Author:
Standish, Michael

Title:
Enhancing Current Software Safety Assurance Practice to Increase System Mission Effectiveness

This is a redacted version of the full dissertation, as agreed by the candidate, the supervisors and the Defence Science and Technology Laboratory (Dstl) as the industrial sponsor of this Engineering Doctorate studentship in the Faculty of Engineering. The redactions cover information that was deemed too sensitive to be published. The redactions have been kept to the minimum level necessary so that the dissertation can still show the research excellence of the candidate.

[This page intentionally left blank]

Enhancing Current Software Safety Assurance Practice to Increase System Mission Effectiveness

Michael Standish

A thesis submitted to the University of Bristol in accordance with the requirements of the degree of Engineering Doctorate in Systems in the Faculty of Engineering.

Faculty of Engineering

March 2020

Word Count: 86,648

[This page intentionally left blank]

Abstract

To deploy a safety-critical system it is imperative to have confidence in the system's underpinning software. This is gained by performing software safety assurance. If there is not a sufficient level of confidence in the software then there is not a sufficient level of confidence in the system. Therefore, the system would not be able to be deployed in applications where safety is paramount. A traditional method to gain confidence in software is to develop it to a process centred on the life-cycle. This is subsequently judged against a set of predefined objectives and the judgement on the level of compliance to the objectives is taken to warrant a degree of confidence in the software. However, if only certain types of evidence are accepted to demonstrate compliance, e.g. process-based evidence, then the solution space is reduced and some technical solutions potentially excluded.

The aim of the thesis is to provide additional methods and success factors to potentially expand the scope of the current safety assurance processes.

The research has explored how other domains, both safety and non-safety, judge confidence in evidence to make their informed decisions. An approach has been developed which allows attributes to be associated with any form of software safety assurance evidence. The chosen attributes allow the characteristics of the evidence to be understood as well as the relationship of the evidence to other forms of evidence.

A Decision Support Framework (DSF) has been researched and implemented which is a usable end-to-end tool for decision makers to construct diverse evidence assurance arguments. The term 'diversity' in this context is to have a variety of independent evidence which allows overall confidence to be gained for the software safety assurance. Judgements on the evidence characteristics are captured via Fuzzy Logic. The framework also allows the practical implications of adopting any evidence to be considered, such as cost and time. Tools have been devised to visualise confidence propagation across different collections of various evidence types. Efficient and effective evidence solutions can be identified by using the tool's Genetic Algorithm (GA) optimisation techniques.

Diverse evidence can be challenging to measure when compared to traditional process-based approaches. However, the research has devised solutions to ameliorate such difficulties via outputs such as a stakeholder communication model. The DSF has been implemented on a number of defence case studies to understand the non-trivial way in which evidence attributes combine and how to construct a diverse evidence approach.

This research has demonstrated how the use of diverse evidence can achieve an equivalent level of compliance to a full process-based approach and therefore that it can form part of a software safety assurance strategy. The research outputs have not previously been implemented within the software safety assurance domain prior to this research.

[This page intentionally left blank]

Acknowledgements

Dr John May, University of Bristol For the constructive discussions, insightful advice, and for guiding me through the academic writing process.

Dr Mark Hadley, Dstl For the perceptive comments, continual motivation, and guidance during my studies and throughout the years.

Dr Theo Tryfonas, University of Bristol For the valuable feedback and support.

Dstl Colleagues For being supportive throughout my studies.

Systems Centre Support Staff, University of Bristol For the pragmatic and timely help throughout the whole of my time undertaking the EngD.

My Family For the endless encouragement and for being so tolerant of the amount of time I've been away. I owe you all much more of my time.

[This page intentionally left blank]

Author's Declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's *Regulations and Code of Practice for Research Degree Programmes* and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE:.....

[This page intentionally left blank]

Research Publications/Conferences

The research has resulted in a number of outputs.

1. External Conferences/Seminars.

   (a) Institution of Engineering and Technology (IET) System Safety and Cyber Security (October 2014): *Safety Case Development: A Process to Implement the Safety three-Layered Framework.* Authors: M. Standish, H.J. Auld, P.R. Caseley, and M.J. Hadley.

   (b) IET System Safety and Cyber Security (October 2014): *The Safety three-Layer Framework: A Case Study.* Authors: M. Standish, H.J. Auld, P.R. Caseley, and M.J. Hadley.

   (c) Safety-Critical Systems Club Seminar: Use of Service History and Field Data - In Support of Safety Justifications (April 2016): *Field Service Evidence: Practical Examples and Potential Refinements to Support Qualification Arguments.* Authors: M. Standish, M.J. Hadley, and E. Lennon.

   (d) High-Integrity Software Conference (November 2016): *Multi-Core (MC) Processor Qualification for Safety Critical Systems.* Authors: M.J. Hadley and M. Standish.

   (e) High-Integrity Software Conference (November 2019): *Using Tiers of Assurance Evidence to Reduce the Tears! Adopting the 'Wheel of Qualification' for an Alternative Software Safety Assurance Approach.* Authors: M.J. Hadley and M. Standish.

2. Ministry of Defence (MOD) Seminars.

   (a) CIS Assurance Group Seminar (July 2018): *The Cake of Alternative Software Safety Evidence: Getting the Ingredients Right and Modifying the Recipe.* Author: M. Standish.

3. Journals/Newsletters.

   (a) CrossTalk. The Journal of Defense Software Engineering (Vol. 29 No. 5, Sep/Oct 2016): *The Measurement of Software Maintenance and Sustainment: Positive Influences and Unintended Consequences.* Authors: R. Ashmore and M. Standish.

   (b) Safety-Critical Systems Club - Safety Systems Newsletter (Vol. 28 No. 1, Feb 2020): *Adapting to Changes in a Software Safety Assurance Approach.* Authors: M. Standish and M.J. Hadley.

4. MOD Formal Customer Reports. The research has directly informed a number of customer provided reports.

   (a) *An Argument for the Adoption of Diverse Software and Complex Electronic Hardware (CEH) Evidence Within a Qualification Strategy.* Authors: M. Standish and M.J. Hadley.

   (b) *[Airborne Platform] Safety-Related Programmable Element (PE) Qualification Strategy.* Authors: M. Standish and M.J. Hadley.

5. MOD Educational White Papers.

   (a) *Use of Diverse Software Evidence within a Safety-Critical Software Airborne Qualification Strategy.* Authors: M. Standish and M.J. Hadley.

6. University of Bristol Industrial Doctorate Centre (IDC) in Systems Events/Reviews.

   (a) IDC in Systems 6th Annual Conference (June 2015): [Poster] *Enhancing Software Safety Assurance For UK MOD.* Author: M. Standish.

   (b) IDC in Systems Engineering Doctorate (EngD) 2nd Year Review (September 2016): [Presentation] *Enhancing Software Safety Assurance For UK MOD.* Author: M. Standish.

   (c) IDC in Systems 7th Annual Conference (June 2016): [Poster] *Enhancing Software Safety Assurance For UK MOD.* Author: M. Standish.

   (d) IDC in Systems 8th Annual Conference (June 2017): [Poster] *Enhancing Software Safety Assurance For UK MOD.* Author: M. Standish.

   (e) IDC in Systems EngD 4th Year Review (June 2018): [Presentation] *Enhancing Software Safety Assurance For UK MOD.* Author: M. Standish.

The Engineering Doctorate

This thesis has been completed as part of an EngD within the University of Bristol's IDC in Systems Programme. The premise of an EngD is similar to that of a traditional PhD (a doctorate in philosophy) in that both produce a distinct 'contribution to knowledge'. The following is based upon published information from the Association of Engineering Doctorates (AEngD) (AEngD, 2018)[1]:

- The EngD provides a more vocationally oriented doctorate in engineering than the traditional PhD and is claimed to be better suited to the needs of industry.
- It combines academic research in an industrial context with taught modules in related subjects.
- The EngD is a doctorate, equivalent to a PhD. However, the EngD student - or Research Engineer (RE) - pursues a research project while based within a company.
- The EngD is a qualification for practising researchers who aim to lead and innovate the development of products, processes, and services in industry. REs develop academic strengths and leadership skills, both technical and managerial.
- All time spent on EngD programmes is recognised by relevant institutions as contributing towards Chartered Engineer (CEng) status.
- The EngD programme merges experience of industrial involvement (and today's real-world issues) with a doctorate's immersion in professionalism. The result: a unique individual with relevant industry knowledge and academic depth, grounded by up to four years of experience.

[1]Further information on EngDs can be found at http://www.aengd.org.uk/ with specific information on the University of Bristol's IDCs EngD found in the IDC handbook (IDC in Systems, 2013).

[This page intentionally left blank]

Contents

List of Figures

List of Tables

Abbreviations

ABS Anti-Lock Braking System.

ADIRU Air Data Inertial Reference Unit.

AED Aviation Engineering Directorate.

AEL Assurance Evidence Level.

AEngD Association of Engineering Doctorates.

ALARP As Low As Reasonably Practicable.

AMC Acceptable Means of Compliance.

AMRDEC Aviation and Missile Research, Development, and Engineering Center.

ANSP Air Navigation Service Provider.

ARP Aerospace Recommended Practice.

ASCE Assurance and Safety Case Environment.

ASIC Application-Specific Integrated Circuit.

ATC Air Traffic Controller.

ATM Air Traffic Management.

ATS Air Traffic Service.

AWE Atomic Weapons Establishment.

BA Breeding Algorithm.

BBN Bayesian Belief Network.

BIT Built-In-Test.

BMJ British Medical Journal.

BSc Bachelor of Science.

CAA Civil Aviation Authority.

CADMID Concept, Assessment, Demonstration, Manufacture, In-Service, Disposal.

CAE Claims, Arguments and Evidence.

CAST Certification Authorities Software Team.

CBA Cost Benefit Analysis.

CCA Common Cause Analysis.

CCF Common Cause Failure.

CCTV Closed-Circuit Television.

CDO Coordinating Design Authority.

CE Counter-Evidence.

CEBM Centre for Evidence-Based Medicine.

CEH Complex Electronic Hardware.

CEng Chartered Engineer.

CGI Consultants to Government and Industries.

CLE Clearances with Limited Evidence.

CM Configuration Management.

CMMI Capability Maturity Model Integration.

COCOMO Constructive Cost Model.

COTS Commercial-Off-The-Shelf.

CPS Crown Prosecution Service.

CPU Central Processing Unit.

CRAN Comprehensive R Archive Network.

CSF Critical Success Factor.

CSV Comma-Separated Values.

DAAA Direzione degli Armamenti Aeronautici e per l'Aeronavigabilità.

DAE Defence Air Environment.

DAG Directed Acyclic Graph.

DAL Design Assurance Level.

DAOS Design Approved Organization Scheme.

DE&S Defence Equipment and Support.

DGAM Dirección General de Armamento y Material.

DGC Dependency-Guarantee Contract.

DGR Dependency-Guarantee Relationship.

DHFLGDM Dynamic Hesitant Fuzzy Linguistic Group Decision-Making.

DLSR Defence Land Safety Regulator.

DMR Defence Maritime Regulator.

DNA Deoxyribonucleic Acid.

DO Design Organisation.

DS Defence Standard.

DSA Defence Safety Authority.

DSAÉ Direction de la Sécurité Aéronautique d'État.

DSF Decision Support Framework.

DSP Delivery Support.

DSS Decision Support System.

DST Dempster−Shafer Theory.

Dstl Defence Science and Technology Laboratory.

DT Delivery Team.

DWP Department for Work and Pensions.

EASA European Aviation Safety Agency.

EBM Evidence-Based Medicine.

EBPM Evidence-Based Policy Making.

ECT Existing Certification.

ECU Engine Controller Unit.

EDA European Defence Agency.

EngD Engineering Doctorate.

ER Evidential Reasoning.

EU European Union.

EUROCAE European Organization for Civil Aviation Equipment.

FAA Federal Aviation Administration.

FACE Future Airborne Capability Environment.

FBW Fly-By-Wire.

FCPC Flight Control Primary Computer.

FHA Functional Hazard Assessment.

FIS Fuzzy Inference System.

FL Fuzzy Logic.

FM Fuzzy Model.

FPGA Field-Programmable Gate Array.

FRBS Fuzzy Rule-Based System.

FSM Finite State Machine.

FTA Fault Tree Analysis.

GA Genetic Algorithm.

GO-Science Government Office for Science.

GRADE Grading of Recommendations Assessment, Development and Evaluation.

GSN Goal Structured Notation.

HI High-Integrity.

HIS High-Integrity Software.

HITL Human-in-the-Loop.

HMCR Harmony Memory Considering Rate.

HOTL Human-on-the-Loop.

HP Hewlett-Packard.

HPC High Performance Computing.

HPM Hierarchical Process Modelling.

I/O Input/Output.

I&C Instrumentation and Control.

IAWG Industrial Avionics Working Group.

IDC Industrial Doctorate Centre.

IEC International Electrotechnical Commission.

IEEE Institute of Electrical and Electronics Engineers.

IET Institution of Engineering and Technology.

IMA Integrated Modular Avionics.

IP Intellectual Property.

IPR Intellectual Property Rights.

ISA Independent Safety Auditor.

ISSS Information Set Safety Summary.

ITAR International Traffic in Arms Regulations.

ITE Independent Technical Evaluator.

KSLOC Thousands of Source Lines of Code.

LC Life-Cycle.

LM Lockheed Martin Corporation.

LRU Line Replaceable Unit.

LSSR Land Systems Safety Regulator.

LufABw Luftfahrtamt der Bundeswehr.

MAA Military Aviation Authority.

MAWA Military Airworthiness Authority.

MC Multi-Core.

MC/DC Modified Condition/Decision Coverage.

MCRI Military Certification Review Item.

MF Membership Function.

MISRA Motor Industry Software Reliability Association.

MOD Ministry of Defence.

MOTS Military-Off-The-Shelf.

MR Mutual Recognition.

NAG Naval Authority Group.

NAN Naval Authority Notice.

NATCS National Air Traffic Control Services.

NATS National Air Traffic Services.

NHS National Health Service.

NICE National Institute for Health and Care Excellence.

NIST National Institute of Standards and Technology.

NoI Node of Interest.

OC Overall Confidence.

OEC Operational Emergency Clearances.

ONR Office for Nuclear Regulation.

PAR Pitch Adjusting Rate.

PBL Performance Based Logistics.

PDS Previously Developed Software.

PE Programmable Element.

PLD Programmable Logic Device.

PM Project Manager.

PSAC Plan for Software Aspects of Certification.

PSE Product Service Experience.

PSH Product Service History.

PSO Particle Swarm Optimization.

PSS Products, Services and/or Systems.

QA Quality Assurance.

QMS Quality Management System.

R&D Research and Development.

RA Regulatory Article.

RAF Royal Air Force.

RCT Randomised Controlled Trial.

RE Research Engineer.

RFI Request For Information.

RGP Recognised Good Practice.

RMF Risk Management Framework.

SA Simulated Annealing.

SAP Safety Assessment Process.

SAR Safety Assessment Report.

SCR Safety Case Report.

SDP Software Development Plan.

SECT-AIR Software Engineering Costs and Timescales – Aerospace Initiative for Reduction.

SIL Software Integrity Level.

SIP Software Integrity Policy.

SLF Safety three-Layered Framework.

SLOC Source Lines of Code.

SME Subject Matter Expert.

SMS Safety Management System.

SOC System-on-Chip.

SofS Secretary of State.

SOP Standard Operating Procedure.

SoS System-of-Systems.

SOUP Software of Unknown Pedigree.

SQEP Suitably Qualified and Experienced Personnel.

SRA Security Related Airworthiness.

SRS Safety Related Software.

SSA System Safety Assessment.

TCB Type Certificate Baseline.

TIM Technical Interface Meeting.

TSO Technical Standard Order.

TST Testing.

UAV Unmanned Aerial Vehicle.

UK United Kingdom.

UML Unified Modelling Language.

US United States.

USA United States of America.

V&V Verification and Validation.

VCA Vehicle Certification Agency.

WAM Weighted Average Method.

WCET Worst-Case Execution Time.

WOW Weight-on-Wheels.

[This page intentionally left blank]

Chapter 1

Introduction

1.1 Context

An aircraft is comprised of many interconnected systems, of varying complexity[1], which provide the fundamental functionality for the aircraft's operation[2]. Many of these systems will perform mission[3], security[4], or safety[5] critical roles and the system's functionality is commonly underpinned by software[6]. If there is a *failure* in the software then there can be a *failure* for the system to perform its function. Unfortunately, there are notable past software failures which have occurred within a range of systems and domains; examples include, but are not limited to, the medical domain[7] and the military domain[8].

To deploy a safety-critical system it is imperative to have *confidence*[9] in the system's underpinning software and this is gained by performing software safety *assurance*[10]. If there is not a sufficient level of confidence in the software then there is not a sufficient level of confidence in the system (and therefore the overall platform). The amount of software that requires suitable *assurance* is growing. There are various estimates stating the rate of growth

[1]For example, from simpler mechanical flight control systems which are common in smaller aircraft to Fly-By-Wire (FBW) control systems which require a flight computer to convert the pilot's intended actions to the movement of aircraft actuators etc.

[2]For example, the flight control of the aircraft (direction and speed), fuel flow control for the propulsion system, and the ability to navigate etc.

[3]Failure may prevent or degrade operation (Stevens et al., 2019).

[4]Failure adversely impacts confidentiality, integrity, or availability (Stevens et al., 2019).

[5]Failure may lead to damage (Stevens et al., 2019).

[6]For example: the flight computer within a FBW flight control system.

[7]Such as the Therac-25 Medical Accelerator in 1985-87. Further information can be found within Leveson (1995).

[8]Such as the Airbus A400M in 2015. Further information can be found within DSIWG (2018).

[9]The term *confidence* provides "trust in a thing" and "showing [a level of] certainty" (Collins Dictionary, 1995*b*).

[10]*Assurance* is "a positive declaration intended to give confidence" (OED, 2018*a*). The distinction between the two terms, *confidence* and *assurance*, in this context is subtle.

of software within avionic systems; e.g. approximately 400% every 2 years (Carlson, 2016), with safety-related code doubling in size every 4 years (Zolotas et al., 2017). For military avionics there is a similar trend of growth (AVSI, 2011). This increase not only has cost implications but also increases the complexity of the software development and its subsequent management.

A traditional way to gain confidence in the software is to develop it to a process. In this context, a process-based approach is one which is centred on the *qualitative* aspects of a *life-cycle*; such as the *waterfall* model which includes stages for requirements, design, and implementation. In essence, process evidence is concerned with the *intent* to build the software right (Hadley and White, 2008). Therefore, non-process evidence includes, but is not limited to, the *quantitative* aspects of a life-cycle (for example testing results). This is in addition to much wider sets of evidence which can be relevant for pre- and post-release phases such as in-service data and reliability modelling. Life-cycle evidence can be measured with a set of predefined objectives[11]. The amount of rigour which is needed for the development of the software is determined by the scope and detail of the predefined objectives; e.g. DO-178C has 71 objectives for any software where a failure could lead to a *catastrophic* event[12] (known as level A). There is a scale of the levels (A-D) to reflect the severity of the software failure. Level D, with 26 objectives, is associated with software where a failure could lead to a *minor* event[13]. The number of objectives reduces with each level: A (71), B (69), C (62), and D (26). The software evidence[14] is judged to ascertain the degree of *compliance*[15] against these predefined objectives. Understanding the amount of *compliance* leads to having a level of *confidence* in the software.

There is debate within the software safety assurance community on the strongest forms of evidence to gain confidence in software. It has been stated that "there is no evidence that a good process will result in a good product (although there is a correlation between bad processes and bad products!)" (Menon, Hawkins and McDermid, 2009*b*). Also, it is the "product that runs and it is therefore the product evidence that provides a direct assessment of that which can fail and give rise to the hazards" (McDermid, 1998). There are other factors not specifically captured in any process-based objectives which influence

[11]For example, objectives are stated within the Motor Industry Software Reliability Association (MISRA) development guidelines for automotive software (MISRA, 2012) and DO-178C for avionics software (RTCA, 2011*a*).

[12]A *catastrophic* event could have "multiple fatalities (usually with loss of the aircraft)".

[13]A *minor* event could "reduce the capability of the aircraft or the ability of the crew to cope with adverse operating conditions to the extent that there would be, for example, a significant reduction in safety margins or functional capabilities".

[14]There are numerous definitions of the term *evidence*; an apt one in this context is that evidence is to enable a *premise for belief* to be held (Stanford Encyclopedia of Philosphy, 2014). The *belief* being that the software meets, for example, the predefined objectives to achieve a level of safety assurance.

[15]To *comply* is to "act in accordance with rules, wishes etc" (Collins Dictionary, 1995*a*).

the confidence in the software[16]. Also, some systems may not have *compliant* process-based evidence available[17]. Therefore, in addition to a process-based approach, there are other methods which can gain suitable levels of confidence in the software, e.g. the number of fault free hours that the software has been in-service on the target systems[18].

Software can belong to one of three categories (see Table 1.1) and this dictates the type of evidence available (UK MOD, 2018). The categories are applicable to all software, from previously developed, i.e. legacy[19], to software which is novel[20].

Category	**Description**
Black-box	Little or no information about the internal workings of the software is available.
White-box	Internal workings, such as the original source code, is available.
Open-box	Not only is the source code driving the software known but it can be adapted.

Table 1.1: Software Information Availability Categories (UK MOD, 2018)

The functionality delivered by a system, or set of systems, will have supporting evidence to provide the safety assurance for its software. Any solution needs to meet the technical *and* assurance requirements; e.g. for the selection of systems to support an avionics architecture, such as Integrated Modular Avionics (IMA)[21], there will be a number of key features which influence the choice of technical solutions; e.g. real-time and safety-constraints. For a system to be considered within a wider technical solution it must have evidence available to demonstrate that the safety constraints can be met. If there is only a certain type of accepted evidence to demonstrate compliance, e.g. process-based evidence, then the solution space is reduced and some technical solutions potentially excluded. This can also lessen the mission

[16]For example: independent Verification and Validation (V&V) by third-parties and any Quality Management Systems (QMSs) in place.

[17]Due to, for example, the software being developed to another life-cycle standard or the process-based evidence not being releasable to those that need to gain the assurance.

[18]With the confidence being established by using guidance such as CAST (1998). Factors such as software error reporting will determine this confidence as it is not purely based on the in-service hours. The premise is that the greater the number of fault free in-service hours then the greater the level of confidence gained in the software. It should be noted that confidence from in-service hours is applicable to a known and constrained system environment, i.e. one which has identical input states. The applicability of the evidence to known environments is also true for software *process* evidence as a change in the operating environment will result in different reliabilities being exhibited.

[19]With the term *legacy* in this context being "of, relating to, or being a previous... computer system" or "of, relating to, associated with, or carried over from an earlier time, technology etc" (Merriam-Webster, 2018).

[20]*Novel* technology is defined as an approach, or item of equipment, which has not undergone any form of United Kingdom (UK) military airborne assurance (this definition is partly derived from Weaver, Kelly and Mayo (2006)).

[21]IMA architectures provide a shared computing, communications, and Input/Output (I/O) resource pool that is partitioned for use by multiple avionics functions (Watkins and Walter, 2007).

effectiveness of any wider solution, i.e the ability for the end goals to be achieved[22].

The use of diverse[23] evidence which could achieve an *equivalent*[24] level of safety assurance compliance *could* assist by expanding the technical solution space and improving the mission effectiveness[25]. This expanded solution space is illustrated simplistically within Figure 1.1. Whilst recognising that process-based evidence is an important part of any safety assurance activities there may be scope to include diverse evidence as part of the overall judgement. This would be suitable, for example, in circumstances in which full process-based evidence was not available.

There is a great reliance on expert judgement[26] to assess these software-based systems (Littlewood and Wright, 2007). Judgements are made on a number of factors, including: the degree of evidence compliance (e.g. as with the process-based objectives); the suitability of any in-service evidence, and the degree of conformance to other measures (e.g. QMS). It is also subjective expert judgements which dictates the strength and contribution of the evidence. In essence, judgements are formed both in terms of the *intra-* and *inter*-evidence; *intra-evidence* due to each form of evidence being judged as a single entity (e.g. its own *quality*) and *inter-evidence* as each form of evidence is judged in relation to other evidence (e.g. the comparison to judge *sufficiency*).

This leads to a number of opportunities to investigate if any enhancements can be made to the current software safety assurance practices[27]; e.g. can diverse evidence (which is not only process-based) provide SMEs with a suitable level of safety assurance? If evidence is accepted (which is not process-based) then how can it be judged to gain confidence? This may involve the quantification of such confidence. The current research on the quantification of assurance confidence has a number of identified weaknesses[28]. There is a need to understand and enhance how diverse evidence can be gathered, judged, and implemented within the

[22]This includes the function that a platform or System-of-Systems (SoS) should perform.

[23]With the term *diverse* meaning to be distinct and to have variety (Collins Dictionary, 1995*i*).

[24]In comparison to a full process-based approach.

[25]The need for safety requirements to inform technical solutions is critical and there are a number of areas which are looking at methods to balance these requirements. Examples of such areas include: Multi-Core (MC) processors within the United States (US) civil airborne domain (FAA, 2017); security and safety requirements restricting features within an avionics architecture (Brosgol and Smith, 2018); the need for a balanced program to optimise safety, performance, and cost (FAA, 2000); and securing embedded systems without jeopardising safety-properties (SEI, 2019).

[26]The judgements are being formed by Subject Matter Experts (SMEs).

[27]The use of the term *enhance* is defined as to "*further* improve" (OED, 2018*c*). The current safety assurance practice is robust when considered in the context of a *greenfield* project with process-based software evidence which is compliant to well recognised and understood standards. The aim of the thesis is to provide *additional* methods and success factors to potentially expand the scope of the safety assurance process. This concept is important to note as the context of any suggested enhancements are to expand upon an existing *robust* safety assurance process.

