

Adapting to Changes in a Software Safety Assurance Approach



Mike Standish and Dr Mark Hadley discuss the challenges of developing non process-based assurance arguments using diverse evidence.

An aircraft (like many systems) comprises many interconnected elements, of varying complexity, which provide the fundamental functionality for the system's operation. Many of these elements will perform mission, security, or safety critical roles and the system's functionality is commonly underpinned by software. If there is a *failure* in the software then there can be a *failure* of the system to perform its function. To deploy a safety-critical system it is imperative to have *confidence* in the system's underpinning software and this is gained by performing software safety assurance. The term *confidence* is defined as providing "trust in a thing" and to "show [a level of] certainty" [1]. If there is an insufficient level of confidence in the software then there is an insufficient level of confidence in the system.

This article provides a number of thoughts on the wider technical and procedural aspects that may need to be considered if there are changes to how process-based software safety assurance is currently conducted. The thoughts relate to vendors, system integrators, equipment procurers, and regulators.

Judging Software Integrity

A traditional way to gain confidence in the software is to develop it to a pre-defined *process*. Within the civil aviation domain, a commonly applied guideline is DO-178C [2]. The guideline can also be adopted within the military aviation domain as an acceptable means of compliance to provide design assurance of airborne safety-related software (SRS).

There are also other approaches cited as being suitable to develop safety-critical software within wider domains (e.g. SafeScrum [3]). However, regulatory bodies are creating new paths to reflect revised software safety assurance stances (e.g. non-prescriptive regulatory oversight). The Federal Aviation Authority (FAA), for example, is aiming to adopt a set of three Overarching Principles (OPs) (*intent, correctness, and innocuity*).

The aim of the OPs is that they will *not* supplant the existing approaches (e.g. DO-178C) but to act as *another route* to gain certification approval [4]. The OPs provide an alternative means to justify the belief in the suitability of the software for inclusion into an aircraft. The revised, non-prescriptive, stances should allow (in theory) wider forms of evidence to be included within software assurance cases.

These would not necessarily be solely focussed on process-based developments and artefacts. However, we believe that adopting such approaches requires a shift in how evidence is currently generated, gathered, and judged as the traditional objective-by-objective activities (e.g. DO-178C) may not feature within a revised software assurance approach.

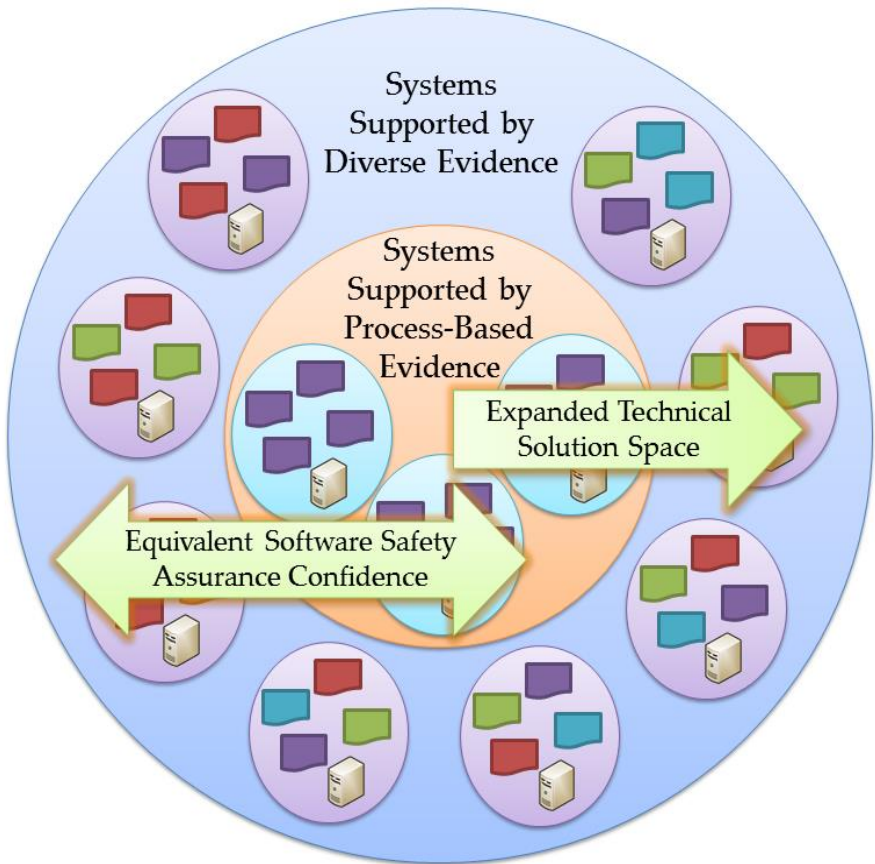
“Regulatory bodies are creating new paths to reflect revised software safety assurance stances”

Use of Diverse Evidence to Expand the “Pool of Solutions”

Having the ability to adopt varying forms of evidence can support, at least, two aims:

- it can potentially assist with fulfilling the aims of the OPs; this indicates a benefit in *implementing a safety assurance argument*; and
- it can have benefits in articulating the evidence and the supporting arguments for *systems which do not conform* to the traditional processes, for example “DO-178C compliance”.

