and Schedule

- Suppliers expressed reservations about being able to resolve issues they haven't costed within contract scope, although praised that the underlying frameworks would potentially provide improved knowledge of product and evidence requirements during tender phases and thus reduce the opportunity for issue resolution within contract.
- Some suppliers and acquirers expressed concern that this would increase the cost of tender processes, and potentially deter some tenderers.
- Some suppliers had reservations about the perceived paradigm shift, and how they would cost effectively educate their staff on how to work within such a framework.

Further distribution to an increased sample size of the aforementioned organisations is presently being undertaken. Final results will be published within the aforementioned PhD thesis, and may also be targeted for journal publication.

8.3 Analysis of Evaluation Results

Analysis of the surveys received indicates the following:

- There is correlation between the respondent comments and the motivating issues. This indicates that the motivating issues are probably valid.
- A cross section of prescriptive versus goal-based 'world views' were evident in responses to motivating issues and general principles revealing that there is diversity in 'world views', although the results don't directly suggest a resolution.
- There was not direct correlation between 'world views' and position/negative comments indicating that there are issues of 'world view', education, paradigm shaping, and framework limitations involved.
- There is correlation between respondent comments on feasibility and usefulness and the general principles on which the framework is based. This indicates that the general principles may be widely agreeable, even if their opinions on the implementation differ.
- Suppliers focussed strongly on cost and schedule implications, and competitiveness with respect to other suppliers. The level of knowledge on the topic of safety assurance varied substantially between suppliers, acquirers and regulators.
- While supplier sentiment was that regulations are already too constraining for their businesses to be innovating, there was acknowledgement of the problems with the current approaches to assurance.
- Acquirers focussed on successful tender processes leading to successful contract execution. The level of knowledge on the topic of safety assurance varied substantially between acquirers and regulators.
- Views of safety and risk varied between respondents and warrants further clarification.

It is hoped that through the on-going conduct of the evaluation, and publishing of results, that consideration be given to apply these concepts to a real world system acquisition. This would overcome the limitations of the constructed environment of a survey and workshop.

9 Conclusion

This paper has examined factors affecting the provision of safety assurance evidence for military aviation software system contracts including the impact of the standards paradigm, integration of the standard with the contract lifecycle, enforcement of design requirements, obtaining of assurance evidence and resolution of shortfalls in product and evidence.

Acquirer (and regulator) certainty in the software systems behaviours and fault tolerance, the inherent argument in the claims and framework used to relate evidence to safety objectives, and the approach used for identifying, analysing and evaluating the tolerability of limitations in evidence are identified as particularly important. The impact of uncertainty in these topics at the time of contract signature has been examined with respect to the potential for a successful contractual outcome. Approaches have been proposed for obtaining assurances and bounding uncertainty by pre-contract and throughout the contract. An example was used to illustrate the benefit in the approach.

Observations on preliminary evaluation results conducted with respect to a framework based on these certainty motivators have been presented to provide support to their validity in industrial practice. Based on these initial observations, further evaluation in industry and acquirer communities is recommended.

10 References

[14CFR25] Title 14 Aeronautical and Space, Code of Federal Regulations Chapter I Federal Aviation Administration, Department of Transportation, Subchapter C – Aircraft, Part 25 *Airworthiness Standards: Transport Category Airplanes*

[AC25.1309] Federal Aviation Administration, Advisory Circular, AC25.1309-1A System Design and Analysis, 21 Jun 1988.

[BAN86] D.R. Berdie, J.F. Anderson, M.A. Niebuhr, *Questionnaires: Design and Use*, Second Edition, The Scarecrow Press, Inc. Metuchen, N.J. USA, 1986.

[DO178B] RTCA Inc., *RTCA/DO-178B: Software Considerations in Airborne Systems and Equipment Certification*, Washington D.C.: RTCA Inc., 1992.

[JTM07] D. Jackson, M. Thomas, L Millet, Editors, *Software for Dependable Systems: Sufficient Evidence?,* Committee of Certifiably Dependable Software Systems, National Research Council, National Academy of Sciences, USA, 2007.

[Kel98] T.P. Kelly, *Arguing Safety – A Systematic Approach to Managing Safety Cases*, PhD Thesis, Department of Computer Science, University of York, 1998.

[KeM01] T.P. Kelly, J. McDermid, *Safety Case Patterns – Reusing Successful Arguments*, Rolls-Royce Systems and Software Engineering, University Technology Centre, Department of Computer Science, University of York, Heslington, York, 2001.

[McD07] J.A. McDermid, *Risk, Uncertainty, Software and Professional Ethics*, 20 August 2007.

[McK06] J. McDermid, T. Kelly, *Software in Safety Critical Systems: Achievement and Prediction*, Nuclear Future, Volume 03, No. 03, 2006.

[McR12] J. McDermid, A. Rae, *Goal-Based Safety Standards: Promises and Pitfalls*, presented at the Safety Critical Systems Symposium, Springer, Bristol, February 2012.

[NTS06] National Transportation Safety Board, *Safety Report on the Treatment of Safety-Critical Systems in Transport Airplanes*, Safety Report NTSB/SR-06/02, Washington, D.C., USA, 2006.

[Opp01] A.N. Oppenheim, *Questionnaire Design, Interviewing and Attitude Measurement*, New Edition, Continuum, London, Great Britain, 2001.

[Rei08] D.W. Reinhardt, *Considerations in the Preference for and Application of RTCA/DO-178B in the Australian Military Avionics Context*, presented at the Australian Safety Critical Systems Association Conference, Aug 2008

[ReM10] D.W. Reinhardt, J.A. McDermid, *Assuring Against Systematic Faults Using Architecture and Fault Tolerance in Aviation Systems*, presented at the Improving Systems and Software Engineering Conference (ISSEC), Aug 2010.

[ReM11] D.W. Reinhardt, J.A. McDermid, *Contracting for Architectural, Claims, and Evidence Assurance for Military Aviation Systems*, Departmental Technical Report, Department of Computer Science, University of York, Oct 2011.

[RMc10] D.W Reinhardt, J.A. McDermid, *Assurance of Claims and Evidence for Aviation Systems*, presented at the 5th IET Conference, Oct 2010.

[SSEI09] R. Hawkins, J. McDermid, Software Systems Engineering Initiative, SSEI-TR-0000041, *Software Safety Evidence Selection and Assurance*, Issue 1, University of York, October 2009.

[Van07] A.H. Van De Van, *Engaged Scholarship, A Guide for Organizational and Social Research*, Oxford University Press, Oxford, Great Britain, 2007.

[Wea03] R.A. Weaver, *The Safety of Software – Constructing and Assuring Arguments*, PhD Thesis, Department of Computer Science, University of York, 2003.