[28]For example, the inability to scale up to account for wider sets of evidence. A review of current research on the topic of quantifying assurance confidence is contained within sub-section 4.2.

military software assurance domain.

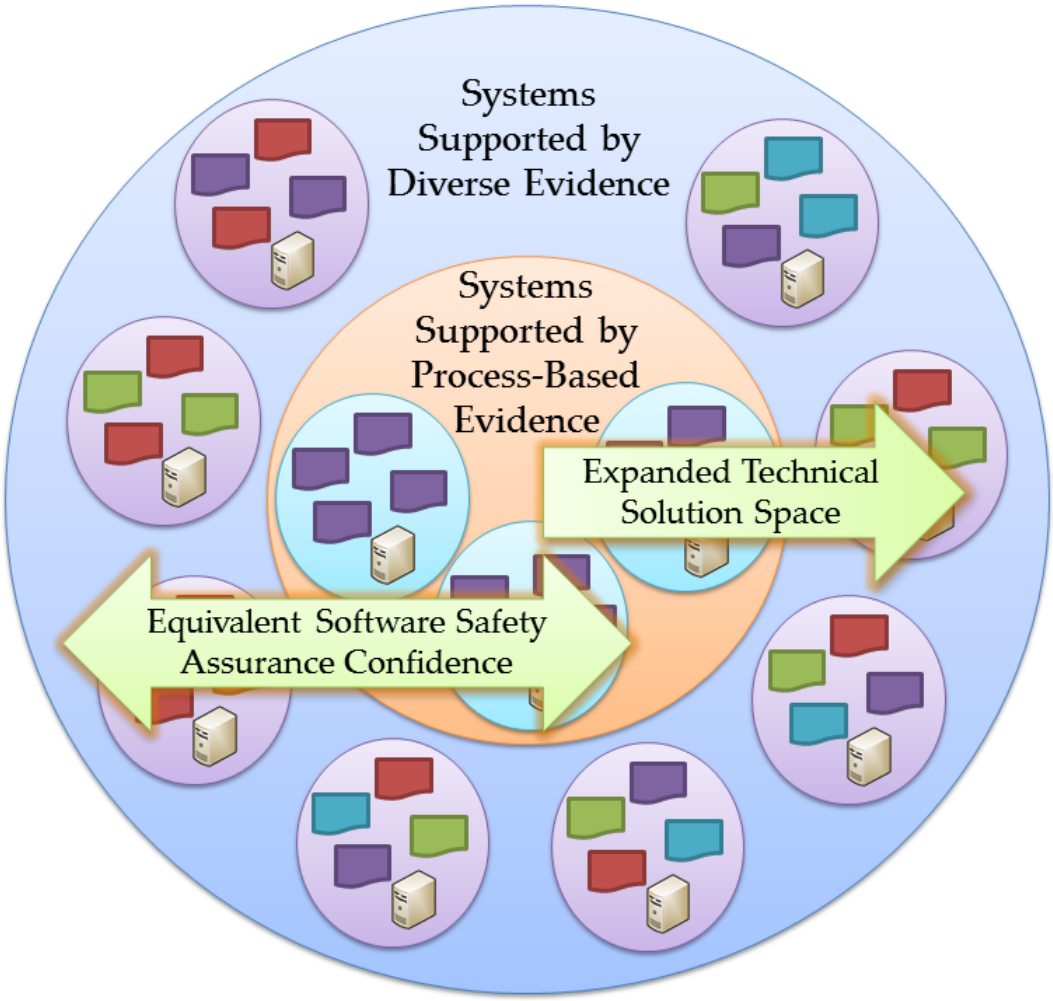


Figure 1.1: Expanded Solution Space Occupied by Systems with Diverse Evidence

1.2 Problem Statement

As safety-related systems will contain increasingly more software and are to become ever more reliant on this software, it is imperative that the software can be *assured*. This allows those that regulate, procure, and operate the software to have confidence that any software failures which lead to *damage* only occur at acceptable rates. *What* different types of evidence are suitable to gain this confidence and *how* should this evidence be structured and judged? If suitable approaches are defined then *how* should any identified military software assurance domain *enhancements* be implemented?

1.3 Research Questions

The thesis is centred upon responding to the following research *grand tour*[29] question:

What enhancements can be made to the current UK defence domain's software safety assurance approaches for capturing and judging supporting evidence?

The supporting research sub-questions are as follows:

1. *What is the current approach to system safety assurance within the UK defence domain and are there alternative system-level approaches?*[30]
2. *What is the current permissible software safety assurance evidence within the UK defence domain and related domains?*[31]
3. *What software safety assurance evidence is relevant/admissible and what are the underpinning principles for the use of such evidence?*[32]
4. *What are the unintended consequences of adopting incorrect metrics when forming decisions and how can system/evidence relationships be communicated to stakeholders?*[33]
5. *What is a suitable structure for software safety assurance evidence and can mathematically derived approaches inform how judgements are made on the evidence and for proposing alternative/optimised solutions?*[34]
6. *What observations and recommendations can be made on how to implement a software safety assurance evidence argument and how to inform a UK defence software safety assurance strategy?*[35]

1.4 Thesis Structure

Following this introduction the thesis comprises ten chapters. There are three broad themes: the argument for an intervention[36] to be made (*why* the research is needed); the justifications made for the research strategy (the *building blocks* for the research); and the execution of the research plan (to create the research *findings*).

[29] A *grand tour* question is the overall and general question which is answered via sub-questions.
[30] Sub-question responded to within Chapter 3.
[31] Sub-question responded to within sub-sections 4.2, 5.2, 5.3, and Chapter 6.
[32] Sub-question responded to within sub-sections 7.1 and 7.2.
[33] Sub-question responded to within sub-sections 7.3 and 7.4.
[34] Sub-question responded to within sub-section 4.3, Chapter 8, sub-sections 9.4 and 9.5.
[35] Sub-question responded to within sub-section 9.6 and Chapter 10.
[36] An *intervention* in this context is to take decisions or perform a role to determine events.

- *Chapter 2; Research Strategy.* Provides a description of the research strategy, why the particular research area was chosen, and the initial exploration of the research, e.g. stakeholder selection. A summary is provided of the steps taken to *execute* the research strategy.

- *Chapter 3; Background and the Problem of Interest.* States the need for software to be assured, the initial scope of the problem of interest, and information on the aspects of a safety case as part of an assurance activity. Context is provided to the safety management terminology, safety case practice, and how a leading MOD assurance standard is applied (Defence Standard (DS) 00-56 (UK MOD, 2014*c*)). Information on what constitutes a Programmable Element (PE) is provided with details on how PEs are assessed within the MOD. A SLF is described which provides a mechanism to help inform a safety argument for a SoS. The approach is beneficial in circumstances where there is limited system information, e.g. due to Intellectual Property Rights (IPR)

- *Chapter 4; Diversity as a Concept and Scope for Further Investigation.* Diverse evidence is discussed in terms of its definition, how it is used in other domains, and the concept of software *design* diversity is also briefly discussed. An assessment of current confidence quantification methods is provided. An outline is provided on how a DSF could assist with a solution to provide enhancements to current software safety assurance methods and the benefits which a framework may provide.

- *Chapter 5; A Review of the Use of Evidence Within Non-Safety Domains.* The definitions applied to the term *evidence* are discussed. Lessons are identified from how evidence is adopted and assessed within non-safety domains, e.g. criminal justice system.

- *Chapter 6; Current Permissible Evidence for Safety-Critical Software Assurance.* Review of the evidence which is currently adopted for software assurance arguments within the MOD (land, maritime, and air domains) and other safety-critical domains, e.g. civil nuclear. Literature/guidelines are stated which add to the sources of diverse evidence.

- *Chapter 7; Potential Permissible Evidence, Underpinning Principles, and Stakeholder Engagement.* Potential evidence is reviewed and a number of fundamental principles are stated which should be considered for any diverse evidence assurance argument. An insight is provided into the unintended consequences of using incorrect metrics and demonstrates the need to choose metrics for software assurance intelligently. A

visualisation/model is described which allows stakeholders to comprehend and debate a varied set of evidential data/sources with a view to drive *improved* decision making, via a *Wheel of Qualification*.

- *Chapter 8; Framework Design and Implementation Decisions.* The key tenets of the DSF design are explained. Implementation details of the DSF are provided with the justifications of the decisions made.

- *Chapter 9; Case Studies, Exploratory Testing, and Evaluation of the DSF.* The aim and purpose of the case studies are stated. Descriptions are provided of the case studies which will be adopted to demonstrate the value of the DSF and the use of diverse evidence. A proposed evidence assessment flow is described which is intended to allow for the initial review and assessment of diverse evidence by SMEs. The outcomes are described for when the DSF is used to gather diverse evidence with details on *how* the features of the DSF were used. Observations from the case studies and the exploratory testing are made. Observations are focussed on the relationships between the attributes of the evidence and how changes are propagated. An assessment of the implemented DSF is made with a comparison to the related research.

- *Chapter 10; Recommendations to Enhance Current Software Safety Assurance Processes.* Details of a number of enhancements that could be made to the currently defined permissible software-related evidence and the subsequent safety assessment process for MOD. A summary is provided of guidance written for Defence Equipment and Support (DE&S) Delivery Team (DT) Desk Officers which can assist DTs with their procurement approaches to gain diverse evidence. A review is conducted of the challenges for the adoption of the findings.

- *Chapter 11; Research Review and Contributions to Knowledge.* Details the original argument for the REs intervention and the progress is judged against the research questions. A review is conducted to assess such aspects as the research quality and validity to state the legitimacy of the research. Reflections are provided on the research itself, any limitations and further work are described, with information on the researches publications/conferences. The contributions to knowledge that the research has made are stated. The implications of the research findings are also provided along with the industrial impact that the research has had.

Chapter 2

Research Strategy

Research is undertaken in a variety of domains, e.g. business strategy and healthcare etc, with each form of research being conducted under different contexts and constraints. Due to this, the purpose for any research will differ and hence so will it's definition. However, for this thesis Collis and Hussey (2009) provide a suitable definition, in that research:

- Is a process of enquiry and investigation.
- Is systematic and methodical.
- Increases knowledge.

Collis and Hussey (2009) also state that research should be purposeful as it is conducted with a view to achieving an outcome. This is supported by Saunders, Lewis and Thornhill (2012) in that research is conducted to *find things out*. It is important to note that the process of research is continually iterative with overlapping stages (Saunders, Lewis and Thornhill, 2012) - this is a reflection of the *reality* of research.

A key element to any research is that the findings are underpinned by a reliable approach which takes into account the paradigm[1] and for suitable data collection/analysis techniques to be used.

This chapter will provide information on:

- *Research Structure*. Provide the structure and outline of the research strategy to be implemented.

[1] A research paradigm is classed as a *philosophical framework* that guides how scientific research should be conducted (Collis and Hussey, 2009). As an aside, from the literature that the RE has read there is an inconstant use of language within the research strategy domain when referring to philosophies and paradigms etc. As an example, Collis and Hussey (2009) refer to *positivism* as a paradigm whereas, Saunders, Lewis and Thornhill (2012) refers to it as a *philosophy* with paradigms being associated with social theory analysis (e.g. *radical humanist*).

- *Research Area and Initial Exploration.* Why the particular research area was chosen and the initial exploration of the research, e.g. stakeholder selection.
- *Research Execution.* A summary of the research execution steps which continue the implementation of the research strategy.

2.1 Research Structure

Research structures are commonly presented in a linear or 'waterfall' flow with one event leading to the next. This implies that at each stage there are clear *exit criteria* from each of the stages which are correct first time. In reality this is not the case, and the nature of research means that iterative refinement is common. This is especially true with the creation of the research questions, for example, which require an understanding of the subject area with further refinement as more knowledge is gained.

The structure in Figure 2.1 shows the flow of activities to articulate the *reality* of the specific research stages (e.g. review of software assurance practice) relevant to the problem of interest. In addition, the structure shown in Figure 2.1 has been refined to reflect the *final* flow of activities to arrive at the thesis findings.

The research structure is composed of two overall themes. The first is the *research exploration and planning* which involves scoping the domain and identifying stakeholders of interest etc (stages 1 and 2 within Figure 2.1). The theme also involves the articulation of the research questions which informs the research approach and the data collection activities. The decisions to inform these stages are justified within sub-sections 2.2 and 2.3. The second overall theme is the *research execution* which is implementing the defined research questions. It is this theme which leads to the research conclusions and recommendations. The decisions made in each of the stages within this *research execution* theme are justified within each of the chapters that follow as appropriate analysis is conducted[2].

2.2 Research Area and Initial Exploration

The activities to understand the research area and the initial exploration of the research form the first stage within Figure 2.1. This stage identifies how the influence/power of stakeholders should be used, which stakeholders are of interest, and determines the initial boundary for the problem.

[2]For example, suitable justification is articulated for the chosen method for reasoning under uncertainty (see sub-section 8.2.4) and the visualisation approach (see sub-section 8.2.6.1).

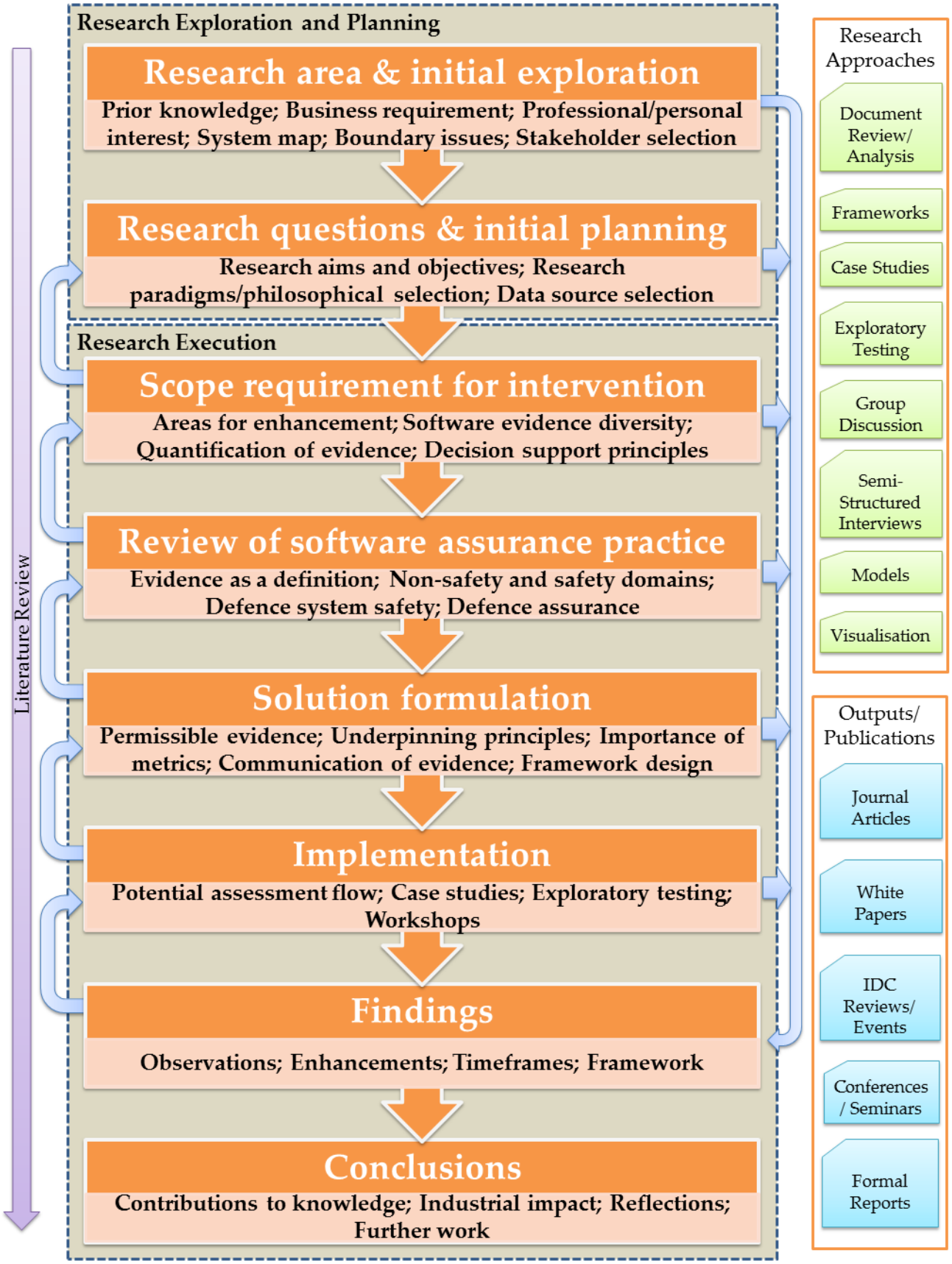


Figure 2.1: Stages of the Research Strategy

2.2.1 Ability to Instigate Change

2.2.1.1 Concept of Influence and Power

Generally, researchers wish to perform activities which result in high impact outputs which have broad ranging consequences. This is natural, in that people wish to make improvements to systems which they believe would benefit from change. However, the *ability* for change to occur, based upon any research outputs, is crucial. If influence cannot be applied by any stakeholder to directly implement the research findings then the value of the research is limited.

In the world of business and organisational management the concepts of *influence* and *power* are recognised as distinct concepts with each allowing various levels of change to occur[3]. In the context of this research the concepts are defined as:

- *Influence*. Provides a catalyst or a force to allow others to change (or for those being influenced to enact change).
- *Power*. There are two forms of *power*: the power to *influence* and the power to enact *change*.

Both concepts are relevant for this research. Typically, REs and those that conduct research will only have the ability to *influence* those that can enact changes. Thus, the *impact* of any research may be limited; although the research may be credible and robust. The RE for this thesis is fortunate in that there is a level of *power* which can be enacted to enable elements of the research findings to be adopted. This *power* is gained due to the nature of an EngD (it is conducted *within* an industrial setting) and due to the role that the RE has within the sponsoring organisation[4].

The *power* is not gained via a management or authoritative role but more from a trusted technical position with the ability to adopt findings from the research. The findings can then feed into the advice provided to customers[5].

2.2.1.2 MOD Organisational Hierarchy

The level of influence and the level of power to enact change needs to be placed in the context of the organisational hierarchy of MOD. The influence/power needs to be targeted at the correct level of the hierarchy with the *ability* to do this being key.

[3]From a management perspective these concepts are described within a range of sources, such as Kramer and Neale (1998) and the famous text by Carnegie (1982).

[4]The involvement of the researchers within some forms of research, such as *qualitative*, is seen as essential and inevitable (Leung, 2015).

[5]The term *advice* in this context is used to described *all* interactions with customers which includes formal reports, verbal communication, and interactions at workshops etc.

Within the business and management domain there is a need to consider the different levels within an organisation[6]. Within the business and management literature there are three well understood levels of an organisation: *strategic*, *tactical*, and *operational* (Griffin, 2007). In the context of this research these levels are defined as:

- *Strategic*. Those with *overall authority* for strategy and policy for defence, e.g. the Secretary of State (SofS) for Defence.
- *Tactical*. Those that *define and set* regulatory guidelines and standards, e.g. the MAA.
- *Operational*. Those that *interpret* regulatory guidelines and standards to achieve the qualification of systems, e.g. DE&S DTs Desk Officers[7].

From a safety assurance perspective the *strategic*, *tactical*, and *operational* elements are shown in Figure 2.2. *Strategic* elements (coloured *orange*) are shown as the Defence Secretary and Defence Board. The Defence Safety Authority (DSA) Director and regulation/certification setting is conducted at a *tactical* level (coloured *light purple*). *Operational* level activities (coloured *light blue*) are conducted by those within DE&S (and those that support the DTs) and the working level staff within the MAA. The RE has a degree of *power* at the *operational* level in terms of the assurance advice to DE&S DTs. There is a level of *influence* of the *tactical* stakeholders. The level of influence/power on the RE can be taken into account when the boundary of the problem is considered.

2.2.2 Scoping the Ability to Influence

There are three forms of influence which are relevant to the research problem:

- The *ability* for the researcher to *influence* stakeholders.
- *What* influences a problem area; which may, or may not, be subject to change (e.g. constraints).
- What *needs* to be influenced to enact change or resolve a problem of interest.

To inform the research a number of system maps and rich pictures were developed, as proposed by Wilson (1990). They were to: firstly, scope the boundary of the problem and; secondly, act as a method to communicate the problem with stakeholders. A rich picture is

[6]In the context of this research an organisation can be a single business unit or a collection of business units which are all *actors* to perform a given role.

[7]This also includes those that judge evidence against *meeting* a standard (i.e. the regulator) and those that are providing evidence for a safety outcome (e.g. technical support to gather evidence against a standard).

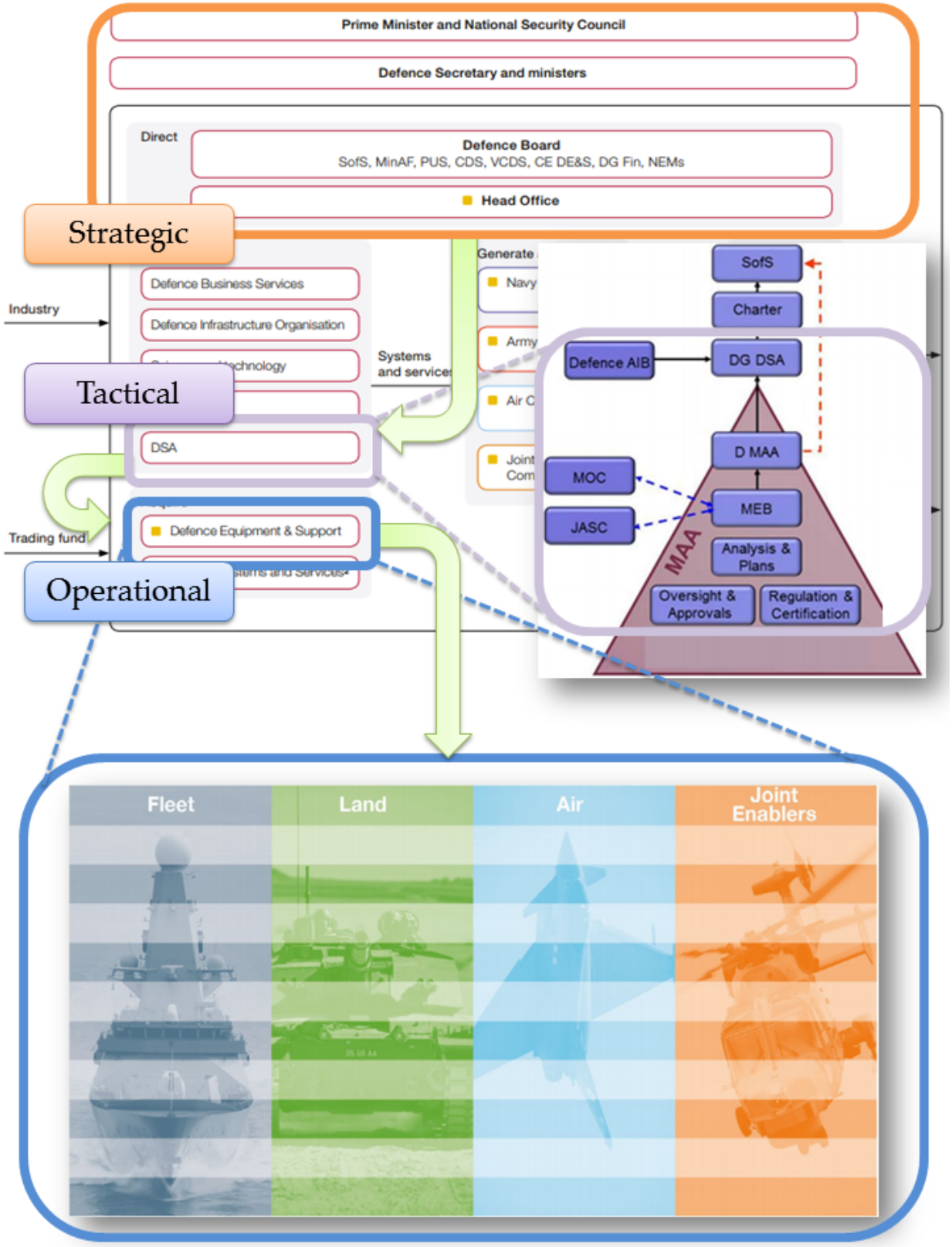


Figure 2.2: Strategic, Tactical, and Operational Units within MOD (adapted from NAO (2015), DE&S (2017), and MAA (2017*b*))

a *visual summary* of the human activity situation which is the concern at the start of the enquiry. A rich picture is *not* a system diagram (Waring, 1996). A rich picture is a very useful form of analysis but the method can be overlooked due to, what could be perceived to be, simplistic methods to create the picture. There is a discipline to creating such pictures and it was used within this research to illustrate *conflicts*, *pressures*, *stakeholders*, and to note perceived *problems*. Such a technique allowed an initial *context* boundary to be discovered with the stakeholders of interest established.

An important aspect to consider is the influence that stakeholders can have on the research. They can influence the *outcome* (the proposed transformation or outputs) and they can influence the ability for the RE to *perform* the research.

The initial stakeholders of interest for the research are shown in Figure 2.3. The stakeholders shown are those which are involved in the *management* of the EngD and those which influence the *objectives*. In reality there is a level of cross-over as the EngD supervisors (industrial and academic) inform the objectives of the EngD; however, Figure 2.3 is sufficient to show the convergence of the types of stakeholders.

A Mendelow (1981) grid proved valuable to understand the stakeholder roles. The grid states the *power* and *interest* of the stakeholders to allow the RE to identify and prioritise these stakeholders. The Mendelow (1981) grid allowed the RE to identify those stakeholders to:

- Monitor.
- Keep informed.
- Keep satisfied.
- Manage closely.