Accepting only systems which are software process compliant over other systems (which are supported by other evidence strands such as in-service data) *reduces* the pool of systems which could provide effective capability solutions, e.g. improved Size, Weight, and Power (SWaP) characteristics. It should be noted that any system considered within a wider technical solution must still have the evidence available to demonstrate that the safety constraints can be met to achieve an equivalent level of safety. This expanded solution space is illustrated simplistically in the figure below.



Managing the Change in a Software Safety Assurance Approach

To adopt a non-process software assurance approach there may need to be a fundamental shift in *how* software safety assurance is conducted within a number of safety-critical domains. The following points detail a number of considerations which may be necessary to implement (in-full or in-part) when evidence is accumulated within a revised software safety assurance approach. The considerations (stated at a high-level in the figure below) range from stakeholder skillsets to how safety-critical software is procured.



Knowledge/Skills of Stakeholders



Through-Life Appraisal



Managing Evidence Complexity



Treatment of Evidence



Judgement of Evidence



Debate and Consensus



Procurement and Contracting

Knowledge and Skills of Stakeholders

Suitably Qualified and Experienced Personnel (SQEP) requirements and skillsets need to alter

The ability to review and judge a wider range of diverse evidence requires a revised skillset compared to a pure process-based approach. These include, but are not limited to: the procurement environment, system and platform interaction, and effective industry/customer engagement.



Knowledge/Skills of Stakeholders

Principles of evidence theory need to be restated

Any revised approach requires evidence to be holistically assessed with an understanding gained into *how* evidence can be structured and supported. There is a need to (re-)educate stakeholders in the construction of diverse evidence arguments, *how* to structure evidence, the steps to improve existing data, and also the *theory of evidence*.



Through-Life Appraisal

Through-Life Appraisal

Judgements should be captured for the life of the software/system

There are distinct phases of any system such as the development and in-service periods. These systems may also be subject to regular updates which require frequent judgements to be made on the software evidence. Standards and guidelines evolve, albeit slowly, but the attributes associated with any evidence can remain constant. An example of an evidence attribute is the *contribution* it makes to the wider evidence it supports. Therefore, a mechanism could be created to capture judgements through-life; acting as “snap-shots” to reflect the evidence characteristics. This is particularly relevant as a system and its software ceases to be developed and is maintained as legacy equipment.

Capture the confidence being built rather than only ongoing problems

In-service evidence should be a method to *build* confidence rather than to only question it. Traditionally, any in-service data for a system is used to gain feedback on problem reports to query the belief in the process evidence (indeed, counter-evidence is an important property to capture). However, to actively validate the prior-belief in any process, evidence could be used to maintain, or *improve*, confidence.

“In-service evidence should be a method to build confidence rather than to only question it”

Actively gather metrics to inform diverse evidence

There is a need to actively seek metrics and evidence *throughout the complete life-cycle* of the software. This includes during the development stage. There is a myriad of metrics that can be obtained to support or refute the confidence in a system, e.g. technical suggestions raised in Technical Interchange Meetings (TIMs), which lead to *actual* implemented changes. Opportunities to gather such data should be encouraged.

Ensure the right metrics are informing the right judgements

There is a need to be mindful of the use of metrics, how they shape the safety arguments, and how they alter the confidence being gained. Measuring the “wrong” thing can have unintended consequences [5], for example, for future decisions taken or by the introduction of inherent weaknesses in the confidence argument. There are other dangers with metrics such as *gaming* a quantification-based approach (i.e. where the choice of metrics is deliberately interpreted so as to exploit favourable outcomes) and the difficulties with Subject Matter Experts (SMEs) agreeing metric values.

Managing Evidence Complexity

Management and stakeholder understanding of an increased range and depth of underpinning evidence

Process-based compliance that is benchmarked against standards/guidelines can result in sets of structured findings against defined objectives. However, a diverse evidence approach will make use of a depth and range of evidence with no pre-defined benchmarks. Therefore, the wider evidence needs to have the structure *managed/captured* in a *consistent* manner. There is also a greater reliance on the judgements made by SMEs.