This was informed by the approach outlined by Cleland (2004) to identify for each stakeholder:

- Stake in the project.
- What the RE needs from them.
- Perceived attitude and risks.
- Risk if not engaged.

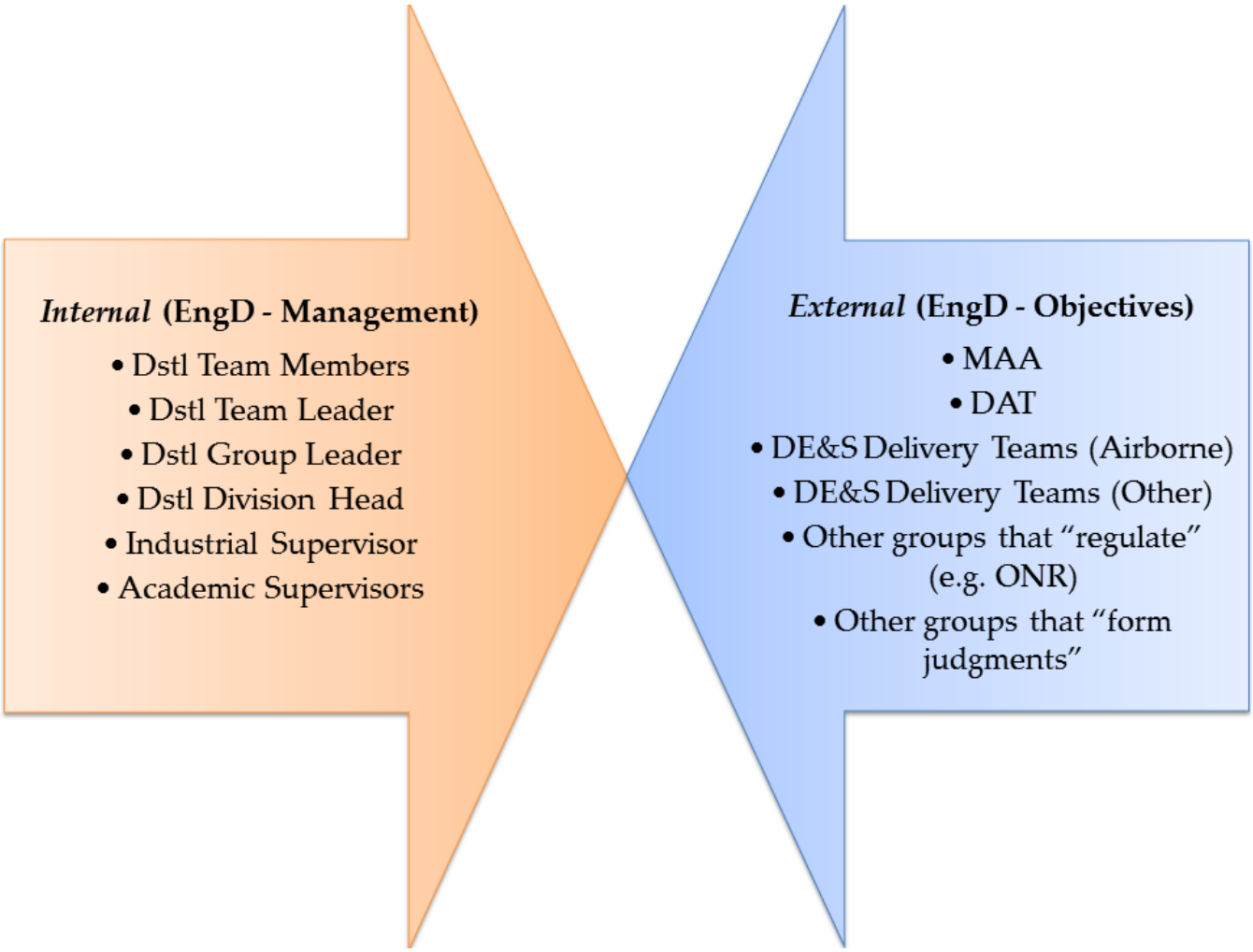


Figure 2.3: Initial Stakeholders of Interest for the Research

The Mendelow (1981) power/interest grid for the research is shown in Figure 2.4. The stakeholders are those that influence the *research* (i.e. the *objectives*) and those that influence the RE (i.e. the *management* of the EngD)[8].

The use of the techniques proposed by Mendelow (1981) and Cleland (2004) allowed the relevant stakeholders to be targeted so that their power/influence could be used to the benefit of the thesis. The activity guided which stakeholders to approach as part of the semi-structured interviews, group discussion, and workshops, for example. The grid was used throughout the research strategy to identify *who* to engage with at each of the stages.

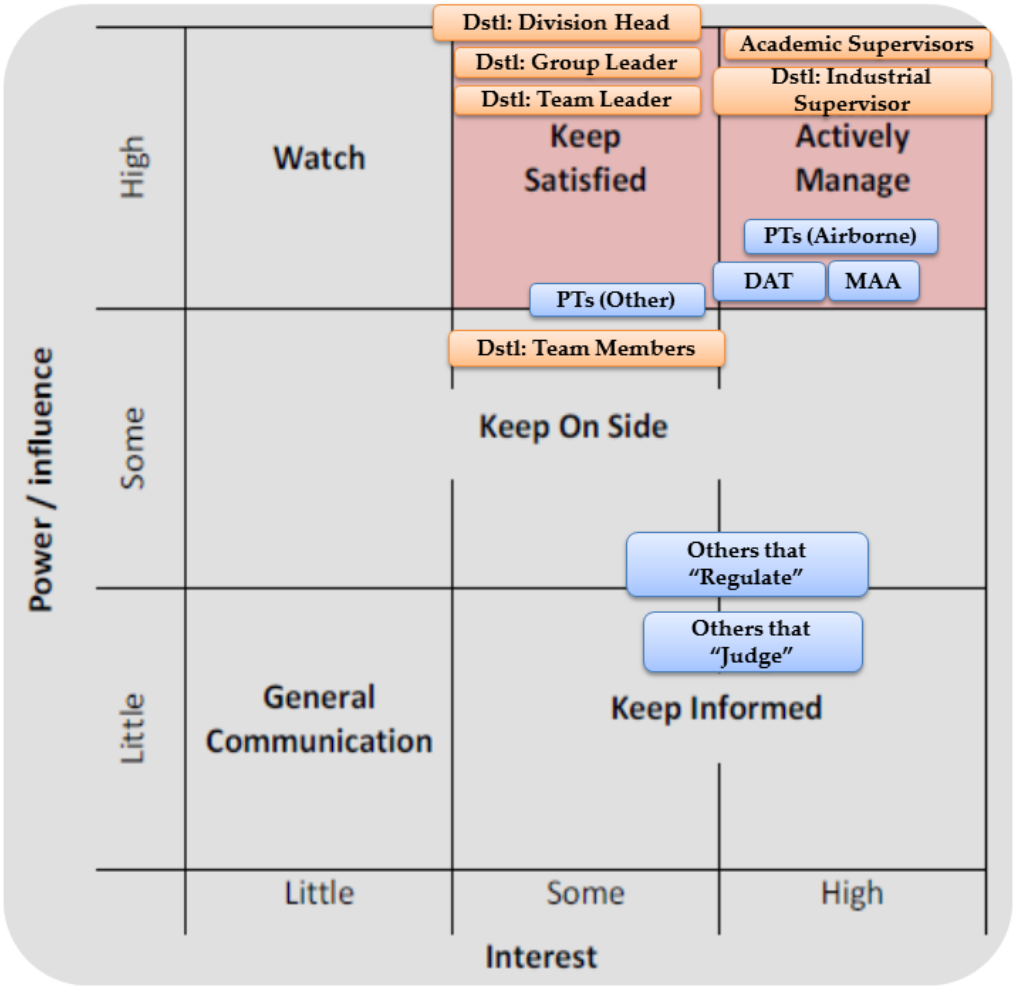


Figure 2.4: Stakeholder Analysis - Power/Interest Grid

[8]The stakeholders shown on the grid are positioned to allow the grid descriptions (e.g. 'actively manage') to be shown. Therefore, the positions of the stakeholders shows their *approximate* power/influence.

2.3 Research Questions and Initial Planning

The generation of the research questions and the initial planning is the second stage within Figure 2.1. This stage involves activities to understand the research paradigms/logic, to formulate the research questions, identification of the data sources and forms of analysis, and the intended research outputs.

2.3.1 Research Paradigms and Underpinning Logic

The *paradigm* of any research is concerned with the philosophical stance of the researcher and how data and theories are developed. A theory can be *established* via observations and experiments with the use of *inductive* logic. This type of logic is in contrast to that which is based upon the development of a theory which then uses observations and testing to *support* the theory (this is termed *deductive* logic). These two logic types are summarised within Figure 2.5.

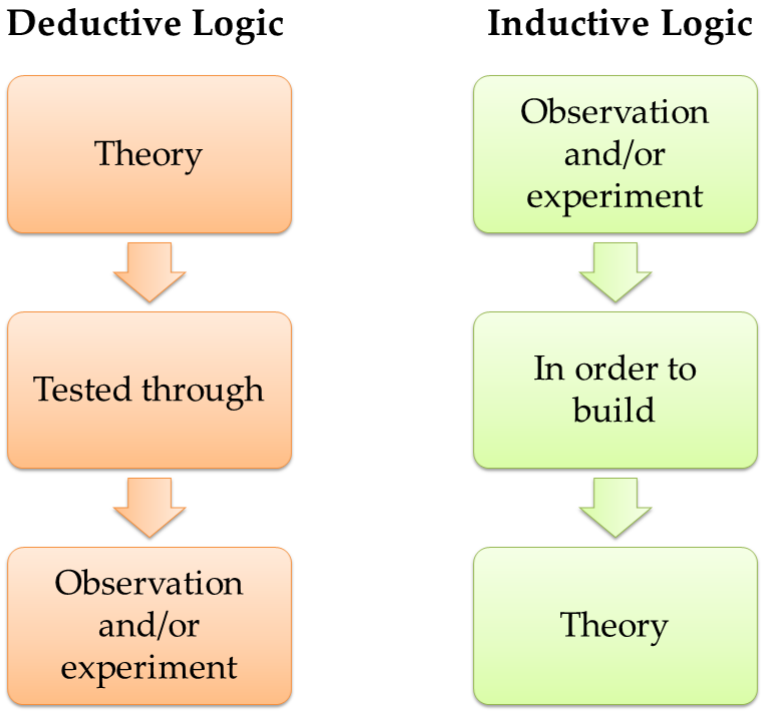


Figure 2.5: Deductive vs Inductive Logic (adapted from Gill and Johnson (2014))

The implementation of *deductive* logic is based upon a paradigm called *positivism* which is very much based upon a so-called 'natural science' perspective. The alternative to *positivism* is a paradigm called *phenomenology* which was developed to study social phenomena with the natural scientists paradigm (based upon *positivism*) being insufficient to capture

uncertain causes (Collis and Hussey, 2009). The two paradigms are based upon a number of underlying assumptions regarding such aspects as ontology[9], epistemology[10], and axiology[11]. The assumptions span the level of independence of the researcher to the level of bias of the researcher. In reality, the paradigms are at *extremes* of a scale with other paradigms sitting within the two. This continuum means that the features and concepts of one paradigm are relaxed and replaced with those of the next (Collis and Hussey, 2009). The research paradigm continuum is illustrated within Figure 2.6.

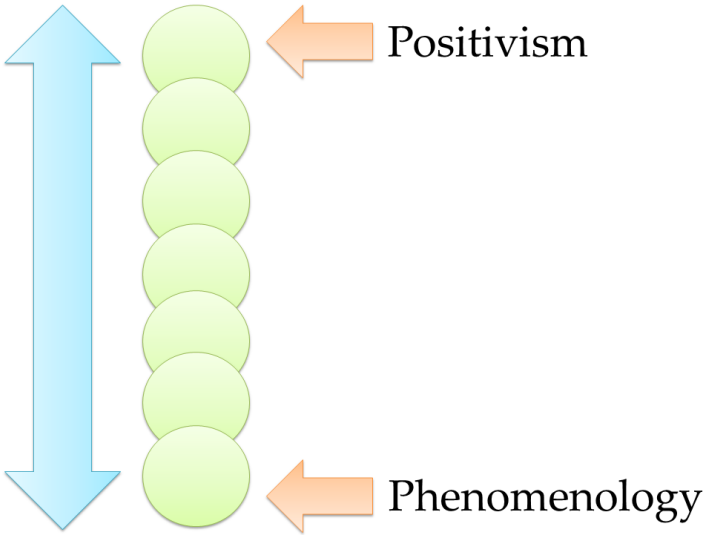


Figure 2.6: Research Paradigm Continuum

Research can have a mix of paradigms as the logic and theory development *can* change during the course of conducting the research, i.e. for each sub-question.

2.3.2 Research Questions

There will be a mixture of paradigms and approaches used to respond to the research questions. The concept of understanding *enhancements* to the defence safety assurance process was originally not based upon a known theory by the RE. There was an element of inductive logic to develop a theory with regard to the use of *diverse* evidence to enhance the assurance process. The use of *qualitative* methods were adopted to derive the theory (e.g. workshops), in keeping with the *phenomenology* paradigm. A theory based upon the concept of diverse evidence to enhance software safety assurance led to a paradigm which was more in keeping with *positivism*, in that the theory was tested via the observations.

[9]The nature of reality.
[10]What constitutes valid knowledge.
[11]The role of values.

The appreciation of the research paradigms and underpinning logic resulted in a research question *grand tour* (GT) question with a set of sub-questions (see Table 2.1). For each of the research sub-questions the associated paradigms are also shown. Table 2.1 indicates *primary* and *secondary* paradigms to indicate the key approach (and data sources) which informed each research sub-question. Any research sub-question was mainly responded to via a primary paradigm with supplementary research utilising the secondary paradigm.

The research paradigm influences the data sources which are used to respond to each sub-question. This is due to the differing research assumptions that were made for each sub-question. The choice of data collection methods for this research has been influenced by Collis and Hussey (2009). Information on the data collection methods and sources are shown in Table 2.2.

ID	**Research Grand-Tour Question**		
GT	What enhancements can be made to the current UK defence domain's software safety assurance approaches for capturing and judging supporting evidence?		
ID	**Research Question**	**Pos.**[1]	**Phen.**[2]
RQ1	What is the current approach to system safety assurance within the UK defence domain and are there alternative system-level approaches?	{•}	•
RQ2	What is the current permissible software safety assurance evidence within the UK defence domain and related domains?	{•}	•
RQ3	What software safety assurance evidence is relevant/admissible and what are the underpinning principles for the use of such evidence?		•
RQ4	What are the unintended consequences of adopting incorrect metrics when forming decisions and how can system/evidence relationships be communicated to stakeholders?	•	
RQ5	What is a suitable structure for software safety assurance evidence and can mathematically derived approaches inform how judgements are made on the evidence and for proposing alternative/optimised solutions?	•	
RQ6	What observations and recommendations can be made on how to implement a software safety assurance evidence argument and how to inform a UK defence software safety assurance strategy?		•

Note(s): 1. Pos=Positivistic; 2. Phen=Phenomenological.
Key: •=Primary paradigm; {•}=Secondary paradigm.

Table 2.1: Research Questions (RQs) with Associated Paradigms

Research sub-question 1 requires an initial phenomenological approach in order to build a theory from relevant observations regarding current system safety assurance approaches within the UK defence domain. Document review and analysis[12] will inform the sub-question with the use of semi-structured interviews to corroborate a level of the findings. The observations will assist in also identifying potential alternative system-level approaches[13]. At this stage a more positivistic paradigm may be suitable to test any theories developed from the initial assurance review. It is envisaged that the approach would be informed via document review/analysis but also supported by the use of models/frameworks to implement a suitable case study. This deductive process will allow any system-level approaches to be tested via a form of experimentation.

Research sub-question 2 follows an initial similar paradigm to sub-question 1 with a phenomenological approach being required to make use of observations from the review/analysis of suitable documentation. Documentation should provide initial sets of information on the current permissible evidence but a level of stakeholder interaction may be required[14]. Further document review/analysis and stakeholder interaction may be needed to understand the concepts of evidence and how/where is it adopted within the safety domain. Articulating the permissible evidence will require levels of corroboration with stakeholders to understand the *realities* of evidence rather than a guidelines/standard perspective. The use of workshops would be of potential benefit. There is also scope to develop relevant theories on how to tackle some of the challenges based upon the initial observations to how evidence is adopted within other safety-critical domains.

Research sub-question 3, again, has a phenomenological stance. There will be a level of document review/analysis to determine the potential permissible evidence as well as the use of stakeholder engagement[15]. Document review/analysis will also be adopted to derive the underpinning principles for the use of evidence to build a theory. Again, stakeholder engagement may possibly be necessary.

With research sub-question 4 the approach taken is more towards the positivistic paradigm as by this stage of the research it is envisaged that there will be a number of theories which can be tested via the use of observation/experimentation. The importance of metrics and the consequences of adopting them incorrectly can be tested via a suitable model. Developing models to illustrate and enhance effective communication of evidence to stakeholders can be supported by suitable workshops and semi-structured interviews. The stakeholder engagement will allow feedback to be gained and to test the developed theories.

[12]In order to identify the focus of any intervention and to define any areas of enhancement to the the current safety assurance evidential approaches.

[13]In essence, via reviewing the context of systems safety assurance.

[14]Potentiality via semi-structured interviews.

[15]Potentiality via semi-structured interviews.

Research sub-question 5 strongly continues the positivistic paradigm with the potential development of a framework to define a suitable structure for software safety assurance evidence. The sub-question is, in essence, learning from the theory developed from the previous sub-questions to test them via a framework using suitable case studies. Stakeholder engagement is required, as with the phenomenological sub-questions, with a potential broader range of methods for the interaction; e.g. workshops, semi-structured interviews. Exploratory testing may provide further insight into the software safety assurance behaviour. The sub-question will play a significant part in testing the thesis main theories via experimentation.

Finally, research sub-question 6 will then move back towards a more phenomenological stance with an inductive process to gain an understanding of the previous observations/experimentation to provide sufficient recommendations. The sub-question will draw upon the document review/analysis of previous sub-question findings and the subsequent deductive process to test the ongoing theories. In essence, the sub-question derives the findings from outputs from each data source to amalgamate and distil the recommendations.

Table 2.2 provides a summary of the data sources and analysis methods aligned to the research sub-questions. The exact sources/methods are not fixed. However, the sources/methods are influenced by the chosen research paradigm for the sub-questions as these inform the research approach. Table 2.2 indicates *primary*, *secondary*, and *informative* elements to indicate the key data sources and methods which underpinned each research sub-question. Each research sub-question was responded to by adopting the primary data sources with supportive research utilising the secondary data sources. The exception is RQ6 which utilised each of the approaches (via previous RQs) to arrive at the recommendations. Further information on each of the data sources and analysis methods is contained within sub-section 2.3.3.

Data Source / Analysis Method	**RQ1**	**RQ2**	**RQ3**	**RQ4**	**RQ5**	**RQ6**
Documentation review and analysis	•	•	•	{•}	{•}	[•]
Workshop			•	•	•	[•]
Semi-structured interviews	•	•	•	{•}	{•}	[•]
Group discussion		•	•		{•}	[•]
Case studies	{•}	{•}		•	•	[•]
Exploratory testing				•	•	[•]
Models	{•}	{•}		•	•	[•]

Key: •=Primary data source; {•}=Secondary data source; [•]=Informative data source.

Table 2.2: Research Sub-Questions with Associated Data Sources and Analysis Methods

2.3.3 Data Sources and Analysis Methods

Research is underpinned by data. The role of the data changes depending on the logical perspective. The data can support the theories being tested, from a *deductive* perspective, or it can allow the theories to be developed from the observations, from an *inductive* perspective. The following data sources and analysis methods were adopted for the research.

- *Documentation review and analysis.* The review of existing literature is paramount to *develop* the initial theories and to ascertain the need for intervention. It also *informs* the observations which are made to support the theories. A systematic and comprehensive literature review was undertaken to gain suitable insights into the problem domain. The review followed the guidance contained within Collis and Hussey (2009) with a wide range of sources adopted with the use of key word searches conducted from reputable journal and conference libraries, e.g. Institute of Electrical and Electronics Engineers (IEEE) Xplore[16]. Literature was reviewed for *relevance* to the problem area and the *value* to understanding the solution space.

- *Workshop.* The benefit of a workshop is that it allows for a level of interaction and discussion which is planned into the structure of the event. The RE designed and delivered a workshop to a number of MAA representatives (software assurance regulators) to gain feedback on the initial research theories and elements of the framework design[17]. The RE also jointly-presented the *Wheel of Qualification* model and underpinning concepts at a workshop attended by defence organisations to articulate the evidence and relationships of interest for a specific project.

- *Semi-structured interviews.* This form of qualitative research is typically underutilised but has great value (Galletta, 2013). Semi-structured interviews were conducted to understand the approaches adopted by a range of safety-related domains. These interviews allowed an understanding to be gained of the *realities* of software assurance from the perspectives of experts in their fields. Purposefully, open-ended questions were designed to allow a greater level of discussion and to gather a rich set of data[18]. Good practice was followed in relation to the types and forms of interview questions/style, e.g. requiring elaboration on initial statements (Collis and Hussey, 2009). One-to-one interviews (face-to-face and via telephone) were deliberately chosen to allow more open discussions. This qualitative data was used to inform further qualitative assessment as part of the documentation review and analysis.

[16]See the following for further information: https://ieeexplore.ieee.org/Xplore/home.jsp.
[17]See Appendix A for further information.
[18]See Appendix B for further information.

- *Group discussion.* Group discussions allow for effective outputs and dialogue (Young et al., 2006) which assisted the RE to develop the theory and to scope design features of the framework. The dialogue was with SMEs in the area of software assurance and so allowed a level of validation to be conducted on the research findings. As with the semi-structured interviews, this data was used to inform further qualitative assessment.

- *Case studies.* The research adopted *experimental* case studies[19] to examine the difficulties in implementing the proposed concepts and techniques using suitable supporting data (Scapens, 1990). The case studies allowed the refinement of the DSF and of the diverse evidence concepts. This supported an iterative approach to adapt and learn from the case study experiments. A case study was also devised to understand SoS safety interface issues to support the SLF.

- *Exploratory testing.* In addition to conducting case studies, exploratory testing was implemented to understand the diverse evidence concepts and the underpinning relationships. The process of exploratory testing is a recognised technique within the software testing domain. The process generates tests of interest whilst being cognisant of the action taken and the subsequent impact (Kaner, Falk and Nguyen, 1999). The exploratory testing allowed further refinement of the DSF and the findings.

- *Models.* Models are representations that can aid in defining, analysing, and communicating a set of concepts (SEBoK, 2018*b*). Conceptual modelling was used to understand the relationships between a number of entities within a variety of contexts, e.g. the *Wheel of Qualification.* The models allowed relationships to be stated between systems and their supporting evidence. They also acted as a method to communicate the qualification status of a system/platform.

2.3.4 Selection of Interviewees

Based upon the data sources and analysis methods it is anticipated that interviewees/SMEs and group participants will inform a range of research activities. The criteria for the selection of these participants will be based, broadly, on the following (Johnson and Weller, 2002):

- *Who has the relevant information?* With the RE working in the software assurance domain numerous participants were able to be identified and contacted. Participants were chosen to provide a cross section of viewpoints from those that set the regulatory policies to those that are tasked with implementing the standards. It was important to understand the *intent* and the *reality* of any standards for example.

[19] *Experimental* studies have interventions introduced with the subsequent effects observed.

- *Who is accessible?* Efforts were made to conduct suitable interviews/workshops which were based on the availability of the *right* personnel. The key is the suitability of the person with the methods to gain information, including the location and time, based upon the value of the information.

- *Who is willing to give relevant information?* A participant *having* the information will not necessarily mean that it will be *revealed*. The choice of interview settings, e.g. at participants place of work, and the question sets assisted with allowing the participants to feel confident in providing relevant information. The fact that participants were informing research which will have academic and industrial impact assisted due to participants feeling that they were actively influencing research.

- *Who is most able to give the information?* A relevant participant may wish to provide information but may have issues communicating the information. The interview question sets and workshop agendas were designed to allow participants to provide information in a style which suited them. Open questions were devised with scenarios and point of discussion included in workshops. These all assisted in creating free discussion and input.

2.3.5 Research Outputs

Research outputs have value in that they can illustrate learning, communicate ideas, gain feedback, and show research progress. The outputs from the research are intended to be proportional and based upon the restrictions that are placed upon the RE and the research itself, i.e. the EngD *management* and *objectives*.

- *White papers*. Creation of white papers published internally within MOD to provide confidence to those that may adopt diverse evidence as part of a software safety assurance strategy. Papers were generated to provide guidance on the procurement approach to ensure that diverse evidence can be appropriately contracted for.

- *Formal reports*. A number of formal reports were delivered to the REs customers to implement some of the findings and enhancements identified as part of the research. The reports were delivered to those that have a degree of *power* (in terms of enacting a level of change) to incorporate the research findings and to apply *influence* themselves.

- *Conferences and seminars*. Presentations on the wider scope of the diverse evidence concept were conducted, e.g. from the theory to specific cases of adopting alternative

evidence. Conference and seminar presentations were provided to internal MOD audiences as well as some of the findings from the research being presented at international conferences[20].

- *Journal articles.* Articles were submitted to a reputable journal to allow elements of the research findings to be shared with a wider audience. It is anticipated that further journal articles will be published based upon the wider research findings.

- *IDC reviews/events.* The IDC in Systems at the University of Bristol have frequent reviews and events. These allow research plans and progress to be discussed with EngD cohorts, other industrial supervisors, and other academic supervisors. The reviews and events assisted in allowing the theory to be discussed and for the plans to be subjected to scrutiny.

2.4 Research Execution

The previous sections within this chapter detailed the research strategy with information on the paradigms and logic which underpins the approach. In essence, the strategy sets the foundation for the research to ensure that the planned structure aligns with defensible and quality[21] research outcomes.

The *execution* of the research is putting the strategy into action. This involves responding to the research questions via the defined paradigms and approaches to complete a systematic process to meet the research objective. The research *execution* comprises of stages 3-8 of the strategy outlined in Figure 2.1 (*scoping the requirement for intervention* through to the *conclusions*).

The research process itself was iterative, as outlined in Figure 2.1. The steps taken for the *execution* of the research is detailed in the thesis structure (section 1.4). The chapters which follow provide the detail on the *execution* of the research via the activities stated in this research strategy. The mixed paradigm and supporting data sources, and hence the forms of analysis, provide a structured approach to deliver the research.

[20]For example, IET System Safety and Cyber Security Conference.

[21]From a research perspective the *quality* is measured by the *validity*, *reliability*, and *generalisability* of the research (as stated, for example, by Leung (2015)).

Chapter 3

Background and the Problem of Interest

There is a need to place the problem of interest within it's context and why it is an important issue to tackle for the defence domain. Is there is a sustained and reasoned argument for the use of diverse evidence? Are there any software assurance challenges that could be alleviated? To identify potential enhancements to the software safety assurance methods there is benefit in understanding software and how it fits in the wider *system* assurance process. In doing so, opportunities for enhancements to the *system* assurance process can also be identified.

This chapter will provide information on:

- *Problem of Interest.* Why there is a need for software to be assured, the initial scope of the problem of interest, and information on the elements of a safety case developed as part of an assurance activity.
- *Context of Systems Safety Assurance.* Introduction to the safety management terminology and information on how DS 00-56 (UK MOD, 2014*c*) is applied. How the safety management processes are applied and how safety cases are adopted within MOD. What constitutes a PE and how PEs are assessed.
- *Current Evidential Approaches: Areas for Enhancement.* Why there is an existing and continuing need for the adoption of diverse evidence to support software safety assurance activities. At a systems level there are a number of enhancements which can be made to current practices. A SLF is described which provides a mechanism to deal with limited information as part of a SoS safety argument.

3.1 Problem of Interest

The problem of interest is focussed on the military airborne domain, although the solutions may have scope to be applied more widely, e.g. MOD land and navy domains and the civil sector. The RE is actively engaged with the software safety assurance of military airborne platforms and this shapes the RE's research motivation and the industry requirement.

Software is one element within the wider *equipment's* assurance requirements, as illustrated in Figure 3.1. With *equipment* itself being an element within the wider *environment* which includes the *jobs/roles* performed and the *people* themselves. Systems cannot be considered in isolation from the operating environment. Indeed, any safety assurance review needs to understand how the system interacts with other systems and the wider environment. These all have an effect on the safety of the system.

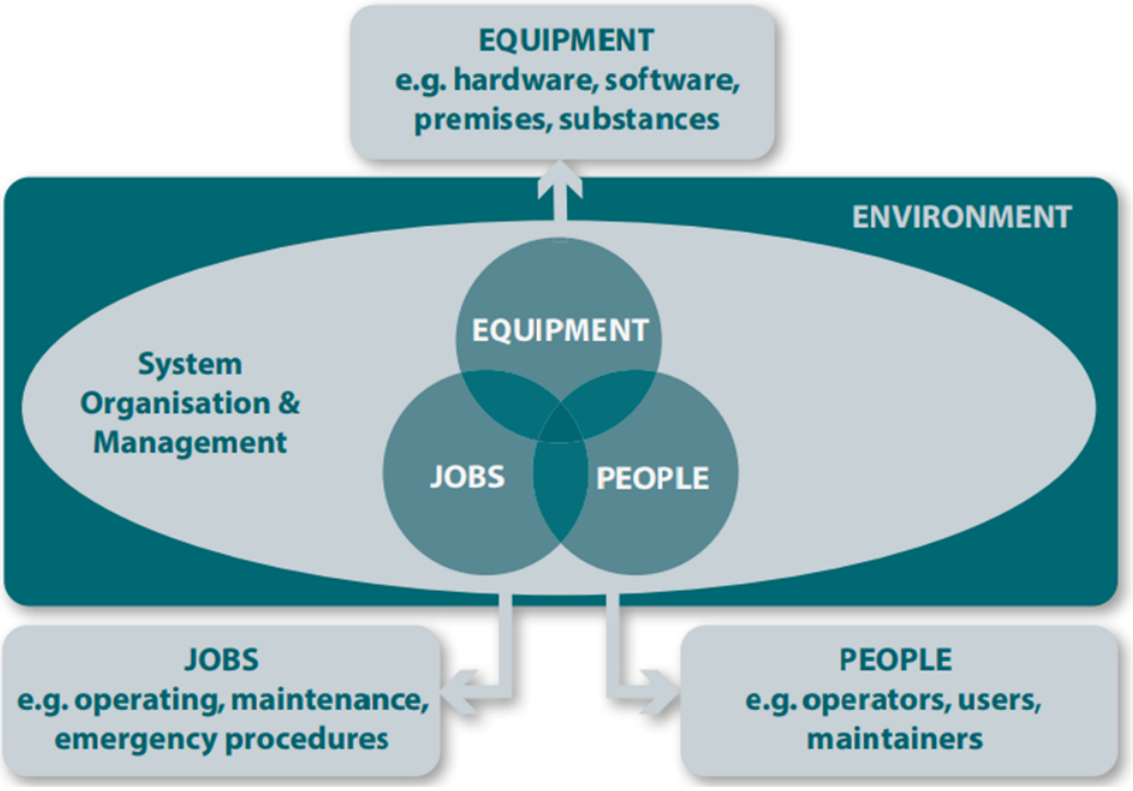


Figure 3.1: Considerations within a Safety Environment (UK MOD, 2018)

Within military systems the safety assurance is assessed at differing levels of abstraction; e.g. DS 00-56 (UK MOD, 2014*c*) is concerned with the "Safety Management Requirements *for* Defence Systems" with DS 00-55 (UK MOD, 2014*b*) focused on the "Requirements for Safety of PEs *in* Defence Systems". In broad terms, DS 00-56 is focussed on the wider

safety management issues at a *system* level with DS 00-55 focussed on the assurance at a *software/CEH* level.

To gain a *full* safety assurance qualification[1] the evidence needs to be judged against the objectives within these standards. Software is one part of the safety assurance environment but it is a *critical* element as software failures can result in *hazardous*[2] or *catastrophic*[3] consequences.

3.1.1 The Importance of Software Safety Assurance

Anecdotally, in 1945 engineers found a moth in "Panel F, Relay #70" of the Harvard Mark II system. The system was running tests when engineers noticed issues. The moth was subsequently trapped, removed, and taped into an engineer's logbook with the words: "first actual case of a bug being found" (Garfinkel, 2005)[4].

The impact of software failures and 'bugs' span a number of domains (e.g. medical, aerospace, gas pipelines etc) and software failures have occurred for as long as systems have been reliant on such software. Historic examples of software failures include, but are certainly not limited to (Garfinkel, 2005):

- Mariner I Space Probe (1962). Intended flight path diversion due to software fault in the flight software. The probe was destroyed.
- Therac-25 Medical Accelerator (1985-87). Malfunction of radiation therapy device due to a software *race condition*[5]. Loss of life occurred.
- Ariane 5 Flight 501 (1996). A bug in an arithmetic routine led to an *overflow condition*[6]. The expendable rocket was destroyed.

Within the airborne domain there are also numerous examples of software faults leading to *hazardous* or *catastrophic* events. These include, but are certainly not limited to:

[1]The term *qualification* being the process of granting the approval for an aircraft or system to be operated in flight.

[2]In an airborne context a *hazardous* event results in "serious or fatal injury to a small number of occupants other than the flight crew" (RTCA, 2011*a*).

[3]In an airborne context a *catastrophic* event has "multiple fatalities (usually with loss of the aircraft)" (RTCA, 2011*a*).

[4]There is some debate to the origin of the term 'bug', e.g. Magoun and Isreal (2013), but for the purposes of this chapter the Harvard Mark II system example is as good as any.

[5]A *race condition* is an undesirable situation that occurs when a device or system attempts to perform two or more operations at the same time. However, due to the nature of the device or system, the operations must be conducted in a specified sequence to be completed correctly (TechTarget, 2015).

[6]An example of an *overflow* is that of an integer overflow which is the result of attempting to place into memory an integer (whole number) that is too large for the integer data type (TechTarget, 2006).

- Airbus A330-303 (2008). While the aircraft was in cruise at 37,000ft one of the aircraft's three Air Data Inertial Reference Units (ADIRUs) started to intermittently output incorrect values. In response, the aircraft's Flight Control Primary Computers (FCPCs) commanded the aircraft to pitch down. The incorrect values were partially traced to the ADIRUs Central Processing Unit (CPU) due to an unknown software design limitation in the FCPCs. 12 occupants of the aircraft were seriously injured (ATSB, 2011).

- Airbus A400M (2015). An issue with a data parameter file resulted in 3 of the 4 aircraft's engines initiating a power-off shortly after take-off. The error was not in the code itself but in the configuration settings within the Engine Controller Units (ECUs) of the engines. 4 aircrew died (Gibbs, 2015, Gallagher, 2015).

Sufficient assurance is needed in the underpinning software so that there is confidence that severe events have a low likelihood of occurrence. However, the amount of software that requires such assurance is growing. There are various estimates stating the rate of growth of software within avionic platforms; e.g. approximately 400% every 2 years (Carlson, 2016), with safety-related code doubling in size every 4 years (Zolotas et al., 2017). Figure 3.2 shows the estimated civil on-board avionics SLOC growth with the predicted software base cost (based upon Constructive Cost Model (COCOMO) II).

From a military avionics perspective there is a similar trend of growth. Figure 3.3 shows the growth of software in military aircraft, stated as KSLOC within the specific aircraft over time.

This increase not only has cost implications but also increases the complexity of the software development and its subsequent management. There are efforts to reduce the cost of certifying airborne software via initiatives such as Future Airborne Capability Environment (FACE)[7] and Software Engineering Costs and Timescales – Aerospace Initiative for Reduction (SECT-AIR)[8]. Confidence in the underpinning software is a fundamental element within any overall platform safety argument. If there is not a sufficient level of confidence in the software then there is not a sufficient level of confidence in the system (and therefore the overall platform).

[7]See the following for further information: http://www.opengroup.org/face.

[8]See the following for further information: https://gtr.ukri.org/projects?ref=113099. The RE is actively engaged with this initiative, e.g. via Ashmore and Standish (2017).

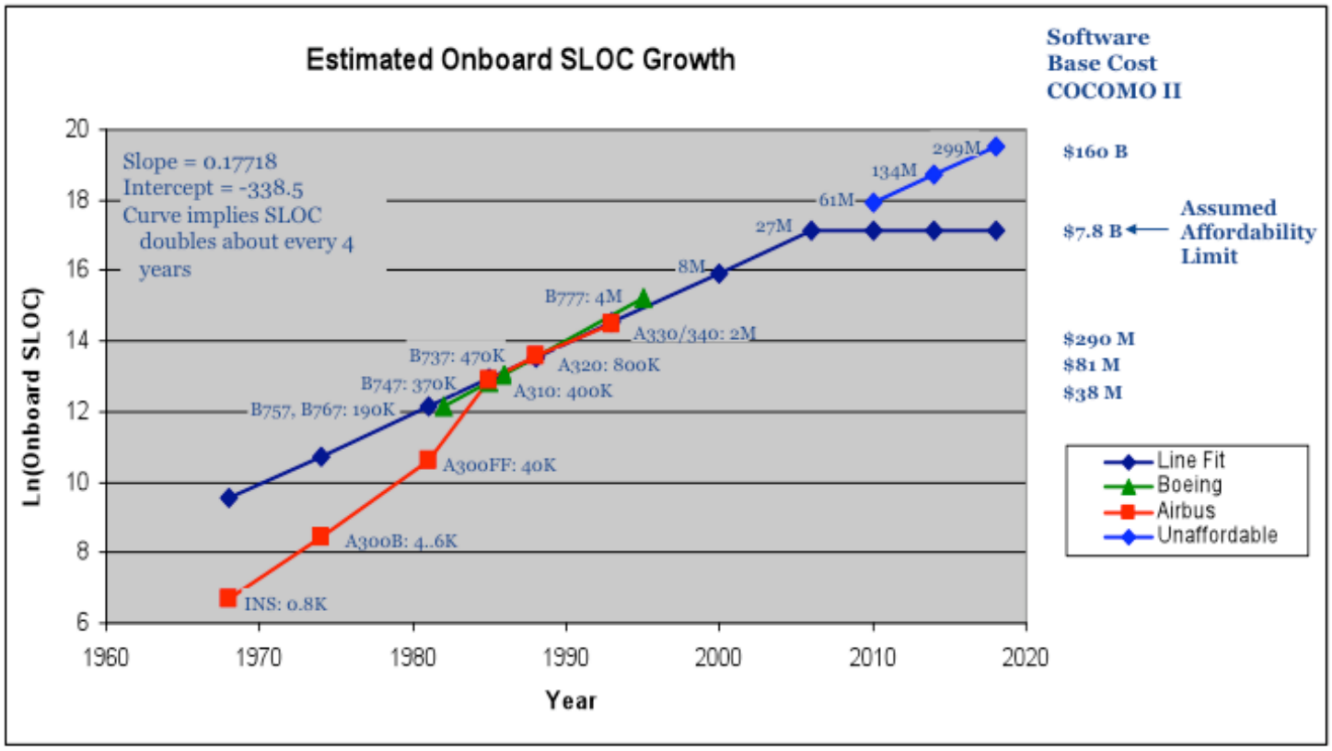


Figure 3.2: Estimated Onboard SLOC Growth on Commercial Aircraft (Redman et al., 2010)

3.1.2 Research Focus on Software Safety Assurance

The focus of the research is on *software* safety assurance and gaining suitable confidence in the evidence. How evidence can be used from other elements of the DSs will be explored; specifically: CEH, the safety assessment process, and airworthiness related security. By concentrating on software assurance means that the principles of evidence confidence can be explored. Any lessons learnt can then be applied to wider evidence strands, e.g. CEH, if the principles developed are appropriate and have merit.

The concepts developed as part of this thesis need to allow software confidence to remain a consideration within the wider system or platform safety case, see sub-section 3.1.3 for further information.

3.1.3 Elements of a Safety Case

From a MOD perspective, DS 00-56 (UK MOD, 2014*c*) defines a Safety Case as a: "structured argument, supported by a body of evidence that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given environment". It is

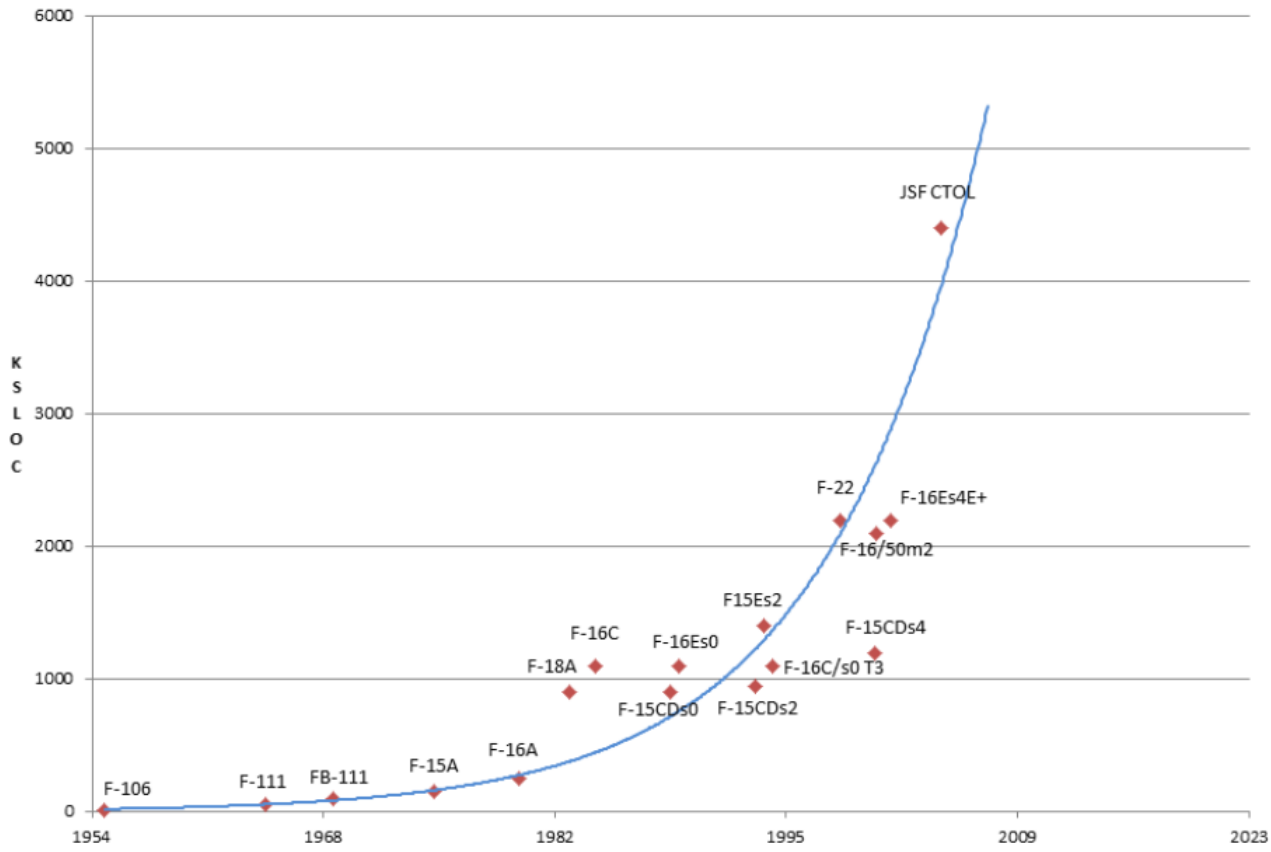


Figure 3.3: Growth of Software in Military Aircraft (KSLOC) (AVSI, 2011)

this body of evidence which is used to gain confidence in a system or platform[9].

Bishop and Bloomfield (1998) provide a view on what a Safety Case needs to do via goals. These goals can assist with ensuring that a Safety Case adheres to the DS 00-56 definition:

- Make an explicit set of claims about the system.
- Produce the supporting evidence.
- Provide a set of safety arguments that link[10] the claims to the evidence.
- Make clear the assumptions and judgements underlying the arguments.
- Allow different viewpoints and levels of detail.

To meet the goals, Bishop and Bloomfield (1998) further define the main elements to be included within a Safety Case:

[9]Further details on MOD Safety Cases are stated within AESMS (2017).

[10]A more suitable term might be that safety arguments form a direct implicit relationship and justification as the term *link* implies a weaker form of relationship.

- *Claim* about a property of the system or some subsystem.
- *Evidence* which is used as the basis of the safety argument.
- *Argument* linking the evidence to the claim; which can be deterministic, probabilistic, or qualitative.
- *Inference* is the mechanism that provides the transformational rules for the argument.

These main elements are shown in Figure 3.4. There are a number of approaches to structure such elements; e.g. the Claims, Arguments and Evidence (CAE) notation developed by Adelard[11].

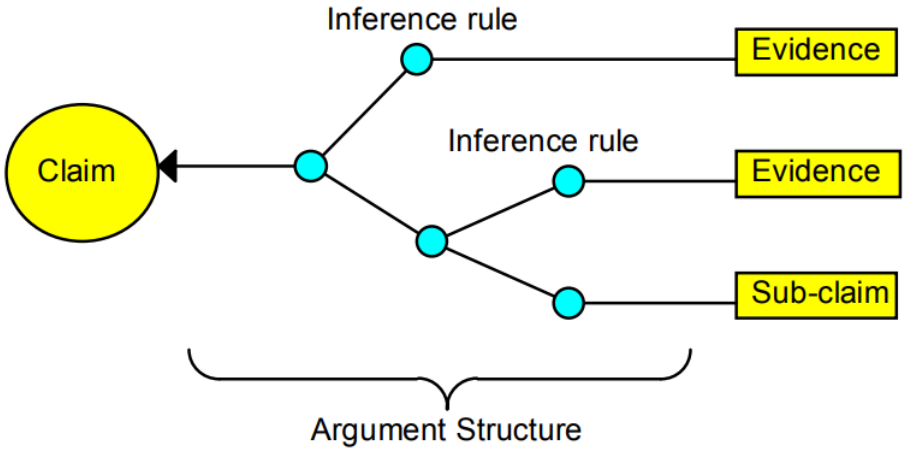


Figure 3.4: Main Elements of a Safety Case (Bishop and Bloomfield, 1998)

It is envisaged that the enhancements to the current methods for software safety assurance will adhere to the principles of these main elements. Any tool created by the research will allow judgements on the software confidence to be captured. This judgement will act as *evidence* to feed into the overall safety *arguments* and the subsequent *claims*. This will allow the arguments/claims to maintain a level of consistency with any wider assurance case being formulated. The structure of the *sub-claim* acting as the *evidence* to support an overall safety case will have a similar structure to that in Figure 3.4. This will ensure that the *sub-claim* for the software is itself based upon *evidence* with rules to form the *argument*.

How a sub-claim for the judgement on the software confidence could feed into an overall safety claim is shown in Figure 3.5. This concept is supported further by the current claims, argument, and evidence approaches which are used to consider existing process-based guidelines, e.g. DO-178C (RTCA, 2011*a*)[12].

[11]CAE notations can be structured using such tools as Assurance and Safety Case Environment (ASCE). See the following for further information: https://www.adelard.com/asce/choosing-asce/index/.

[12]Further information on the MOD assurance process for PEs is contained within sub-section 3.2.3.

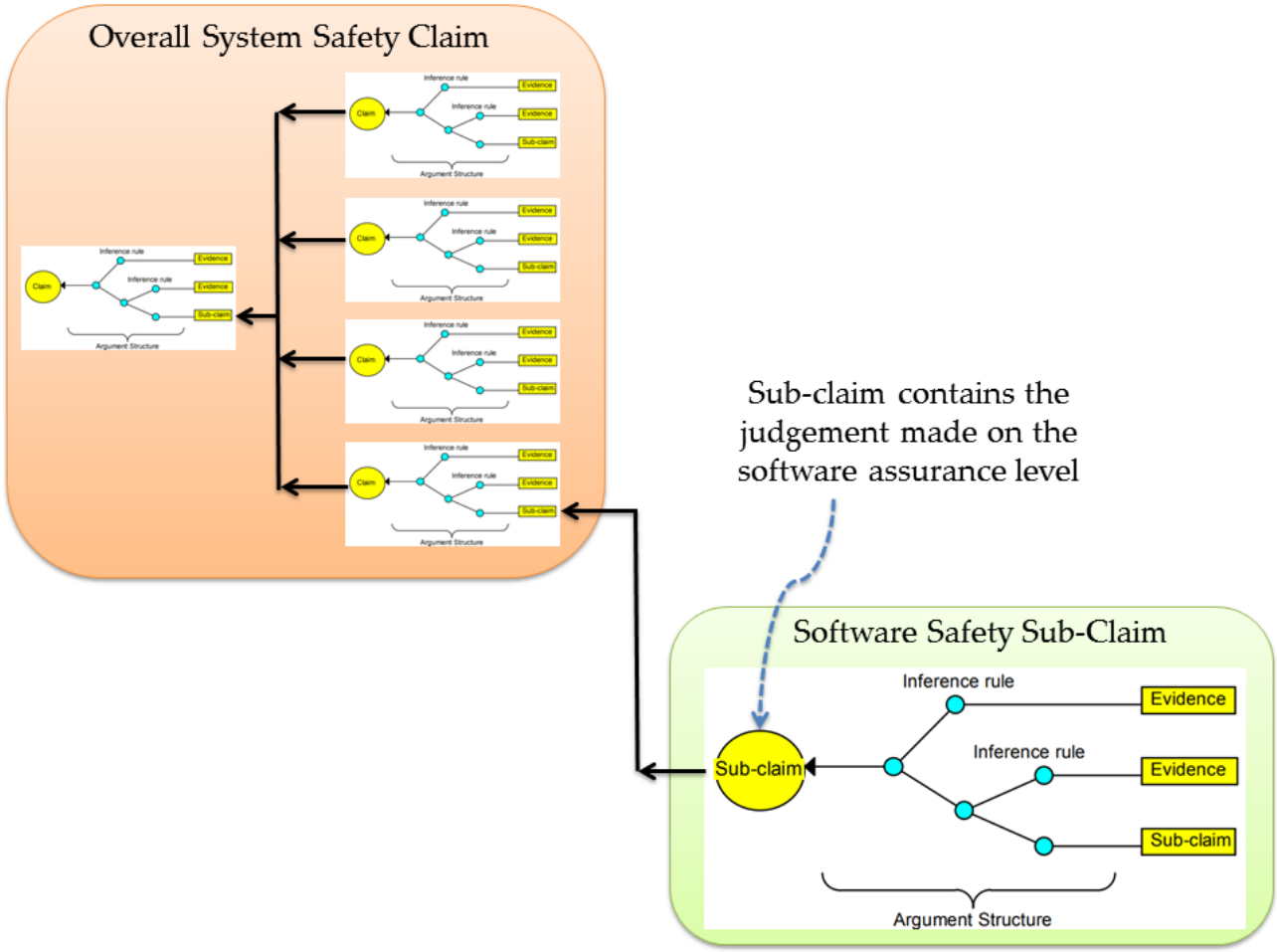


Figure 3.5: Link Between the Overall Safety Claim and Software Sub-Claim (adapted from Bishop and Bloomfield (1998))

There are different layers of abstraction for hazard identification and management. DO-178C, for example, states that safety analysis is conducted at the *systems* level. However, it has been argued that specific hazards could be identified at a software *development* level, such as hazard analysis at the source code stage (Hawkins et al., 2011). As an example, the software design hazard identification stage could assert that: hazardous design errors have not been introduced and that hazardous behaviour has been assessed and mitigated. A more traditional safety assurance process would assert that for the design: safety requirements are appropriate for the design and that safety requirements are satisfied.