Growth in the solution space requires assistance for decision makers

A correlation may exist between the *diversity* of the evidence and the *quantity* of evidence which is under review. Stakeholders can select a myriad of evidence to support an assurance argument and therefore the potential solution space grows significantly. This larger solution space may be complicated for stakeholders to assess and to determine suitable actions.

“Evidence and its confidence should be driven from the context of the system/software and not just necessarily from a process-based approach”

Communication of evidence to stakeholders is key

Allowing stakeholders to comprehend the evidence and the confidence which can be placed in it is obviously an important element. This allows judgements to be debated and understood. Not all stakeholders who have a vested interest in the outcomes need to have visibility of *all* of the evidence judgements. An abstraction of the information would be of use to gain buy-in and to assist with stakeholder discussions. An approach such as the *Wheel of Qualification* could assist with this [6].

Treatment of Evidence



Not all objectives are equal

Within any standard/guideline, particularly those which are focussed on process conformance, there are unstated degrees of *importance* to each of the objectives. At present, a number of the standards and guidelines that are adopted for software

assurance do not provide information on the weighting of the objectives and therefore give no indication of the priorities or consequences of any shortfalls. Any assurance regime should allow for the priorities or weightings of the objectives to be stated within the extant standards/guidelines. An option is to allow weightings, e.g. *contribution*, to be formally established for objectives and to allow these to be managed and reasoned with.

“At present, a number of the standards and guidelines do not provide information on the weighting of the objectives and therefore give no indication of the priorities or consequences of any shortfalls”

Evidence should be collated to be judged rather than it occurring at distributed stages

Presently, a range of evidence is captured as part of the assurance process and evidence is captured via a *staged* review process rather than making holistic judgements on all relevant evidence. The range and depth of evidence could be significantly increased in any revised assurance approach.

What does “good” look like?

With the adoption of diverse evidence there are a number of standards/guidelines that can inform a view of what an ideal scenario would look like for any given evidence. An example is Capability Maturity Model Integration (CMMI) for process improvement. The metric for the success may not be *full* compliance with a standard/guideline but to allow a measure to be gained of the shortfalls and therefore, the confidence that can be assigned to such evidence. However, diverse and radical evidence that does not have precedence will lack a supporting framework. Such evidence will be significantly reliant on SME judgement.

Need to build evidence from the bottom-up

Due to the novelty of a diverse evidence approach, there is a need for SMEs to understand the *reasoning* for the choice of evidence and the place it has within the assurance argument.

The relevance and weight of the underpinning evidence needs to be ascertained. There is an argument that, in some contexts, the assurance confidence needs to be gained from the *available* evidence and this requires a bottom-up approach. Evidence and its confidence should be driven from the context of the system/software and not just necessarily from a process-based approach.

Diverse evidence associated with a system will be unique and should be treated as such

In essence, each system has unique characteristics; e.g. a level of process-based conformance, a level of in-service data, a level of third-party oversight, etc. This is a change from the pure process-based approach, which doesn't always account for the wider supporting evidence. Therefore, there is a requirement to treat systems as bespoke items with such judgements on the evidence being captured.

Elements of a system, e.g. software, will not be as easily labelled as being standard/guideline compliant

As the solution space opens up, the confidence that is assigned to systems will be derived from non-process evidence. Therefore, the systems and software may not receive a "label" that succinctly states a process-based standard compliance. This is due to there being a wider and diverse set of evidence providing the *confidence*. There should be consideration to remove the labels that are attributed to software, such as "DO-178C DAL B compliant". The concept of "assurance confidence" would be a valid approach to reflect the range of underpinning evidence supporting the target measurement.

Judgment of Evidence

Judgements could be captured with consistent attributes

An assurance approach that uses wider evidence could have the ability to allow *priorities and the purpose of evidence to be managed/captured*. Evidence attributes (e.g. *contribution*) allow the detail of any judgements to be captured and to maintain a consistent narrative to describe the evidence. If a *quantitative* approach is not adopted, the evidence attributes can still drive the judgements and act as prompts for any *qualitative* arguments. A consideration is to also capture the *overheads* (e.g. time, cost, or quality implications) associated with gathering/generating any evidence to achieve a target level of confidence.



A more dynamic evidence landscape requires a defensible position rather than a repeatable one

A move from process-based evidence assessment to one which uses a wider set of evidence, means there is a need to accept, and place value on, the subjective opinion of SMEs. This approach is, arguably, more difficult to measure. Regulatory assurance stances should take an approach to provide further credence to SME judgement. This would allow judgements to be made from a *defensible* position rather than one that is based upon known and accepted benchmarks, i.e. that which is *repeatable*. A *defensible* claim by a stakeholder would be one that is justified with an opinion which can be argued to be *good* [1] (akin to joint reports/submissions regarding differing expert witness perspectives within the criminal justice domain).