The abstraction level for the hazard analysis influences elements of a safety case. Specifically, the evidence which is gained, the arguments which are formed, and therefore the claim which can be made. In keeping with DO-178C, and the approach which is adopted widely within industry and government, the assumption is that the software confidence judgement from this research will feed into a system-level safety analysis.

3.2 Context of Systems Safety Assurance

The MOD has a defined safety assurance approach at the *systems* level for safety-critical systems. This is distinct from the lower-level airworthiness assurance activities conducted for the *software*. At a broad level this is illustrated within Figure 3.6 which shows the aircraft, system, and item supporting processes to achieve the aircraft level certification. The safety assessment processes are fed from the system process activities. Each of the system development processes have supporting item development processes, hardware and software, which are subject to assurance activities.

3.2.1 MOD Approach to Safety Management

DS 00-56 (UK MOD, 2014*c*) is the MOD standard to state the *safety management requirements for defence systems*. It provides the requirements and guidance for the achievement, assurance, and management of safety. DS 00-56 was initially published in 1991 and has evolved over time to take into account numerous MOD strategic directions and approaches, e.g. the policy for MOD to be as *civil as possible, and only as military as necessary* (McDermid and Williams, 2014).

DS 00-56 enables the acquisition of Products, Services and/or Systems (PSS) which are compliant to relevant safety legislations, regulations, and policies. DS 00-56 is applied by defence contractors when it is stipulated by the MOD. DS 00-56 can be applied to a broad spectrum of defence procurements and this is apparent in the definition of PSS (UK MOD, 2014*c*, McDermid and Williams, 2014). The relationships within the PSS concept is

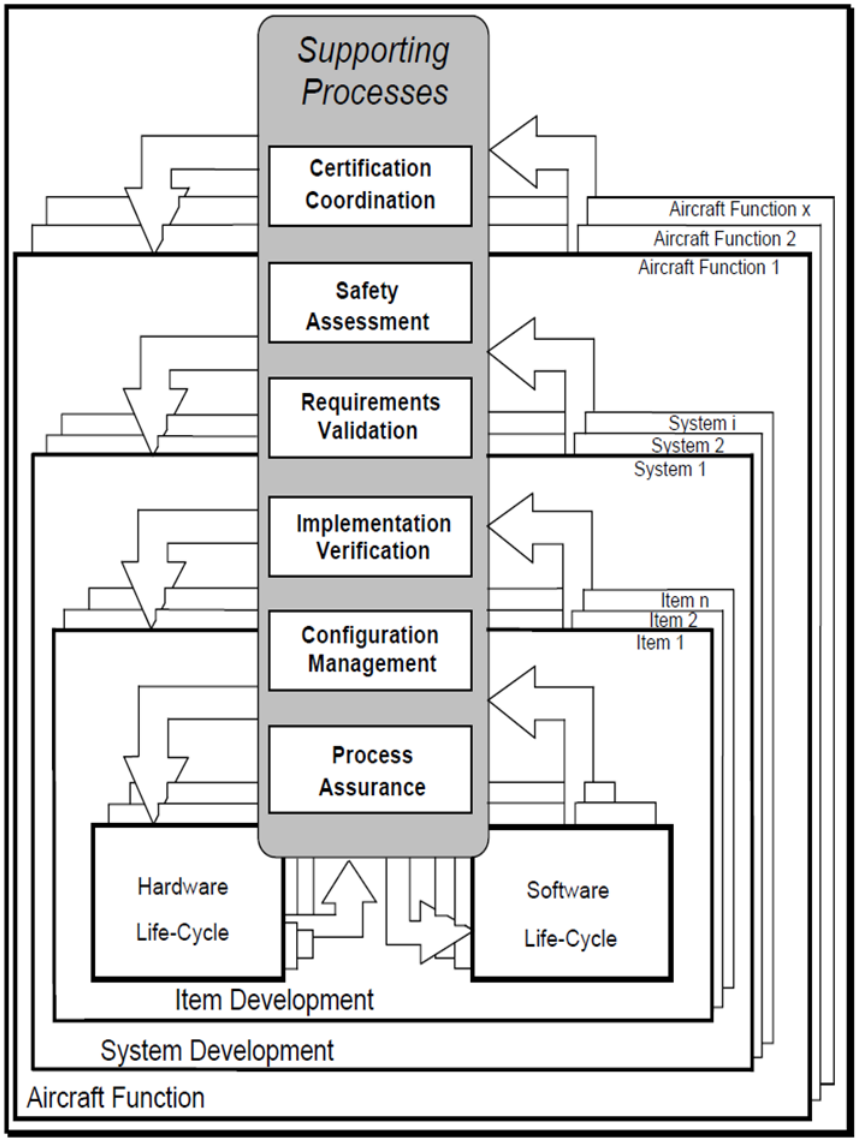


Figure 3.6: Aircraft Function Implementation Process (SAE, 2010)

illustrated in Figure 3.7.

- *Product.* A product is a smaller-scale element than a system which cannot be assessed for safety outside the context of its use, e.g. an engine or its components.
- *Service.* A service can be any activity which is applied to a system, e.g. maintaining/updating military vehicles.
- *System.* A system is a combination of elements that are used together to perform a given task or achieve a specific purpose. The elements may include personnel, procedures, materials, tools, products, facilities, and/or data as appropriate; e.g. an air traffic control facility with integrated radar and radio equipment.

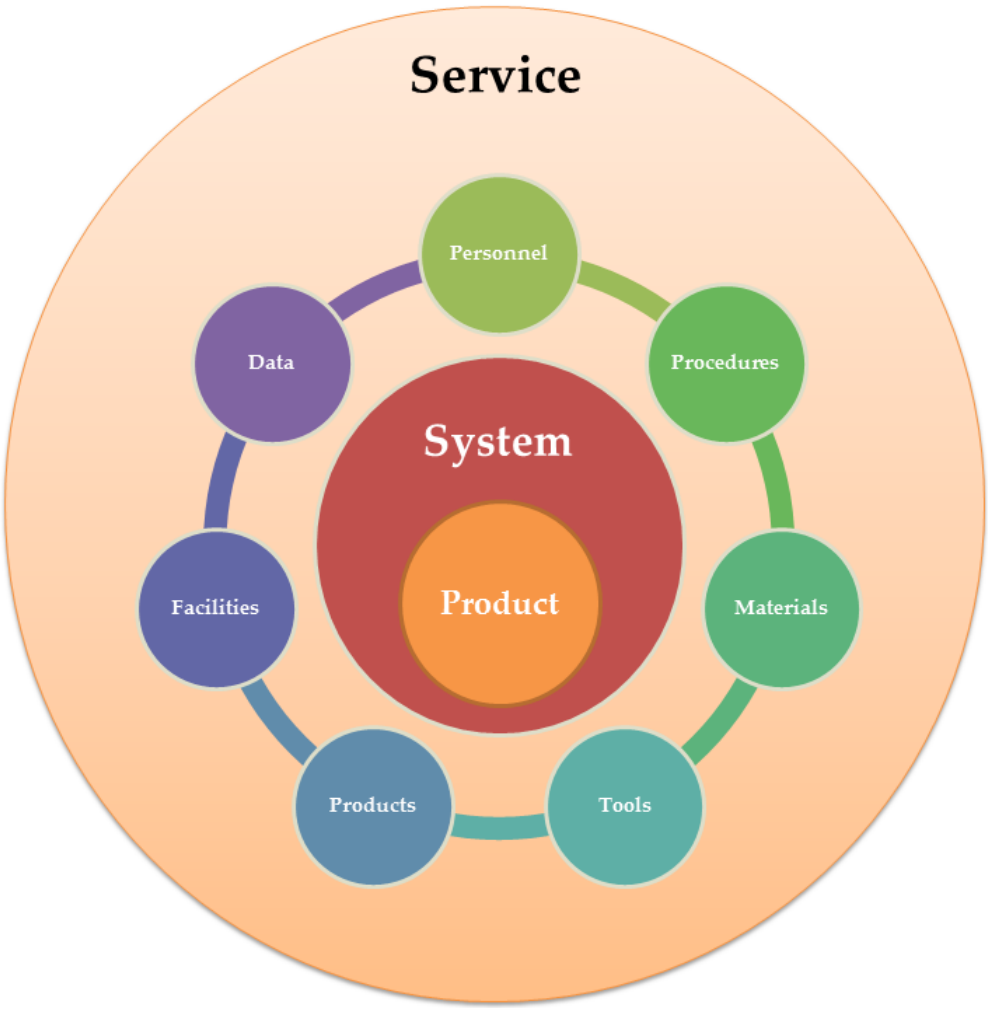


Figure 3.7: Products, Services and/or Systems (PSS) Relationship

[Sub-section text redacted]

3.2.2 Safety Cases and Safety Assessment Reports (SARs)

From a civil perspective in the UK, greater safety oversight by the Government appears to have been adopted after the Flixborough explosion in 1974[13] (Leveson, 2011). However, the concept of a 'safety case' was not introduced until the 1990 publication of the findings of the 1988 Piper Alpha disaster[14] (Leveson, 2011). They have been adopted in the MOD since the mid-1990s; this was due to recommendations made within the Jones Report on Equipment Safety Assurance (Inge, 2007).

Safety Cases are a method to structure an argument which can help to establish confidence. However, it can be difficult to establish the level of safety evidence needed to gain sufficient confidence in a system. The *judgement* of the engineers and practitioners will influence the type and amount of evidence deemed necessary. Judgements will be shaped by knowledge of the regulations or previous reviews of similar systems. DS 00-56 has a requirement for a suitable Safety Case to be generated as part of the wider Safety Management System (SMS).

The use of Safety Cases is common within other safety-related domains, e.g. within the civil nuclear sector. The UK nuclear regulator (Office for Nuclear Regulation (ONR)) describe a Safety Case as *a logical and hierarchical set of documents that describes risk* and *clearly sets out the trail from safety claims through arguments to evidence* (ONR, 2016). There are similar definitions and objectives across a number of domains which implement Safety Cases. This is positive, as it infers that the approaches developed for this thesis for the military airborne software assurance domain may be compatible with wider domains, e.g. civil airborne, and how they adopt Safety Case evidence.

The MOD Acquisition Safety and Environmental Management System (AESMS, 2017) states that the following evidence should be included within a Safety Case:

- Safety requirements have been met.
- Safety requirements are valid.
- Assessment undertaken is valid.
- Derived safety requirements are traceable and sufficient.
- Safety Management System (SMS) is defined.
- Suitable staff competencies.
- Applicable legislation, regulations, and policies have been adhered to.

[13] Information on the incident can be found in HSE (2018*b*).
[14] Information on the incident can be found in The Guardian (2013).

- All contractual safety requirements are met.

DS 00-56 also states the need to generate a Safety Case Report (SCR) which summarises the arguments and evidence of the Safety Case and documents progress against the safety programme. A SCR is produced/updated at key milestones of the procurement process, e.g. Initial Gate[15]. Within the air domain, the focus of the researches problem of interest, Safety Cases are referred to as Safety Assessments with SCRs referred to as Safety Assessment Reports (SARs) (UK MOD, 2014*c*). Further information is contained within the MAA Regulatory Article (RA) 1205[16] (MAA, 2017*c*) with its implication for DTs defined within RA 1220[17] (MAA, 2016*a*).

DS 00-56, in the context of Safety Cases and SCRs, makes reference to *arguments* being supported by *evidence*. Kelly (2011) defines supporting *evidence* as the "results of observing, analysing, testing, simulating and estimating the properties of a system that provide the fundamental information from which safety can be inferred". High-level *arguments* are defined as the "explanation of how the available evidence can be reasonably interpreted as indicating acceptable safety".

DS 00-970 (UK MOD, 2014*a*)[18] contains safety-related requirements for what are termed PEs. DS 00-970 also states that all aspects of the PE should be supported by a SAR as described within DS 00-56.

3.2.3 Assurance of Programmable Elements (PEs)

3.2.3.1 Defence Standard 00-55

DS 00-55 (UK MOD, 2014*b*) defines PEs as those PSS that are implemented in *software* or *programmable hardware* which includes any device that can be customised, e.g. Application-Specific Integrated Circuits (ASICs), Programmable Logic Devices (PLDs), and Field-Programmable Gate Arrays (FPGAs). DS 00-55 aims to provide the requirements and guidance for the achievement, assurance, and management of safety of PEs. The guidance is contained in a number of sections:

- PEs safety management, including: PEs safety governance and PEs information sharing.
- General requirements, including: requirements definition and PEs failure assessment.

[15]The commencement of the *assessment* phase within the MOD acquisition model. The acquisition model is termed Concept, Assessment, Demonstration, Manufacture, In-Service, Disposal (CADMID).

[16]RA 1205: Air System Safety Cases.

[17]RA 1220: Project Team Airworthiness and Safety.

[18]UK MOD (2014*a*) is focussed on the "Design and Airworthiness Requirements for Service Aircraft", (Part 13: Military Common Fit Equipment).

- Standards selection, agreement and design integrity.
- PEs management, including: safety requirement traceability and PEs risk reduction and mitigation.
- Assurance. PEs safety evidence and PEs safety assurance reporting.

[Sub-section text redacted]

3.2.3.2 Defence Standard 00-970

PEs are also considered within DS 00-970 (UK MOD, 2014*a*). This standard is stated by the MAA as being the default certification specification for MOD military registered aircraft (MAA, 2018*b*). Specifically, for military airborne software and CEH the relevant guidance is contained within requirement 1.7 of DS 00-970. Requirement 1.7 refers to "Safety Related Programmable Elements".

Compliance to requirement 1.7 of DS 00-970 is defined within four sub-sections which must be considered for any procurement (MAA, 2015). The supporting evidence will be agreed between the DTs and the MAA.

- *System level safety considerations.* At the system level, the Safety Assessment process should define the top level safety requirements and the design objectives of the PEs. These are detailed in the guidance within ARP4761[19] (SAE, 1996) and ARP4754A[20] (SAE, 2010).
- *Airworthiness related cyber security assurance.* DO-326A[21] (RTCA, 2014*a*) and the associated DO-356[22] (RTCA, 2014*b*) combined with arguments made in relation to JSP440[23] (UK MOD, 2001) should be used as an Acceptable Means of Compliance (AMC) with the cyber security requirements of DS 00-56.
- *Safety Related Software (SRS) assurance.* DO-178C[24] (RTCA, 2011*a*) and its supplements[25] can be considered an AMC to provide design assurance of airborne SRS when

[19] Aerospace Recommended Practice 4761. Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment.
[20] Aerospace Recommended Practice 4754A. Guidelines for Development of Civil Aircraft and Systems.
[21] DO-326A. Airworthiness Security Process Specification.
[22] DO-356. Airworthiness Security Methods and Considerations.
[23] JSP 440. The Defence Manual of Security.
[24] DO-178C. Software Considerations in Airborne Systems and Equipment Certification.
[25] DO-178C supplements:

- DO-248C. Supporting Information for DO-178C and DO-278A (RTCA, 2011*b*).
- DO-330. Software Tool Qualification Considerations (RTCA, 2011*d*).

supported by a robust, documented, and auditable Safety Assessment as described within DS 00-56.

- *Safety related CEH assurance.* DO-254[26] (RTCA, 2000) can be considered an AMC to provide design assurance of airborne safety related CEH when supported by a robust, documented, and auditable Safety Assessment as described within DS 00-56 (as is the case for the safety-related software).

Figure 3.8 shows the route to determining the software, CEH, safety assessment process, and airworthiness-related cyber security qualification requirements. The research has a focus on software assurance[27] and it is these which are to be subject to the initial research analysis. However, diverse evidence is still valid in gaining compliance to other elements of requirement 1.7 within DS 00-970, i.e. system level safety considerations and cyber security assurance.

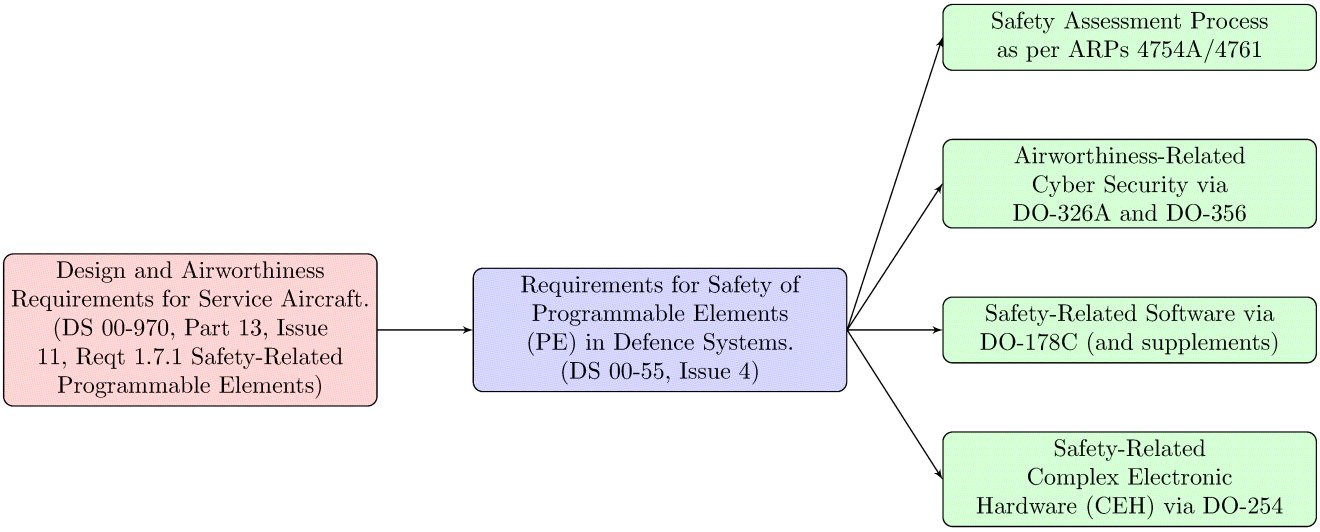


Figure 3.8: Flow of MOD Military Airborne Software/CEH Qualification Guidance (based upon MAA (2015))

– DO-331. Model-Based Development and Verification Supplement to DO-178C and DO-278A (RTCA, 2011*e*).

– DO-332. Object-Orientated Technology and Related Techniques Supplement to DO-178C and DO-278A (RTCA, 2011*f*).

– DO-333. Formal Methods Supplement to DO-178C and DO-278A (RTCA, 2011*g*).

[26]DO-254. Design Assurance Guidance for Airborne Electronic Hardware.
[27]Also, potentially CEH assurance if the research principles are deemed appropriate.

3.2.3.3 DO-178C

DO-178C is considered an AMC within DS 00-970 to assess software process life-cycles. The guideline aims to allow software life-cycles to be developed and reviewed in a consistent manner by numerous stakeholders, e.g. Independent Technical Evaluators (ITEs).

DO-178C contains a number of objectives, stated in the appendix tables within the guideline, which cover the software life-cycle processes. The life-cycle processes intended to be addressed by the guideline should always be considered as part of the wider systems and hardware processes. The key elements of DO-178C are:

- Software Planning.
- Software Development Processes:
  - Software requirements process.
  - Software design process.
  - Software coding process.
  - Integration process.
- Integral Processes:
  - Software verification process.
  - Software configuration management process.
  - Software quality assurance process.
  - Certification liaison process.

The amount of objectives to be complied with, and hence the level of robustness, is commensurate with the required software integrity. The integrity of the software is expected to be determined via the Safety Assessment process as a result of the application of ARP4754A (SAE, 2010). In particular, the integrity of the software is based upon its failure condition. The failure condition categories are in Table 3.1.

The failure condition categories are linked to five software levels (A-E), commonly referred to as DALs[28]. The relationship between the failure condition categories and the software levels is shown in Table 3.2. The system safety assessment process determines the failure condition by the extent of the anomalous software behaviour causing, or contributing to, a failure of the system function.

[28]Section 1.1 contains information on the number of DO-178C objectives for each DAL.

Category	**Failure Condition Description**
Catastrophic	*Multiple* fatalities (usually with loss of the aircraft).
Hazardous	Reduce the capability of the aircraft or the ability of the flight crew to cope with adverse operating conditions to the extent of: • *Large reduction* in safety margins or functional capabilities. • Physical distress or *excessive* workload (aircrew cannot perform tasks accurately or completely). • Serious or fatal injury to a *small number* of occupants other than the flight crew.
Major	Reduce the capability of the aircraft or the ability of the flight crew to cope with adverse operating conditions to the extent of: • *Significant reduction* in safety margins or functional capabilities. • *Significant increase* in workload (aircrew efficiency impacted, discomfort to the flight crew). • Physical distress to passengers or cabin crew (possibly including injuries).
Minor	Would *not significantly* reduce aircraft safety and involve actions which are well within their capabilities. Extent of the failure condition includes: • *Slight reduction* in safety margins or functional capabilities. • *Slight increase* in workload. • Some physical discomfort passengers or cabin crew.
No Safety Effect	Would have *no effect* on aircraft safety. No increase to crew workload would occur.

Table 3.1: Failure Condition Categories (based upon Marcil (2012) and Rierson (2017))

Level	**Category**
A	Catastrophic
B	Hazardous
C	Major
D	Minor
E	No Safety Effect

Table 3.2: Software Levels and Failure Condition Categories (SAE, 2010)

The defined DAL for a software development (e.g. via DO-178C) is based upon the identification and assignment of a probabilistic safety objective as part of the system safety engineering activities; for example via ARP-4761 (SAE, 1996) (Ledinot et al., 2016). In essence, it is accepted that the appropriate development of the software to a defined level of rigour (e.g. meeting *all* objectives within the DO-178C guideline) can help to achieve the probabilistic safety objective. However, it should be noted that this is not a *scientific* justification that it achieves the probabilistic objective, but a consensual approach which

is currently suitable given the lack of a more sound software reliability estimation method. This review of the software to judge conformance consists of a blend of *qualitative* assessment (e.g. the requirements specification) and *quantitative* assessment (e.g. test results).

Using ARP-4761 (SAE, 1996) definitions the assignment of a DAL is providing an indication (based upon a low level of risk) of the likelihood of an occurrence of a defined severity of failure per flight hour. As an example, within Table 3.3 the quantitative probability of a *catastrophic* event occurring should be *extremely improbable*, i.e. less than 0.000000001 per flight hour. Increases to these levels of probability starts to introduce high to medium levels of risk (FAA, 2000).

Level	**Category**	**Probability (Quantitative)**[1]	**Probability (Descriptive)**[2]
A	Catastrophic	1.0E-9	Extremely Improbable
B	Hazardous	1.0E-7	Improbable
C	Major	1.0E-5	Improbable
D	Minor	1.0E-3 / 1.0	Probable

Note(s): 1. Per flight hour. 2. Federal Aviation Administration (FAA) definitions.

Table 3.3: Software Levels, Failure Condition Categories, and Quantitative/Descriptive Probabilities (based upon SAE (1996))

Qualitative assessments are based upon judgements and the thesis provides an approach to *capture* the judgements in a consistent manner. The potential intent of any tool developed as part of this thesis is to make judgements on the *acceptability* of any evidence. It provides a confidence level built by *structured consensus building*. In order to achieve this a method may be required to *quantify* these judgements. In a strictest sense the safety approaches adopted at a system safety engineering level (e.g. ARP-4761), at a software level (e.g. DO-178C), and within this thesis are not providing a *statistical confidence* as such[29]. Nor does the thesis provide strict probabilities of any other kind, it provides an informal confidence level built by a process of *structured consensus building*.

3.2.3.4 DO-254

DO-254 (RTCA, 2000) was created due to electronic hardware within safety-critical aircraft systems becoming increasingly complex. Due to the complexity of the hardware within the systems they may be increasingly at risk of design errors causing failure conditions. The DO-254 guideline is an attempt to address/reduce these risks by providing design assurance guidance for the development of airborne CEH.

[29] *Statistical confidence* signifies the level of confidence in the method of constructing an interval and in some cases is the defined "margin of error" of an experiments sample size (Sauro and Lewis, 2016).

A hardware design life-cycle is described within DO-254; however a preferred life-cycle or structure is not provided, e.g. waterfall. The design life-cycle processes are as follows:

- Hardware Planning Processes.
- Hardware Design Processes:
  - Requirements capture.
  - Conceptual design process.
  - Detailed design process.
  - Implementation process.
  - Production transition process.
  - Acceptance test.
- Validation and Verification Process:
  - Validation process.
  - Verification process.
  - Validation and verification methods.
- Configuration Management Process.
- Process Assurance.
- Certification Liaison Process.

In essence, the hardware life-cycle follows a similar process-based approach as the software life-cycle outlined within DO-178 A, B, and C[30].

Within DO-254 the definitions of the failure conditions are the same as those stated within Table 3.1. The relationship between the integrity levels and the failure conditions correspond accordingly, as with DO-178C, but with the addition of the term *Severe-Major* to accompany the *Hazardous* failure condition. Appendix A of DO-254 states the level of *rigour* which should be applied for the commensurate hardware level.

Figure 3.9 shows the link between the system development process (including the safety assessment), the software life-cycle process (DO-178C), and the hardware development process (DO-254). The software and hardware functions/requirements and design information feed into the system development process.

[30]Release dates for the DO-178 versions: A-1985, B-1992, C-2011.