Debate and Consensus

Collective judgements need to be captured

An approach underpinned by judgement rather than process conformance may require altered methods for the evidence measurement by stakeholders. A number of subjective stakeholder opinions and their statements of evidence acceptance need to be captured. This information should be able to be debated and reasoned upon, via a suitable structure and method.

Move to a more consensus based approach for evidence judgement and acceptance

To remove any issues regarding single SME judgement, a more consensus driven approach with stakeholder cooperation and approval of the evidence and judgements is needed. As a result, there may be a *joint acceptance* by the stakeholders of the software safety assurance argument.

Procurement and Contracting

Procurement types and stages influence the diverse evidence adopted

There are a number of procurement options for delivering capability with each having merits and demerits. Which option is chosen has consequences in terms of the availability and type of evidence that will be received as part of any procurement. In addition, the stage of the life-cycle, e.g. design, will influence the evidence available and can act as an opportunity to influence any future evidence which is received.



Method to contract for software assurance evidence will need to reflect diverse evidence approach

If systems are treated as unique entities, then the method to gain evidence will also have to be bespoke. With a process-based approach, the evidence requested is very much artefact-driven, e.g. provision of a development plan. However, the *relevant* and *available* evidence (and its context) needs to be understood. Requests to suppliers for the provision of evidence will need to be informed via conversations and an understanding of the software. Suppliers will need to be engaged in a revised manner compared to traditional assurance approaches.

Closing Remarks

This short article aims to highlight a number of considerations if wider diverse software safety assurance evidence is adopted within any revised assurance approach. Any change in stance to how software safety assurance is conducted is certainly not solely reliant on the underpinning evidence. There is a need to look at the systems and processes that support the assurance activities, and this includes how any assurance cases will be articulated to stakeholders [6]. The end-to-end process for undertaking safety assurance needs to reflect *how* any evidence is generated, gathered, and judged.

The authors of this short article have adopted the use of wider diverse evidence within a number of projects and there are challenges with such an approach (e.g. stakeholder buy-in). It certainly isn't an easier route to software qualification; however, there are rewards as the approach, if implemented correctly, can allow systems to be assured efficiently and effectively.

Disclaimer

This article is an overview of UK Ministry of Defence (MOD) sponsored research and is released for informational purposes only. The contents of this article should not be interpreted as representing the views of the UK MOD, nor should it be assumed that they reflect any current or future UK MOD policy. The information contained in this presentation cannot supersede any statutory or contractual requirements or liabilities and is offered without prejudice or commitment.

References

- [1] Collins, "English Dictionary and Thesaurus", 1995.
- [2] RTCA, "DO-178C: Software Considerations in Airborne Systems and Equipment Certification", 2011.
- [3] G. Hanssen, T. Stålhane, and T. Myklebust, "SafeScrum - Agile Development of Safety-Critical Software", 2018.
- [4] C. M. Holloway, "Understanding the Overarching Properties", July 2019.
- [5] R. Ashmore, M. Standish, "The Measurement of Software Maintenance and Sustainment: Positive Influences and Unintended Consequences", CrossTalk Journal, October 2016.
- [6] M.J. Hadley, M. Standish, "Using Tiers of Assurance Evidence to Reduce the Tears! Adopting the 'Wheel of Qualification' for an Alternative Software Safety Assurance Approach", High Integrity Software (HIS) Conference, 5th November 2019.

Mike Standish CEng MBCS MINCOSE

Mike is a senior engineer in systems at the Defence Science and Technology Laboratory (Dstl). Mike has experience of all aspects of software and systems lifecycles, which has been gained in over 15 years within the defence sector. Mike is in the final stages of completing an Engineering Doctorate (EngD) in Systems at the University of Bristol.

Dr Mark Hadley CEng MIET

Mark has been involved in the airborne safety critical software domain for over 20 years with Dstl (and its predecessor organisations) working on a range of UK MOD airborne systems. Mark is a senior principal consultant in software and provides Independent Technical Evaluation (ITE) and SME advice to a host of MOD Project Teams.

DSTL/JA119137. © Crown copyright (2019), Dstl. This material is licensed under the terms of the Open Government Licence except where otherwise stated. To view this licence, visit <https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3> or write to the Information Policy Team, The National Archives, Kew, London TW9 4DU, or email: psi@nationalarchives.gsi.gov.uk