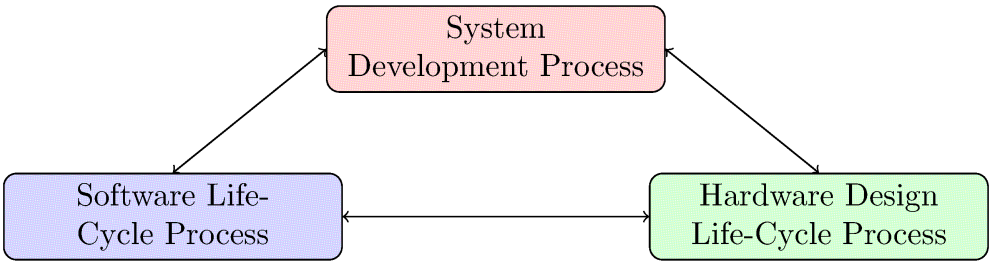


Figure 3.9: Link Between System, Software, and Hardware Processes

3.3 Current Evidential Approaches: Areas for Enhancement

The MOD software and CEH safety assurance[31] approaches have a focus on using the process-based objectives within guidelines such as DO-178C and DO-254. There is scope to use wider non-process evidence, e.g. PSH via CAST-1 (CAST, 1998)[32].

Gaining a full picture of process compliance, e.g. via DO-178C, requires access to the life-cycle data artefacts, e.g. a Software Development Plan (SDP), which are produced as a result of the software development life-cycle. This allows the development and testing activities to be sufficiently understood for the implemented software. There are a number of approaches to gain a level of software safety assurance confidence and SMEs may judge these diverse approaches to be more suitable than a process-based method.

There are also factors which may require the MOD to gain additional evidence in *support* of a process-based claim. Examples of such factors include: the evidence being in a different form; such as the chosen development process not meeting the specific objectives within DO-178C.

The following sub-sections provide a number of non-exhaustive examples of why additional evidence may be needed in-lieu of the process-based approach. Some of the examples may also be applicable to the provision of the non-process based evidence itself. However, this reaffirms the need for wider sets of diverse evidence to be captured to mitigate such issues; e.g. Commercial-Off-The-Shelf (COTS)/Military-Off-The-Shelf (MOTS) systems may have limited *life-cycle evidence* available for *immediate* review. This may be due to the vendor applying limitations on to whom the information can be released. The same release restrictions may also apply to the *in-service data* of the same COTS/MOTS system.

[31] Via DS 00-970 (UK MOD, 2014*a*) which is focussed on the "Design and Airworthiness Requirements for Service Aircraft" (Part 13: Military Common Fit Equipment). Further information on the MOD assurance process is contained within sub-section 3.2.3.

[32] Certification Authorities Software Team (CAST) Position Paper (CAST-1) - Guidance for Assessing the Software Aspects of Product Service History of Airborne Systems and Equipment.

3.3.1 Software Level

3.3.1.1 Adoption of Novel Technology

Novel technology is defined as an approach, or item of equipment, which has not undergone any form of military airborne assurance (this definition is partly derived from Weaver, Kelly and Mayo (2006)). For these types of technology there may need to be a fundamental assessment of the assurance requirements. This is due to the principles in the extant standards/guidelines not being able to be applied in full, e.g. Radack, Tiedeman and Parkinson (2019).

There is a desire to adopt novel technologies due to the performance benefits and the subsequent positive impact on capability. An example of a *novel technology* is the use of MC processors which, due to the increase in the number of cores on a single processor, have perceived assurance risks in terms of determinism[33]. MC processors have been used within domains which are not safety-critical for a number of years. However, the use of MC processors within the UK military airborne domain is novel, and due to this there are assurance considerations that need to be addressed.

Due to the very nature of their novelty some of these technologies may not fully adhere to the extant guidelines/standards objectives. Due to this there may be a reluctance for these technologies to be accepted by regulators[34]. This demonstrates how extant guidelines/standards and safety assurance requirements could restrict the adoption of novel technologies. However, novel approaches may need to become part of future platforms and the safety assurance regulations must take this into account, e.g. the adoption of MC processors. The RE and the EngD Industrial Supervisor[35] have been active in ascertaining 'solutions' to adopting such novel technologies and have provided thoughts on a stepped process for MC processor qualification[36].

There is a balance between the *extant* assurance requirements/constraints and the technical designs/solutions to provide a capability. Could the use of diverse evidence assist with achieving this balance? Further information on this topic can be found in sub-section 7.2.8.

3.3.1.2 Procurements via International Partners/Vendors

There are a number of benefits to adopting a procurement strategy for the UK military airborne domain which favours the use of international or multi-national partners and vendors.

[33]For example, ensuring that the Worst-Case Execution Time (WCET) is predictable.

[34]A technology maybe 'novel' to one domain but not another (e.g. use within the rail industry but not the airborne domain) and therefore the transferring of applicable evidence could be possible.

[35]Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems.

[36]Presented at the High-Integrity Software (HIS) 2016 event (Bristol, UK). For further information see http://www.his-2017.co.uk/session/multi-core-mc-processor-qualification-for-safety-critical-systems.

Such benefits include shared costs and shared exploitation of Research and Development (R&D). The F-35 Lightning II platform is an example of a UK platform which is procured as part of a set of partner countries[37]. The use of diverse evidence can assist where an SME would like to form a view on the software assurance level in-lieu of process-based evidence; e.g. IPR and International Traffic in Arms Regulations (ITAR) considerations may delay the provision of process-based evidence.

Procurements from international partners provide opportunities for enhancements to the current evidential approaches. The software life-cycle approach and the supporting activities which are implemented may have additional forms of evidence which can be part of the judgement on the software safety assurance confidence. An example of such evidence are third-party V&V activities which are performed that, although are not part of an extant guideline/standard, will still assist in forming a judgement on the confidence.

Other forms of evidence may be suitable to derive a confidence level as it can be gained in a timely manner or the evidence may be judged to be of greater relevance. The additional information can assist with qualifying the software when compared to the extant process-based requirements. Therefore, MOD software and CEH assurance process may benefit from the adoption of alternative approaches which use diverse evidence.

3.3.1.3 Adoption of Commercial-Off-The-Shelf (COTS) and Military-Off-The-Shelf (MOTS) Equipment

For over a decade there has been a clear directive for the MOD to use COTS/MOTS components where possible. Indeed, the 2005 Defence Industry Strategy (UK MOD, 2015) stated that MOD should "increasingly accept COTS technology". There are a number of well understood benefits to the use of COTS/MOTS, such as reduced costs due to the scale of production.

The software and CEH of COTS/MOTS products are not necessarily developed to fully meet regulatory requirements[38]. There are instances where full compliance is achieved with relevant evidence (e.g. SDP) being available for review by regulators. However, in general, COTS equipment may require additional evidence to support a safety judgement. The guidelines do take into consideration the use of COTS equipment but the expectation is that they meet the process-based requirements or that a PSH argument is established. Due to the nature of COTS equipment the potential reduced level of available process evidence means there is a benefit in taking into account a more diverse range of evidence.

[37]See the following for further information: https://www.f35.com/global.

[38]This topic is well reported: such as within the nuclear domain, e.g. Picca (2018); civilian airborne domain, e.g. Daniels (2018); and in the general software assurance field, e.g. Menon (2018), Hall (2018), Spriggs (2018), and Barker (2018).

3.3.1.4 Reuse of Pre-Existing Evidence

In some scenarios the MOD may need to qualify the software or CEH of equipment that has previous history and evidence from recognised bodies, e.g. European Aviation Safety Agency (EASA) or the FAA. From the perspective of the FAA a Technical Standard Order (TSO)[39] would be issued to signify that the design and production of the specific system/part has been approved. Part of the TSO process would be to gain confidence in the development activities conducted by the equipment vendor.

Recognising the approvals that have been granted via certification bodies would be of value. In addition, the recognition also has wider benefits such as the reduction in rework and a reduction in any costs associated with the certification efforts.

3.3.2 System Level: Safety three-Layered Framework (SLF)

The MOD safety assurance process is strong and defensible; however, there are potential areas for enhancement. The following sub-sections provide an outline of an approach to enhance the development of safety cases for MOD. The approach was published in two papers at the 9th IET International Conference on System Safety and Cyber Security (2014). The papers were written in collaboration between the RE, two Dstl colleagues[40], and a colleague from Atomic Weapons Establishment (AWE)[41]. In addition, the findings have been discussed with the REs academic supervisors[42].

The SLF adopts diverse evidence to ensure that a suitable level of confidence can be gained via the safety assurance process. This can be achieved in circumstances where there may be a *need to know* only certain information and there may be IPR constraints. The theory of the SLF allows a solution for *system* level safety evidence issues and it has applicable lessons for the *software* safety domain.

3.3.2.1 SLF: Context

The complexity of a system normally determines the scale and severity of its safety assurance challenges. The argument for why a given system, or SoS, is safe is captured in a Safety Case which links the claims to the supporting evidence via arguments.

[39] A TSO is a minimum performance standard for specified materials, parts, and appliances used on civil aircraft (FAA, 2018).

[40] Paul Caseley (Dstl, Senior Fellow) and Dr Mark Hadley (Dstl, Senior Principal Scientist in Software Systems).

[41] Helen Auld.

[42] Dr John May (University of Bristol, Reader in Safety Systems) and Dr Theo Tryfonas (University of Bristol, Reader in Smart Cities).

There are challenges to making a Safety Case comprehensible when it is significant in size (e.g. hundreds of pages long) as it is often difficult, if not impossible, for an individual to develop a complete understanding of it. In addition, due to the complexities of military systems which the MOD procures there will be many organisations and individuals involved in creating the Safety Case. Therefore, there is a real possibility that the application and the environment of the system could be viewed differently by each of the contributors. An issue such as this could introduce doubt into the validity of the Safety Case due to the incompatible contexts.

There have also been wider issues with the development of Safety Cases. An independent review, led by Charles Haddon-Cave QC, into the loss of the UK Royal Air Force (RAF) Nimrod MR2 Aircraft XV230 in Afghanistan in 2006 found that there was "a Safety Case regime which [was] ineffective and wasteful" (Haddon-Cave, 2009). The safety case was criticised in the review and it was found that the belief that the Nimrod was *safe anyway* and *acceptably safe to operate* blinded many of those involved in the Nimrod Safety Case.

Additional evidence supports the claim that the way in which Safety Cases have been traditionally developed is not sufficient. Steinzor (2010) identified a number of lessons from the 2010 BP Deepwater Horizon Facility (DHF) oil spill in the Gulf of Mexico. Steinzor (2010) states that the general size and approach to the development of British safety cases was not satisfactory. Safety Cases from related case studies are referred to within Steinzor (2010) as being: "bulky"; that "typically a safety case for a medium-size North Sea production platform covers anywhere from 390-610 pages"; and there can be an "over-reliance on *cookie-cutter* prototypes of critical documentation".

It could be argued that the types of issues experienced with safety cases for complex SoS will increase in the future due to the growth in the complexity of systems. Also, the increased globalisation of the manufacturing industry means that a given system may contain components or sub-systems provided from numerous nations. The systems may not even be operated in these nations. These concepts introduce issues which may evolve due to differing safety cultures[43].

The aim of the SLF is to only allow *need to know* safety related information to be exchanged. This allows the IPR of the systems to be protected, e.g. from other nations. In addition, a SoS may also include legacy[44] elements which may not be fully understood or documented to the level which would be required to comply with modern standards.

The SLF attempts to enhance the development of safety cases to deal with complex

[43]The issue of safety cultures at *inter*- and *intra*-organisational levels is not a new one and is still subject to debate, e.g. within Rollenhagen and Wahlstrom (2007) and Jaiswal et al. (2018).

[44]The term *legacy* in this context being "of, relating to, or being a previous... computer system" or "of, relating to, associated with, or carried over from an earlier time, technology etc" (Merriam-Webster, 2018).

SoS which may have components of varying provenance. The SLF manages the system complexity by using modularisation of the system with well-defined interfaces to make the individual safety cases comprehensible. The approach has been informed by the work of the Industrial Avionics Working Group (IAWG) (Fenn et al., 2007).

The SLF is an approach which characterises the system's internal and external safety relationships as arguments, engineering models, and via detailed analysis. This abstract model was developed to enable suitable levels of assurance confidence to be achieved for high integrity SoS.

Engineering models are adopted to understand the interfaces, to allow the safety of the entire system to be understood, and to allow scenarios to be executed in a modelled environment. The SLF also recognises the need for detailed analysis to be integrated into the model to ensure the validity of the key safety arguments. Consistency is checked when integrating modules to ensure the safety case is valid for all of the stakeholders, for a given application in a given environment.

3.3.2.2 SLF: Introduction

Modular safety cases provide a number of benefits over traditional monolithic safety cases, e.g. ease of construction and a focus on integration boundaries (IAWG, 2010). The IAWG process attempted to provide a more structured and efficient method but there are still limitations with the approach. As an example, the safety case module arguments *reference* the evidence rather than having the evidence direct and explicit as part of the model[45]. The SLF aims to meet the limitations of existing processes by applying a modularised approach at a systems level and allowing a greater level of detail to be exposed in a coherent manner for the assurance of safety-critical system components.

The SLF consists of a flow of information which is fed from the top level down to populate each stage; modular safety cases, engineering models, and detailed analysis such as formal models (as shown in Figure 3.10).

As the stages are developed for each iteration the level of safety assurance and confidence increases due to the greater depth of analysis which is conducted. The SLF layers consist of:

- *Understanding and argument layer.* Provides a language that the customer/user can understand for the safety, security, and dependability of the system element.
- *Modelling evidence layer.* Relevant engineering designs of the arguments showing that the designers have understood the interfaces and safety requirements.

[45]Whereas, within the SLF a potential element of the abstract model *is* the evidence.

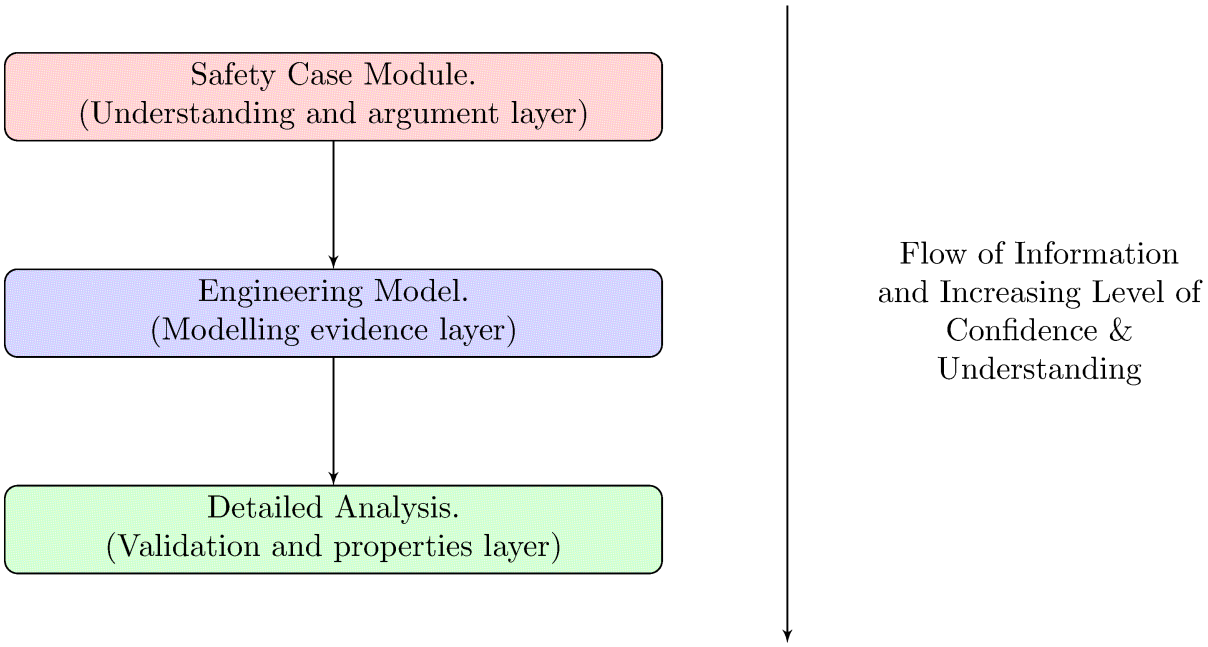


Figure 3.10: Outline of the SLF Hierarchy

- *Validation layer*. Detailed analysis of the essential safety properties and provides the necessary confidence for specialists to justify their safety, security, or dependability assertions.

The SLF helps to form a judgement on the interface interactions, how the relationships occur, the formal proof of the relationships, and how the relationships can be met. It enables dependency relationships to be defined at appropriate supply interfaces and at differing levels of the SLF, e.g. between different systems and suppliers of an ABS (see Figure 3.11).

3.3.2.3 SLF: The Layers

3.3.2.3.1 Modular Safety Cases: An Understanding and Argument Layer

The IAWG refined the modular safety case concept via research into avionic software assurance (Fenn et al., 2007). Utilising this approach allows elements of systems to be separated into modules. This is dependent on each module having a well defined interface definition and that the definitions are derived in a consistent manner for the neighbouring systems. Architectures are used to identify the interfaces and interactions in the system. This allows a safety case to be built on external outputs from each of the modules in a uniform manner. These interfaces must be captured in a consistent and unambiguous form to allow sub-system and end-to-end safety assessments to be made. Interfaces can be defined us-

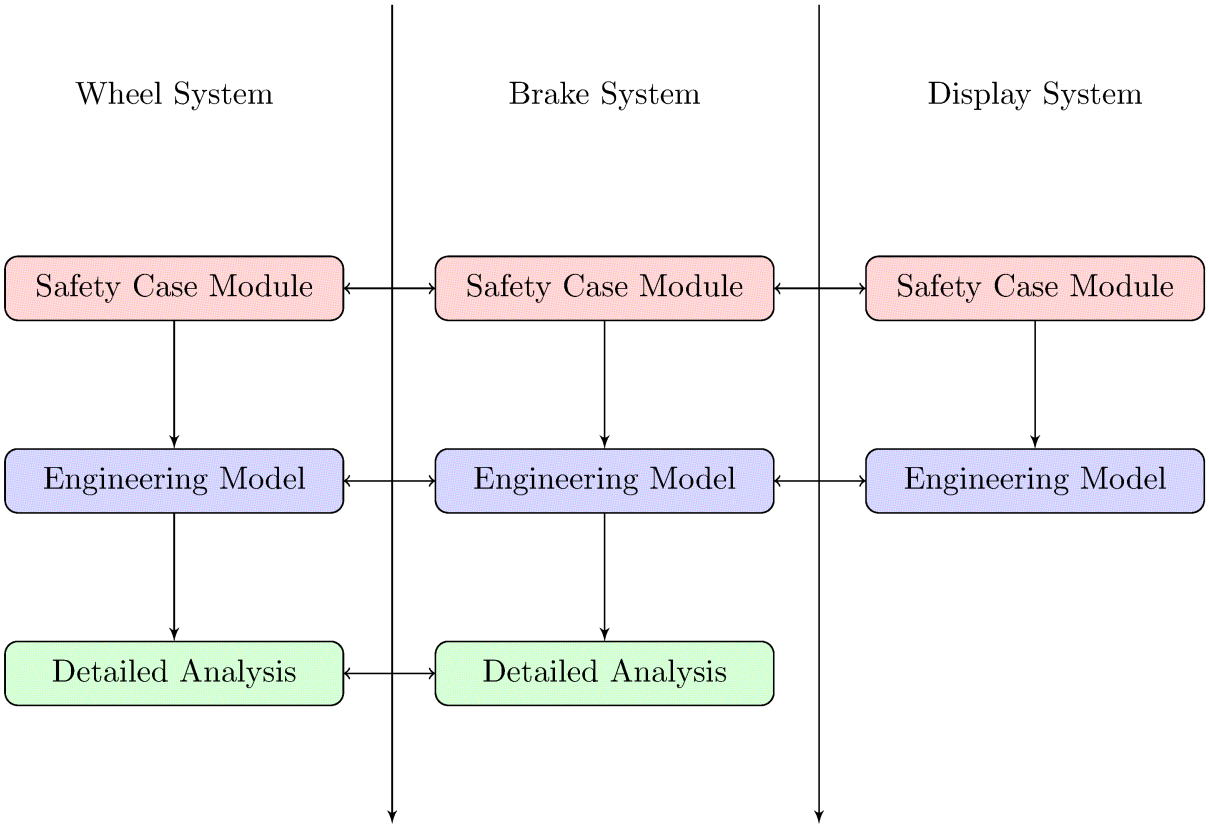


Figure 3.11: SLF Supplier Interfaces for an ABS Example

ing contract based arguments, such as Dependency-Guarantee Relationships (DGRs)[46] and Dependency-Guarantee Contracts (DGCs)[47]. The contract arguments should specify all of the safety related behaviour of the system element, e.g. failure modes.

Provided that the external and internal interfaces are fully defined, access to the internal arguments in the safety case modules should not be necessary as sufficient information is available to derive assurance conclusions. This allows suppliers to ensure confidentiality while making essential information accessible on a *need to know* basis.

For low risk safety related system elements the modular safety case layer and its defined interfaces may be sufficient for the overall safety case. The concept of only sharing sufficient information in each SLF layer is shown in Figure 3.11 with the Display System not requiring any detailed analysis modelling, e.g. due to a lower assurance integrity level.

[46]A module will guarantee to exhibit specific behaviour provided specific dependencies hold. If the dependencies on which the module relies do not hold then it cannot uphold its guarantees. For a safety case argument to use the guarantees provided by a module it will be necessary to confirm that the dependencies the module relies on have been satisfied (Fenn et al., 2007).

[47]DGCs state an association where one module's Dependency is satisfied by anothers Guarantee (Fenn et al., 2007).

3.3.2.3.2 Engineering Models: A Modelling Evidence Layer

To support the modular argument layer with any significant safety related implications evidence from engineering modelling using standard engineering processes may be used, e.g. MathWorks Simulink[48]/Stateflow[49] or Unified Modelling Language (UML). The engineering models may be part of the existing design, or an adaptation, but should accurately specify the contract based safety interfaces and essential internal behaviour of the system. These models will help inform trade-offs in the safety architecture and the safety functions. Along with standard safety analysis, the models will provide engineering evidence of safety functions and safety related interfaces.

Depending on the form of the engineering modelling process, scenarios can be executed through animation, including human responses to particular failure modes. This animation can occur across all systems at the engineering layer of the SLF. Only the interface elements of the engineering models need to be shared between development teams. This maintains the separation of the detailed design and hence ensures confidentiality.

Where COTS elements are being used in the system it may be necessary to create models that accurately reflect the declared COTS interface. These models can be used to explore the safety implications.

3.3.2.3.3 Analysis Models: Validation of Properties Layer

The safety behaviour of high integrity interfaces may require detailed analysis. Formal mathematical notation allows safety critical arguments and engineering models to be further refined into a notation which is precise and unambiguous. It can be used to produce well defined pre- and post-predicates for contract-based interface declarations. This ensures correctness and understanding of the interfaces between modules in the safety case. It can also be used to capture the module specifications for parts of the system that are essential to understand the safety properties; e.g. the mode or state of a system element which influences the interface[50].

Combining the interface and module specifications which are captured in formal notation can allow formal safety (and security) models to be created. These models can be for systems composed of bespoke and COTS elements. This can be achieved even when the technical details of these elements are not known. These models can be configured to different scenarios to understand safety (and security) arguments. Although the true nature of the COTS is not known, it may be possible to explore areas of concern and then mitigate through architectural

[48]For further information see MathWorks (2014*a*).

[49]For further information see MathWorks (2014*b*).

[50]The use of formal methods is also prevalent for solutions being adopted for evolving concepts such as Machine Learning, e.g. as detailed at the FLOC (2018) summit.

re-design or additional bespoke functionality. The information captured via these models can be used with other diverse evidence, e.g. system test results, to derive a level of confidence.

3.3.2.4 Inclusion of Legacy Systems

The SLF allows for legacy systems to be included within a wider SoS safety case as long as the interface between the legacy system and the rest of the SoS is well defined and understood. This is due to the process assuring the *interfaces* between contributing systems with each having a self-contained safety argument. This means that there can be a varied quantity of information provided. Appropriate evidence must be gained before it can be considered part of the wider SoS.

3.3.2.5 SLF: Implementation Process

The SLF is implemented via a six stage process:

1. *Define Safety Case Goals and Functions.* Goals are specified based upon the top-level safety requirements. Functions are captured which specify the characteristics or actions which the system must perform to contribute to the system safety.

2. *Define Safety Case Regions.* Regions are created for each safety function and include the elements of the system required to provide the safety function.

3. *Define Safety Case Modules.* Safety case modules exist to meet the safety goals and include all of the elements required to meet that goal. DGRs are created for each module.

4. *Create Arguments.* Safety arguments are generated which can be formed of existing evidence, engineering models, and/or detailed analysis.

5. *Create Engineering Models.* Engineering models are created for each module and for a system as a whole.

6. *Integrate.* Arguments are integrated for the entire system. DGCs are created for the safety case module information flows.

It should be noted that at any point in the process additional goals, safety functions, modules or interfaces may be identified. In these cases they should be added to the framework and the process restarted from the appropriate point. The flow of the six stage process is shown in Figure 3.12.

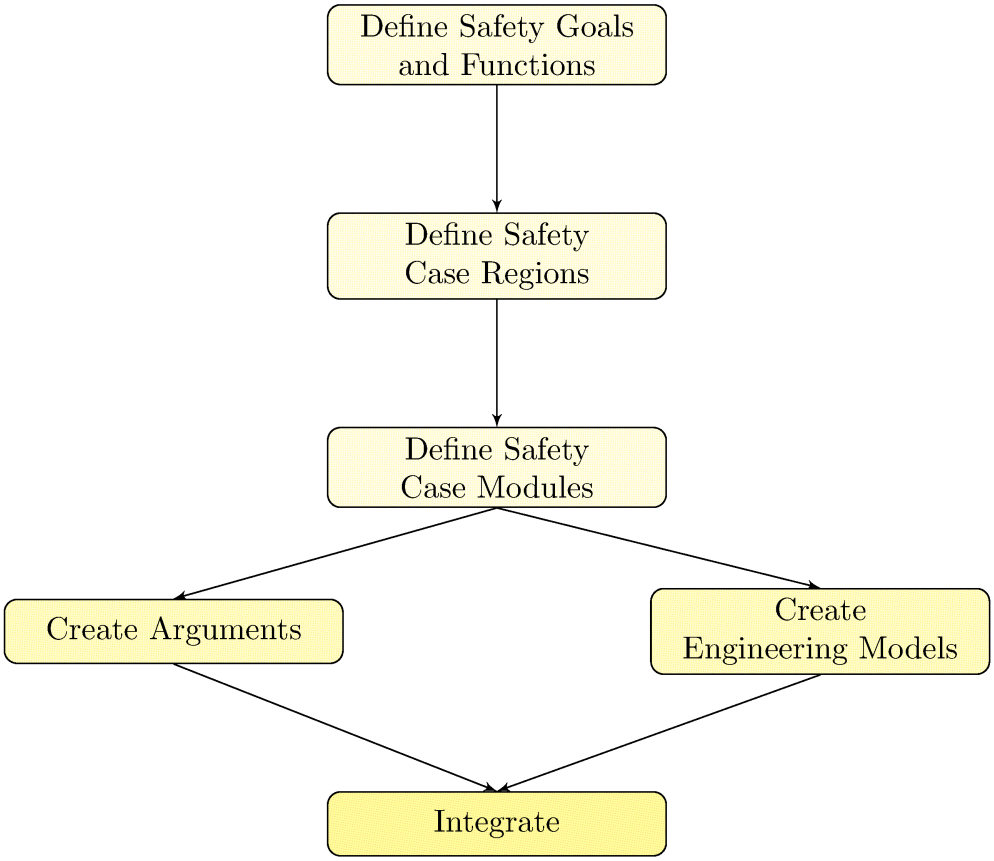


Figure 3.12: SLF Implementation Flow

3.3.2.6 SLF: Conclusions

The papers produced as a result of the SLF research contain greater detail on the SLF stages and includes a worked case study. The main points from the papers have been included within this section to outline the concept of the approach. A number of clear advantages of the SLF process have been identified:

- The SLF enables arguments to be formed with traceability to the evidence and their models, if applicable. These arguments are grouped together into modules allowing each to be understood in isolation and the whole to be understood by an individual.
- The structure of the SLF allows the information to be presented in a consistent and unambiguous way. The approach utilises, where possible, existing processes and terminology. *Need to know* is maintained with the SLF as only relevant information is released between systems.
- In certain circumstances the *entire* SoS safety case may be used to assure the system.

This assurance would be provided by relevant stakeholders under limited information access.

- The processes supporting the SLF allow various stakeholders with a variety of levels of interest to appreciate and comprehend the systems and their interactions.
- The SLF allows for human factors to be included within the modular safety cases and engineering models. Human Factors safety case concepts (UK MOD, 2008) and traditional methods can be used in conjunction with the SLF.
- Integration risk can be reduced as the SLF concept model should ensure that interfaces are described with common terminology and definitions. This provides confidence that those responsible for the separate parts of the SoS are utilising the same language for system boundary descriptions.
- The SLF and the supporting processes can be adopted for a range of SoS with varying degrees of complexity. They have utility for commercial and military safety case generation in line with the Information Set Safety Summary (ISSS) concept[51] within DS 00-56.
- Legacy, bespoke, and COTS systems can be integrated to form the SoS safety case. Issues surrounding IPR and the sharing of sensitive information can be circumvented due to the method in which information is exchanged within the SLF.

The SLF has acted as a 'stepping stone' towards fulfilling the *grand tour* question. It has allowed lessons and principles to be learnt from a *systems* level with the further research applying these lessons at a lower *software* level. The SLF has not acted as a significant contribution to the research but it has assisted in allowing a number of key elements of the DSF and the rationale for the approach to be articulated.

The SLF has shown that there are non-process based approaches that can be applied in order to generate and review safety assurance evidence. At a systems level there is a need to gain an overall level of confidence in the safety assurance of the wider-system (or SoS). To gain this confidence there is a necessity to review supporting evidence (which can be determined by extant standards) and to reason with the *available* evidence. Evidence which is a requirement within a standard/guideline is not always available for review due to IPR/ITAR issues, for example. The SLF has shown that systems can trace to a number

[51]The ISSS contains the core information which third parties, e.g. the MOD or system integrators, need to know in order to discharge their safety responsibilities. The ISSS would normally also contain information about failure modes (McDermid and Williams, 2014).

of evidential types (not only process-based) to generate a safety assurance judgement; in essence supporting the use of diverse evidence.

The models within the SLF (e.g. engineering models) have also shown the value in only exposing certain degrees of evidence to varying stakeholders. Different stakeholders will have diverging levels of interest in the supporting evidence and the claimed confidence in the system[52]. A software level assurance solution such as the DSF can feed into a wider systems level assurance approach such as the SLF.

3.4 Summary of the Background and Problem of Interest

There is expected to be an increasing level of software implemented within future military avionic platforms. This software requires suitable levels of assurance due to its safety-criticality. In addition, a focus on software as an area of assurance is legitimate due to the fundamental role that software plays within a wider system. Any research which aims to enhance software assurance needs to focus on this area to avoid *diluting* the findings. There are a number of factors which may limit the level of evidence that can be provided to the MOD in support of a process-based claim. These factors range from the required evidence being of a different form to that expected (e.g. a different development process being adopted) to there being limitations to the release of supporting evidence (e.g. due to IPR). In such circumstances, the use of diverse evidence is a promising approach.

The *safety management requirements for defence systems* are stated within MOD standard DS 00-56 (UK MOD, 2014*c*). The standard provides the requirements and guidance for the achievement, assurance, and management of safety. DS 00-56 is focussed on allowing compliant PSS to be acquired. The acknowledgement within DS 00-56 that products, services, and systems all require suitable requirements and guidelines allows compliance to be captured for a range of MOD capabilities and not only the traditional forms of equipment.

Safety Cases are adopted within DS 00-56 to allow arguments to be structured to assist with establishing confidence. Knowing the level of evidence required to support such Safety Cases can be challenging. Other domains' use and definition of Safety Cases has a similarity with MOD. This is positive in that the approaches developed for this research should have applicability to other domains. A Safety Case should include a range of evidence which allows confidence to be gained that the safety requirements have been achieved. Within DS

[52]The DSF (Chapter 8) and the 'Wheel' (sub-section 7.4) are based upon the concept of stakeholder dialogue to prompt discussion and to gain an understanding of the Line Replaceable Units (LRUs) and the evidence

00-56 a key premise is that *arguments* are supported by *evidence*.

The requirements and guidance for the achievement, assurance, and management of safety of PEs is stated within DS 00-55 (UK MOD, 2014*b*). The failure of unintended behaviour of PEs within PSS must be managed. The level of confidence for any failures or unintended behaviours to be within acceptable levels of probabilities can be achieved by establishing a sufficient *design integrity* of the PEs. Process-based Recognised Good Practice (RGP) is the preferred method within DS 00-55 for compliance to the standard. However, DS 00-55 also allows other forms of compliance to be proposed which theoretically permits the use of wider evidence to express *equivalent* safety findings.

The assurance of PEs are also considered within DS 00-970 (UK MOD, 2014*a*) which contains four sub-sections which must be considered for any procurement of safety related PEs. The sub-sections relate to system level safety considerations[53], airworthiness related cyber security assurance[54], safety related software assurance[55], and safety related CEH assurance[56]. The research has a main aim to support the confidence which can be gained in safety related software assurance. The current defined approaches for compliance, e.g. DO-178C, are process-based.

The SLF is an approach to enhance how MOD develops suitable Safety Cases. The SLF adopts the use of diverse evidence to ensure that appropriate levels of confidence can be gained via the safety assurance process. The framework is based upon a modular approach using well defined interfaces which allows *need to know* safety related information to be exchanged. Evidence is *directly explicit* within the SLF as the engineering models and detailed analysis models contain supporting evidence relevant to the safety argument. The use of the SLF at the *system* safety level supports the adoption of wider diverse evidence to achieve suitable confidence in the safety assurance at a *software* level.

Chapter 3 has responded to the research sub-question: *What is the current approach to system safety assurance within the UK defence domain and are there alternative system-level approaches?*.

[53]Based upon the guidance within ARP4754A (SAE, 2010) and ARP4761 (SAE, 1996).
[54]Based upon the guidance within DO-326A (RTCA, 2014*a*).
[55]Based upon the guidance within DO-178C (RTCA, 2011*a*).
[56]Based upon the guidance within DO-254 (RTCA, 2000).

Chapter 4

Diversity as a Concept and Scope for Further Investigation

How diverse[1] evidence can be judged, measured, and combined needs to be placed within the context of existing approaches. It is important to review how any new approach can compliment and/or progress the current work in this area. Any intervention to enhance software safety assurance will need to be considered within the scope of what can be feasibly achieved and how any research findings could be adopted.

This chapter will provide information on:

- *Support for Diversity as a Concept.* A definition of *diversity* is provided and a brief overview of how diverse evidence is used in other domains. The concept of software design diversity is also briefly discussed.
- *Software Evidence Diversity and Quantification of Assurance Arguments.* The benefits of *diversity* are considered from a review of the academic studies conducted to date. There is a need for further investigation due to the current research findings in this area being inconclusive.
- *Providing Support to Assist Decision Making.* How a DSF could assist by providing enhancements to current software safety assurance approaches. The benefits that a framework may provide are briefly described.

4.1 Support for Diversity as a Concept

The concept of *diversity* is one which is commonly used. However, the scenarios in which diversity is applied results in subtle and fundamental differences in its application.

[1]The term *diverse* being to have variety or to be assorted (Collins Dictionary, 1995*i*).

4.1.1 What is Diversity?

The terms *diverse*, *diversity*, and *diversify* are referred to throughout this thesis and they are defined as:

- Diverse, *adj.* 1. Having variety, assorted. 2. Distinct in kind (Collins Dictionary, 1995*i*).
- Diversity, *n.* 1. The state or quality of being different or varied. 2. A point of difference (Collins Dictionary, 1995*k*).

The key premise in this thesis is that there is a *variety* of *distinct* evidence that can be put forward to gain confidence in a system. The variety of evidence is not currently used consistently within the safety assurance process. The definition of the term *diversify* is also relevant and articulates an interesting perspective.

- Diversify, *vb.* 1. (tr.) To create different forms of; variegate; vary. 2. (of an enterprise) To vary (products, operations, etc.). in order to spread risk, expand, etc. 3. To distribute (investments) among several securities in order to spread risk (Collins Dictionary, 1995*j*).

Thus, diversification may *reduce dependencies* on certain evidence types and *increase confidence* in the software (or PEs) through a number of evidential strands.

4.1.2 Use of Diverse Evidence within Related Domains

A number of domains which contain safety-critical software have process-based artefacts as their predominant form of evidence. However, when required, other evidential strands such as in-service data can be used to provide partial or full alternative approaches to process-based evidence. An example includes the assessment of ground-based avionics systems which are judged against DO-278A (RTCA, 2011*c*)[2]. Any shortfalls in evidence can be mitigated by in-service data. In addition, evidence can be adopted which is based upon the Suitably Qualified and Experienced Personnel (SQEP)[3] status of those performing the activities, i.e. those that produce the system/software. As an example, a vendor may claim that software engineers must have attained certain qualifications (e.g. Bachelor of Science (BSc)) and a level of direct experience (e.g. 10 years) to develop safety-critical source code. This can help gain confidence as part of a wider set of supporting evidence.

[2]DO-278A is a process-based guideline within the air-traffic domain.

[3]The term *SQEP* originated in the UK nuclear industry and is now used within other domains (such as defence). SQEP provides recognition that the skills and understanding of an individual can be relied upon to resolve (or advise on) a technical problem to the required standards (NAFEMS, 2018).

Other domains are not consistently in a position to review the process-based evidence for a system of interest. Indeed, it is not only process-based evidence that is unable to be consistently judged but also other evidence of interest, e.g. SQEP and in-service data. In these instances the approach is to gather a range of the *available* evidence to form a suitable judgement. Any limitations on the evidence would also then limit the confidence that could be placed in the software/CEH. This type of scenario would commonly be more prevalent with COTS equipment.

Further, more detailed, analysis of the non-safety and safety domains which conduct evidence assessments are contained in sub-sections 5.2 and 6.2.

4.1.3 Software Design Diversity

Diversity from a software design perspective allows key properties, such as *resilience*, to be considered within a risk reduction strategy (Popov et al., 2014). Diversity among redundant components can reduce the risk of common failures caused by design faults. Without diversity these faults would be replicated in the redundant components. In essence, in the simplest case two or more versions of these components are built independently and placed within an architecture so that the system will perform correctly (or safely) if a certain 'quorum' of components do. To ensure independence between the versions, measures need to be applied for the development processes and the designs to be as different as possible (Popov et al., 2014).

From a nuclear industry perspective the concept of *diversity* is one which is summarised well within a report on "Defence-in-Depth and Diversity: Challenges Related to I&C Architecture" (World Nuclear Association, 2018). The definition of *diversity* provided in the report is:

- The presence of two or more independent (redundant) systems or components to perform an identified function, where the different systems or components have different attributes so as to reduce the possibility of common cause failure, including common mode failure (World Nuclear Association, 2018).

The World Nuclear Association (2018) report identifies a number of attributes of *diversity* in the context of the above definition, such as: human, life-cycle, software, and equipment. The attributes and their associated criteria are contained within Figure 4.1. The focus of the diversity is in relation to mitigating CCF concerns with Instrumentation and Controls (I&Cs).

The topic of software diversity has been subject to a relatively substantial level of research from the 1970s to the present (Baudry and Monperrus, 2015) with various approaches devised

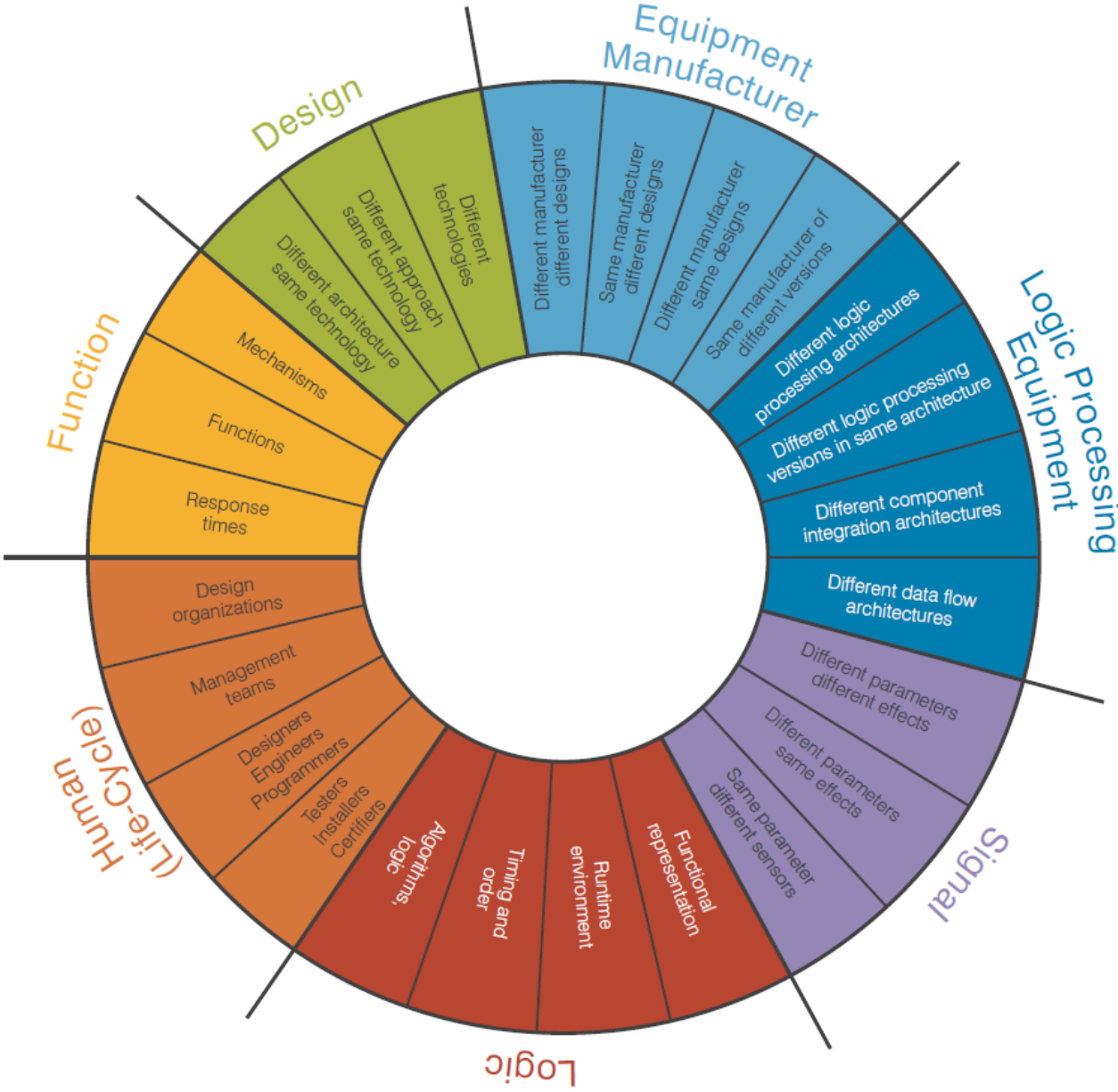


Figure 4.1: Diversity Attributes in Relation to CCF Mitigations (World Nuclear Association, 2018)

to implement the *diversity* concept. Approaches include N-version programming or the concept of a sufficiently *simple* secondary system as backup to a more complex primary, e.g. Littlewood et al. (2000)[4].

Within existing research diversity has been used to influence a number of software design areas. This includes influencing the architectural features of the software itself to the development and testing of the design. The application of design diversity can be used to generate reliability assessments and to inform *proof* arguments (e.g. Littlewood (2000)). Specific architectural properties of the design can be established such as with the creation of very asymmetric architectures, and associated safety and reliability claims, which is claimed by Littlewood, Popov and Strigini (1999) to have benefit over two fully functional diverse subsystems[5]. Design diversity has been applied to phases of the development process, including: requirements analysis to maintenance/evolution (Schaefer et al., 2012); and the use of multiple-version software to validate specifications (Avizienis, Lyu and Schutz, 1995). The combination of software detection techniques which optimise differing fault finding procedures (Littlewood et al., 2000) have also strengthened design diversity. Kharchenko and Brezhnev (2015) and Kharchenko (2016) claim that diversity allows not only reliability and safety improvements, but also *security*.

However, within the existing research there are perceived issues with software design diversity. The usefulness of software diversity is a controversial topic in some forums due to the results from empirical evidence showing varied effectiveness (Popov et al., 2014). In addition, the efficacy of the approach to mitigate some design concerns is debated, e.g. in relation to CCF or software reliability (Baudry and Monperrus, 2015). It is also acknowledged that it is problematic to assess the reliability of: fault-tolerant systems (Littlewood, Popov and Strigini, 2001), design diversity (Popov and Strigini, 2001), and multiple-version software (Popov et al., 2003). *Diversity* is not a simple concept to measure, although there are supporting techniques proposed such as fault simulation/failure searching (Luping, May and Hughes, 2001) and also fault injection (Luping and May, 2014). However, it remains that software diversity is not an alternative for careful quality control (Bishop, 1995).

Despite these perceived issues regarding the effectiveness of software design diversity, from an overall academic and practical perspective diversity finds favour and has precedence. Obviously, the notion of software *design* diversity and software *assurance* diversity differs but clear linkages can be made to be able to articulate the benefits.

[4]A full review of software design diversity is not appropriate for this thesis as Baudry and Monperrus (2015) contains a recent and interesting review and should be referred to for an introduction to the topic. The point to note is that the concept of *diversity* is a commonly understood approach for providing resilience within safety-critical systems.

[5]Littlewood, Popov and Strigini (1999) describe *functional diversity* as having differing versions of the system design *and* a deliberate decision to make the inputs into the systems different.

4.2 Software Evidence Diversity and Quantification of Assurance Arguments

The concept of diversity to reduce risk via *variety* and *independence* is one that is subject to research in the field of software *assurance*. There have been various research studies to implement frameworks and methodologies using both *qualitative* and *quantitative* measures of attributes. These aim to assist decision makers in understanding the confidence in the evidence and the subsequent safety claims. However, as will be shown, the results of these studies are far from conclusive and further work is required to demonstrate the advantages of a diverse evidence approach.

4.2.1 Diverse Evidence as a 'Good Thing'

There is general consensus within the safety-critical systems domain that the use of diverse evidence can be seen to be plausibly a "good thing" (Littlewood and Wright, 2007). However, the methodology and structures to accept and reason upon diverse evidence are subject to debate. There have been claims that such assertions do not have a theoretical underpinning (Littlewood and Wright, 2007). Diverse evidence does have scope to support a number of methods which construct assurance claims, for example methods based upon argumentation schemes (Yuan and Kelly, 2011) or dependability statements (Bloomfield and Littlewood, 2006).

Research has suggested that there are benefits of applying diverse software assurance evidence. An increase in the confidence of safety claims (Yuan and Kelly, 2011) is possible when compared to single evidential strands (Bloomfield and Littlewood, 2006, Littlewood and Wright, 2007). Indeed, it can be said that these research findings can be expected (Littlewood and Wright, 2007). However, research has also concluded that counter-intuitive results can occur when multi-legged arguments are constructed via probabilistic means (Littlewood and Wright, 2007)[6]. The richness of data gained via counter-intuitive results cannot be captured via methodologies which apply *qualitative* measurements of attributes as there is no underpinning data to lead to such observations.

Analytical treatment of certain methodologies, e.g. Bayesian Belief Network (BBN), can allow *what-if* calculations on the effects of additional diverse legs (Littlewood and Wright, 2007). Forms of *what-if* analysis could also be applied to assist decision makers to understand the impact of any changes on the diverse evidence prior to their implementation. Cost Benefit Analysis (CBA) could prevent expensive evidence collection measures being instigated to

[6]Noting that counter-intuitive results are valid and valuable outputs from any research as they offer the ability to learn about the problem of interest.

mitigate perceived shortfalls (Littlewood and Wright, 2007). The research to date provides some potential methods and benefits but they are not clearly articulated.

4.2.2 Quantifying Confidence Within Evidence Assessments

Quantifying confidence[7] within a safety argument has significant backing based upon the number of models which have been proposed, e.g. Guo (2003), and the varied paradigms to implement such methods, e.g. BBN. However, the concept does not gain universal support; e.g. Graydon and Holloway (2016) states that further validation is required before it can be recommended as part of a basis for deciding whether an assurance argument justifies fielding a critical system. Due to the intrinsic link between the use of diverse evidence and that of quantifying confidence there are lessons which can be learnt from the current approaches.

There are number of research papers which are based upon the topic of confidence judgement for safety assurance with some having a more specific aim to assess *software*-based systems. However, the direction and focus of the various forms of research are subtly different; for example: to understand the *behaviour* of the strands of evidence, to articulate the *benefits* of quantifying confidence, or to investigate appropriate methods to *combine* evidence.

Littlewood and Wright (2007) adopt the use of *multi-legged* arguments to support the generation of dependability claims for software-based systems. The use of *multi-legged* arguments and BBN allows Littlewood and Wright (2007) to manipulate the data to identify any *behaviour of interest*. Bloomfield and Littlewood (2006) examine the benefits of *diversity* to an assurance case approach. Similarly, Hobbs and Lloyd (2012) adopt the use of BBN to build an assurance case to illustrate the *benefits* that such an approach can have, such as flexibility and expressive capabilities. Dahll (2000) utilises BBN from a different perspective which is to examine how to *combine* disparate sources of information.

Other research takes the position of claiming that a *specific* approach is the most suitable to capture safety assessment confidence values. Dempster−Shafer Theory (DST) is favoured by Zeng, Lu and Zhong (2013) and conversely Guiochet, Hoang and Kaâniche (2015) and Guo (2003) propose BBN as suitable methods to capture the safety case confidence. However, Guiochet, Hoang and Kaâniche (2015), Wang, Guiochet and Motet (2017), Yuan et al. (2017) each have a focus on the most appropriate *argument forms*, e.g. redundancy, as the method to drive the propagation of confidence assessment. The method to achieve the confidence propagation varies, e.g. BBN, DST, and subjective logic. Duan et al. (2015) also adopts subjective logic to represent confidence in assurance case evidence but also adopts the use

[7]The term *confidence* in this instance is to have "trust in a thing" and "showing [a level of] certainty" (Collins Dictionary, 1995*b*).

of the Joseng Opinion Triangle and Beta Distribution.

Zhao et al. (2012) propose a wider set of analysis to derive the evidence and the confidence judgements with BBN forming a part in a limited chain of steps, e.g. creation of model instances. The approach to adopt methods such as Evidential Reasoning (ER) within a wider approach is also utilised by Nair et al. (2015) and Cyra and Gorski (2008).

The varying approaches taken by the existing research supports the use of confidence measurement for safety assessment. In addition, there are a number of particular areas of the existing research which provide further and more detailed scope to target particular shortfalls in the domain.

4.2.3 Scope for Further Investigation

There are various studies which describe the specific concept of quantifying confidence within an assurance argument. There is a need to understand the limitations of such approaches so that any devised solution advances these concepts. The following sub-sections provide a review of a body of current work which claim to use diverse software evidence, quantitative measurement, or include related topics. Any identified limitations can lead to further investigation and may form part of the thesis outputs.

4.2.3.1 Existing Approaches Lack Sufficient Case Studies and Adoption

Littlewood and Wright (2007) acknowledges that there are limitations with their proposed approach with the research being based upon a simplified and idealised example[8]. Research concepts have been presented using incomplete treatment for the sake of brevity (Delic, Mazzanti and Strigini, 1995) or provide a simplification of the captured attributes (Bouissou, Martin and Ourghanlian, 1999). In some instances only special examples of diverse argument legs were reviewed as part of the research (Bloomfield and Littlewood, 2006) with others using very small sample sets, e.g. evaluation via a single example (Yamamoto, 2015). Also, it has been observed that areas of research adopt simplifying assumptions to demonstrate concepts (Bloomfield and Littlewood, 2006).

There is a need for the research concepts to be applied to problems which are: (a) proportionate to the *scale* of the target systems, and (b) validated via a sufficient *quantity* of case studies. In essence, if the concept is to be applied to *multiple* forms of evidence with numerous diversity legs then the case studies should reflect the intended scale. Also, a suitable number of case studies should be adopted which illustrate and *test* a variety of the features proposed by the concept.

[8]In addition, the concept is limited to a model of a two-legged argument with a deliberate simplification of the real situation with each of the argument legs unrealistically simplified.

The proposed approaches, e.g. by Littlewood and Wright (2007), may be conceptually valid but they do not remove the need for further research in this area. The simplification of examples, or not implementing a scaled concept, is valid for research which is at a broad conceptual level. However, to fundamentally challenge how evidence diversity and argumentation is formed in the real world needs additional research.

4.2.3.2 Existing Approaches Require Further Analysis

The proposed approaches to diversity within the existing research need to be *further* refined before they are considered effective, e.g. in dependability arguments (Bloomfield and Littlewood, 2006). A number of the existing approaches have not been validated via: operational testing (Fenton et al., 1998), real projects (Neil and Fenton, 1996), or real-life case studies (Delic, Mazzanti and Strigini, 1995). Concepts proposed, in some instances, only addressed a small part of a large and difficult problem (Bloomfield and Littlewood, 2006). Some of the research had further work ongoing at the time to provide additional confidence in the approach (Weaver et al., 2005). Others required additional research to validate the approach (Neil and Fenton, 1996) or to undergo years of calibration to yield reliable forecasts (Bouissou, Martin and Ourghanlian, 1999). Many of the proposed approaches aimed to progress the concept of assurance quantification but there is further research required to validate the findings (Yuan and Kelly, 2011).

The methodologies which are adopted within the research are also very much open to discussion. Approaches such as BBNs need to be treated with "great respect and humility" as it is possible to have a false sense of certainty and security from the numerical values (Littlewood and Wright, 2007). Ayoub et al. (2013) adopts DST which results in an assumption on evidence nodes acting *independently* with further work required to consider the dependencies.

There is acknowledgement within the research that their concepts are part of a wider decision making process and so they need to be placed within context. Safety assurance argument decisions can not be based upon the sole output of the techniques (Fenton and Neil, 2001). Many examples of the existing research require further analysis to establish the role which the research outputs would have within the wider safety assurance argument. This acknowledgement that further analysis is required certainly validates the need for further research due, in part, to the inconclusive study outputs.

4.2.3.3 Diverse Evidence Analysis is not Currently Fully Understood

The current literature indicates that the analysis of diverse evidence is complex, e.g. the relationships between one-legged and two-legged argument topologies are not trivial (Little-

wood and Wright, 2007). Also, an indicator of the complexity of the concept is that the models developed to implement a diverse evidence argument can lead to unexpected and counter-intuitive outcomes (Littlewood and Wright, 2007). Models can be difficult to comprehend even with simplistic case studies and may need analytical treatment to understand fully (Littlewood and Wright, 2007). It is difficult to determine which attributes should be part of a diverse argument. There is also difficulty in determining and measuring these attributes, e.g. an *independence* attribute can be *extremely elusive* (Littlewood and Wright, 2007). How to structure diverse arguments also has no consensus due to the myriad of case studies and approaches. It is not clear how suitable goals for a diverse argument can be achieved, e.g. the ability to make *claims* at the highest structural level in the argument (Bloomfield and Littlewood, 2006). How goals and numerical values should be formally expressed is also contentious (Bloomfield and Littlewood, 2006).

How to measure diverse arguments is also subject to some debate. Some studies state that the size and complexity of safety arguments, when combined with subjective composition, means that it is difficult to *quantitatively* assess (Weaver, Fenn and Kelly, 2003). Indeed, Weaver, Fenn and Kelly (2003) believe that a *qualitative* approach is sufficient and not unreasonably burdensome on those that create and assess the arguments. Delic, Mazzanti and Strigini (1995) supports this view, in that the *quantitative* safety evaluation of software products is difficult, and as a result the software safety case is usually a weak link in the demonstration of system safety. In contrast, other approaches use *quantitative* assessments which need analytical treatment to determine the nuances of the evidence and the relationships (Bloomfield and Littlewood, 2006, Littlewood and Wright, 2007). These divergent views illustrate the difficulty in finding a definitive approach to measure confidence within diverse evidence.

The general consensus is that diversity is a useful paradigm; however, there is a caveat: it is not a panacea for the problems which exist when building dependability cases (Bloomfield and Littlewood, 2006). There is a limit to what diverse evidence can achieve and therefore any proposed solution must be cognisant of this. Any proposed concept should be scrutinised to ensure that *naive trust* is not placed in the results (Bloomfield and Littlewood, 2006).

4.2.3.4 Expert Judgement is Needed

The existing research provides different views on the layer of abstraction that the expert judgements can be applied to; e.g. if there is a need to form an opinion only on the *overall* quantified confidence level[9] or opinions being formed and propagated at *all levels* of the

[9]In essence, drawing conclusions only on the *root* node of the evidence set, if presented as a tree structure.

review process[10]. In either case, expert judgements are required from a number of safety assessment stakeholders[11]. However, there are a wider set of actors who can influence the ability of the system to meet the assurance requirements. As an example, there is evidence that good managers and designers can determine the difference between failure and success, in the context of software quality (Neil and Fenton, 1996).

The quality of any software is determined at the design and implementation stages, with those that make judgements on the software quality, e.g. safety assessment stakeholders attempting to determine the quality of what has been *achieved*[12]. This illustrates that any approach should recognise its place in a wider decision making process which is influenced by a range of supporting evidence. This includes aspects such as: individual skills/experience; the novelty of the application; and time/cost constraints. Irrespective of the processes adopted factors such as these can impact the quality of the software (Yuan and Kelly, 2011).

Choosing which evidence is to be included within a diverse safety argument and the knowledge about the implications of such evidence is subject to expert judgement; in essence all relevant evidence should be accounted for. This selection process requires competence (Delic, Mazzanti and Strigini, 1995), e.g. the choice of PSH evidence over process-evidence and the expectations for the level of supporting data[13].

It is claimed that it can be difficult to assess *how* the final judgements on the evidence have been reached, and therefore much has to be taken on trust (Fenton et al., 1998). A structured approach to capture and reason upon the diverse evidence could allow third-parties to understand the final judgements made by SMEs. Methods to assess diverse evidence can act to *validate* judgements which have been made due to the systematic way in which the judgements are captured.

A method to quantify confidence to judge diverse evidence can provide a number of further benefits, such as: determining where the focus of effort is required for generating evidence; remove incoherent and inconsistent arguments; act as an approach to build consensus in argument structure and inference judgements; and can provide a shorthand for understanding the associated time and financial costs to create the arguments.

Haddon-Cave (2009) believes that quantitative risk assessment is an art and not a science with there being currently no way to avoid engineering judgement. There is general acknowledgement that expert judgement is *fundamental* to any safety argument. The assessment

[10]In essence, having to draw conclusions on the *leaf*, *parent*, and *root* nodes of the evidence set, if presented as a tree structure.

[11]The stakeholders include, but are not limited to, the MAA, DT, ITE, Independent Safety Auditor (ISA), and SMEs to derive the confidence level. These roles range from those that form direct opinions on the evidence to those with the authority and overall legal obligations for the release of any system/platform.

[12]It is also stated that software is designed rather than manufactured (Yuan and Kelly, 2011).

[13]Judging PSH data is not a trivial task with a need to consider such attributes as the level of change, the environment of use, and the error detection capability.

of software-based systems has long been acknowledged to be difficult and there is a great reliance upon expert judgement (Littlewood and Wright, 2007). It is envisaged by the RE that this reliance will continue.

4.2.3.5 Differing Underpinning Principles to the Existing Approaches

An important aspect to the existing research is that the underpinning approaches can differ, e.g. the argumentation schemes (Yuan and Kelly, 2011). In addition, the underlying methods to measure diverse evidence can also differ, e.g. the use of the term *relevance* as a metric (Hawkins and Kelly, 2010). The evidential legs, such as software requirements or reliability modelling, can differ in both the content and type of claim (Bloomfield and Littlewood, 2006); e.g. the evidence can be to gain a *prior* belief or to support the belief via in-use evidence.

Due to the various methods which can be adopted to analyse the diversity of evidence there needs to be assumptions made about the evidence itself (Bloomfield and Littlewood, 2006). These assumptions should be formally captured with restrictions placed on any results, e.g. assumptions regarding the *independence* of DST evidence. Any restrictions on the results would also, therefore place restrictions on the level of confidence which can be claimed (Bloomfield and Littlewood, 2006).

The methods to understand diverse evidence and how it interrelates have also been subject to debate within the research. The concepts of *dependence* and *independence* play an important role in determining the levels of confidence that are derived from multi-legged arguments (Bloomfield and Littlewood, 2006). There is also the concept of *relevance* for how the evidence impacts on the parent goal (Weaver, Fenn and Kelly, 2003). The attributes and measurements implemented in the research do not provide a clear approach as there is no dominant theory which emerges.

4.2.3.6 Differing Applications of the Existing Approaches

The various methods which make judgements on software assurance evidence differ subtly in intent. The adoption of diverse evidence can allow a greater insight into how the results of the arguments have been determined (Littlewood and Wright, 2007). It can also improve the understandability and repeatability of assessments, due to the judgements being represented by mathematical models (Bouissou, Martin and Ourghanlian, 1999). Bloomfield and Littlewood (2006) examined how diversity might be used to increase confidence in *dependability* claims (reliability and safety) and specifically how a probabilistic approach, when successfully applied to design diversity, can equate to a diversity of safety argument.

Other research has discussed how to combine sources of information in the safety assessment of software-based systems via BBNs (Dahll, 2000). Research has looked at improving the assessments of dependability claims of software intensive safety-critical systems by taking account of, and combining, the many types of evidence available, e.g. failure data and competence of the development team (Hall et al., 1992).

Within the research there has also been a focus on the structure of the argument. The theory being that the argument structure allows stakeholders to determine if individual items of evidence are needed. It also allows reviewers to determine if the evidence satisfies the requirements (Weaver et al., 2005). The structured approach of Goal Structured Notation (GSN) is used to determine what evidence is required to satisfy requirements whilst being applicable to a variety of evidence-based software engineering approaches (Weaver et al., 2005). In addition, patterns have been adopted for the safety argument structures based upon tiered models with a focus on the software safety requirements themselves (Hawkins and Kelly, 2010).

It is clear that the aims and the intent of the research which adopts diverse evidence have differing objectives. This indicates that the utility of diverse evidence structures can have a broad spectrum of purposes. This is positive for the thesis as it allows greater scope for how the outputs can be adopted.

4.2.3.7 Inconclusive Results from the Reasoning Under Uncertainty Approaches

Any approach which aims to quantify the confidence of evidence needs to account for the reasoning being undertaken with uncertainty. This is due to a person, in any situation, not having all of the information to describe, prescribe, or predict *deterministically* a system and its behaviour (Zimmermann, 2000). The sources of this uncertainty can be due to a number of reasons: a lack of information; too much information; and conflicting evidence etc (Zimmermann, 2000, Colyvan, 2008). There are a number of approaches to assist with this reasoning under uncertainty, e.g. DST; however, there is no definitive approach which is fully supported or adopted.

Existing research states that the approach adopted should be suitable for the context and be fit for the purpose in which the results will be applied; e.g. to *guide* decisions on the suitability of adopting software within an overall system safety claim. Wright and Cai (1994) states that for their approach there was scope for a number of different mathematical formalisms and no single formalism was shown to be superior to any other. Fenton et al. (1998) supports this as no single formalism for uncertainty was perfect for their purposes.

There are a number of approaches proposed to reason under uncertainty or to act as a

method to measure confidence, for example[14]:

- ER: Nair, Walkinshaw and Kelly (2014).
- BBN: Hobbs and Lloyd (2012) and others, e.g. Zhao et al. (2012).
- DST: Ayoub et al. (2013), Cyra and Gorski (2008), and Zeng, Lu and Zhong (2013).
- Subjective Logic: Duan et al. (2015).
- Weighted Average: Yamamoto (2015).
- Hierarchical Process Modelling (HPM): Yearworth et al. (2015).

How the reasoning approaches have been applied within the research is subject to debate. Graydon and Holloway (2016) believes that there is potentially limited confidence in a number of the proposed methods; e.g. there is no empirical evidence that the approach adopted by Guo (2003), adopting BBN, provides a trustworthy basis for deciding whether to release a system into service. This same assessment is also made by Graydon and Holloway (2016) for the approach and concepts proposed by Hobbs and Lloyd (2012).

The range of techniques adopted to achieve similar aims for confidence measurement demonstrates that there is not one universally accepted approach. The choice of approach can also be influenced by the precision and computational complexities associated with it.

4.2.3.8 Minimal Existing Visualisation Techniques within Existing Approaches

It is helpful for the propagation of the confidence calculations to be easily ascertained. This can assist with allowing stakeholders to *comprehend* the data. The original concept for GSN was to assist with the structuring of *qualitative* information and the supporting arguments to aid comprehension. Some approaches which are based upon structured arguments do attempt to quantify the measured confidence. However, there is currently a limited ability to visualise these confidence values.

Within some of the approaches there is no linkage between the evidence structure (e.g. GSN) and the actual quantified data which states the confidence levels, for example. There is a need for a combined approach to allow the data of interest to SMEs, e.g. the confidence values, to be comprehended. As an example, Ayoub et al. (2013), Duan et al. (2015), Zeng, Lu and Zhong (2013) adopt GSN to capture the structure and the relationships of the evidence. However, there is no direct link between the GSN representation and the

[14]Sub-section 8.2.4.4 contains a fuller review of relevant approaches applicable to this research

procedures to capture and calculate the confidence values. This same observation is equally applicable to approaches proposed by Hobbs and Lloyd (2012), Cyra and Gorski (2008).

Allowing stakeholders to gain an *understanding* of the evidence is key. Kirk (2016) states that there are three stages to facilitate understanding: *perceiving* (what does it show?); *interpreting* (what does it mean?); and *comprehension* (what does it mean to me?). The quantification of confidence should assist with these three stages where possible. There is scope to enhance how the analysis of the results is currently presented to stakeholders.

4.2.3.9 Inconsistent (or Lacking) Information on the Use of Outputs

A number of the proposed strategies do not specify *how* to use the results to determine if a system is sufficiently safe, nor do they say *which* attributes should be measured. This is the case for Cyra and Gorski (2008), Duan et al. (2015), Guiochet, Hoang and Kaâniche (2015), Yamamoto (2015), and Denney, Pai and Habli (2011). This lack of specification indicates that there is scope for a method which provides an output of direct utility. Approaches to quantify confidence or to judge diverse evidence should be intuitive and reflect how the *outputs* will be used by stakeholders.

4.2.3.10 Lack of Simplicity for Capturing Evidence within Existing Approaches

For an approach to add value to any decision making process it needs to be intuitive to how the judgements on the evidence, such as PEs, will be captured. Ideally, the data captured by the approach should be capable of being efficiently gathered by SMEs.

There are examples of approaches, e.g. Wang, Guiochet and Motet (2016), which allow data to be captured intuitively. However, there are other approaches; such as Guo (2003), Nair, Walkinshaw and Kelly (2014), Duan et al. (2015), which apply data capture methods that require interpretation and concepts of abstraction. A number of approaches require the evidence capturing process to be restructured to allow an overall confidence value to be gained. There can be a lack of simplicity to how the data is gathered.

4.2.3.11 Lack of Scalability to Capture and Assess Evidence within Existing Approaches

A number of the existing methods have reasonably complicated structures to quantify the confidence. This includes the way the attributes are captured, the evidence and attribute association, and the argument formation. As an example Wang, Guiochet and Motet (2017) use the argument types *dependent* and *redundant* to propagate the confidence within the structure. However, the argument types must be *specifically user defined* for each parent-

child relationship in the tree structure. This would be problematic and time consuming for a larger tree.

The use of diverse evidence within a safety argument involves multiple forms of evidence to be captured and assessed. For an approach to adopt diverse evidence it must be able to scale as more evidence is captured. A failure to do so would limit the applicability of the approach and place undue restrictions on the evidence which can be captured. Any approach should scale as more diverse evidence is captured to ensure a usable and valid solution.

4.2.3.12 Lack of Ability to Optimise and Conduct *What-If* Analysis within Existing Approaches

A key aspect of a decision making process is the assessment of alternative solutions (Turban, Sharda and Delen, 2010). Allowing judgements to be captured using a valid reasoning approach is only one element of a wider decision making process. It would be valuable for stakeholders to devise and assess alternative scenarios for confidence measurement via *what-if* analysis. The alternative options available to stakeholders could be to *gather* specific evidence, e.g. further design documentation, or to *generate* evidence, e.g conduct additional testing. The choice between these alternatives will be based upon a number of factors such as feasibility, cost, time, and the value that it provides.

The use of diverse evidence is to implicitly adopt a *wider* set of evidence. A wider set of evidence increases the number of potential solutions, also known as the *solution space*; this is a positive consequence[15]. However, the increase in the solution space also increases the potential evidence options to be reviewed and assessed.

The current approaches which allow confidence to be quantitatively measured do not provide tools to explore the solution space. A tool would allow alternative evidence to be proposed to determine the impact compared to a known baseline, i.e. *what-if* analysis of the potential options. However, current tools lack an ability to *compare* or to *measure differences*. They also lack the ability to ensure that any decisions are based upon *efficient changes*.

4.2.3.13 Existing Approaches Lack an Ability to Inform Decisions Based Upon Multiple Risks

There are a number of factors to consider before gathering any evidence; it is not solely based upon the *availability* of data. There should be the ability to base any evidence gathering decisions on known or perceived risks.

[15]See Figure 1.1.

What could be perceived to be the *right* evidence to gather, based upon confidence values, may not be achievable in *reality*. The practicalities of obtaining evidence must also influence the decisions made. As an example, a Modified Condition/Decision Coverage (MC/DC) testing objective is part of a rigorous software testing regime which can be nearly 40-50% of the system development cost (Ammann and Offutt, 2008). In addition, MC/DC is approximately 40% of the total testing time (Dupay and Leveson, 2000). This provides an indication of the retrospective cost and time to apply such a technique. Having such factors as cost and time included within any intrinsic decision making process would be of value. This would avoid the need to iterate between the tools output and the *reality* of the implementation. In essence, the tool should model the reality of the cost and the time to implement changes to the evidence.

Approach (a) within Figure 4.2 shows the approach adopted by current confidence quantification methods. Any judgements on the cost/time impacts for changes to evidence must be made on the tools *outputs* with *interpretation* needed to compare other evidence strands. Approach (b) within Figure 4.2 shows a method where the confidence of the evidence *intrinsically considers* the cost/time impact of any changes to the evidence. Judgements are needed to consider the cost/time implications but these are recorded within the tool. When presented to the decision maker the confidence in the evidence, and any potential options to alter the evidence, already takes into account the perceived *overhead* to implement any changes. Such a tool in the context of software safety assurance does not currently exist.

4.2.3.14 Myriad of Attributes to Measure Confidence and Diversity within Existing Approaches

The current approaches to measure confidence and to capture diverse evidence arguments adopt a myriad of attributes, such as *relevance*. The attributes and their definitions are inconsistent. Many of the approaches consider the attribute *quality*; this can be to partially measure confidence or to equate to confidence itself, e.g. Nair, Walkinshaw and Kelly (2014) and Denney, Pai and Habli (2011). Other approaches consider *belief* and *plausibility*, e.g. Cyra and Gorski (2008). *Correctness* is another attribute which is captured, e.g. Denney, Pai and Habli (2011). *Trustworthiness* is considered by Nair, Walkinshaw and Kelly (2014) with *completeness* an attribute within Guo (2003). There are many others.

The lack of consistency is to be expected. This is due, in part, to the attributes deemed important to SMEs differing. This illustrates that there is not a clear and defined set of attributes which to measure evidence confidence and diversity.

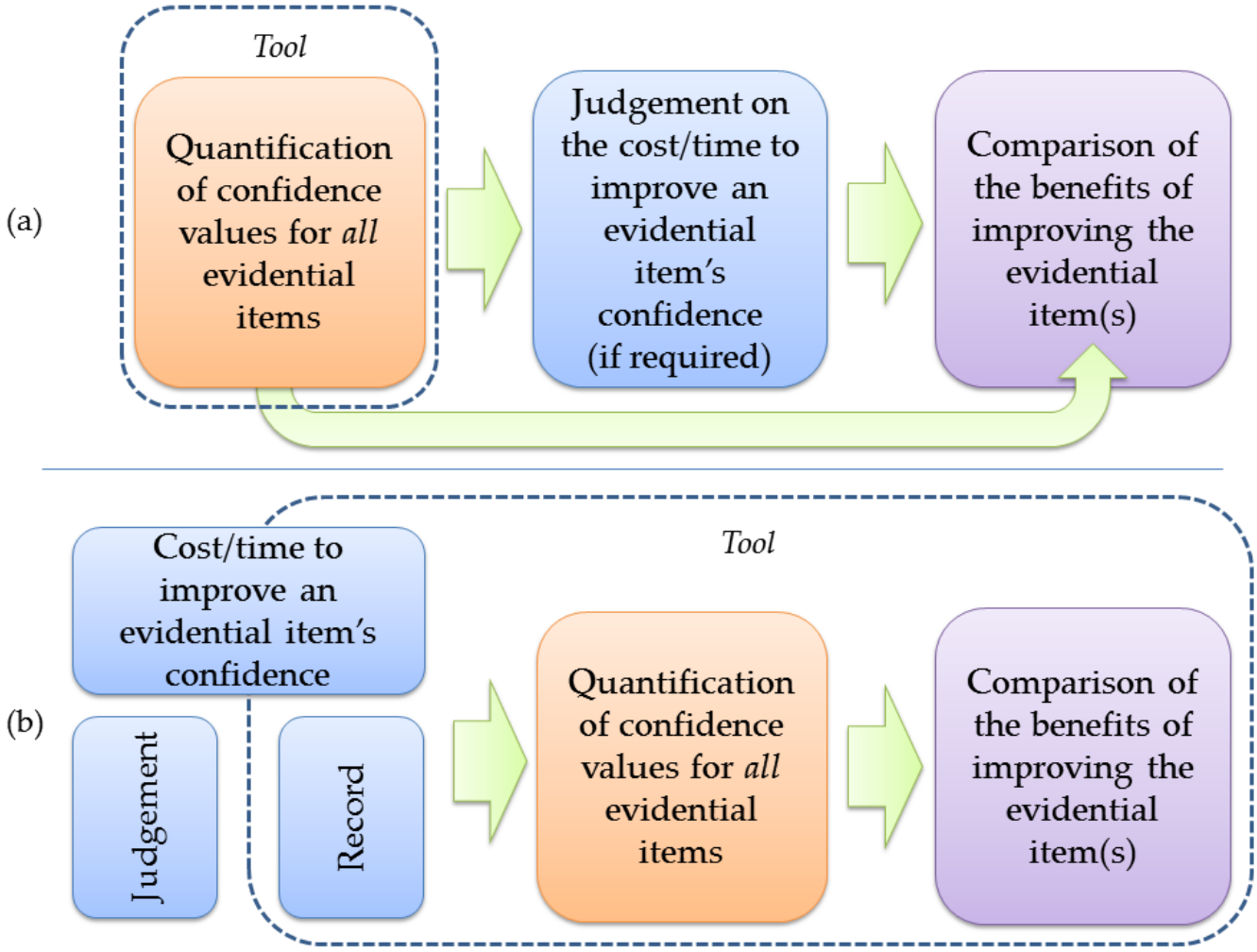


Figure 4.2: Differing Approaches to Consider Cost/Time When Gathering/Generating Evidence

4.2.3.15 Remark on Shortfalls within Existing Approaches

From the review of the closely related research in the fields of diverse evidence and confidence quantification there are clear shortfalls and a lack of support for a clear model or paradigm. There are a number of avenues for further investigation which, if tackled suitably, could enhance the understanding in this domain. This thesis will consider the shortfalls in the current approaches and how they could be resolved by the research.

4.3 Providing Support to Assist Decision Making

Based upon the shortfalls within the existing research there is the potential for a suitable framework to be constructed which allows software safety assurance assessors and suitable stakeholders to:

- Devise arguments using diverse evidence.
- Understand the permutations of such arguments.
- Arrive at a suitable level of confidence.

A key element to this is the ability to assist with the *decisions* being made. Holsapple and Whinston (1996) states that from a management perspective a *decision* is a choice. The concept of *choice* is debated and there are a number of definitions to describe the term:

- Choice is a *course of action*.
- Choice is a *strategy for action*.
- Choice leads to a *certain desired objective*.

Such definitions suggest that the *decision making* process is an act which culminates in the selection of one option from a set of multiple alternative courses of action (Holsapple and Whinston, 1996). A Decision Support System (DSS) is a system that *assists* in such an activity.

A DSS can provide support in reviewing alternatives with a further capability of a DSS being to potentially recommend alternative approaches.

Within decision support theory there are a range of definitions which allow potentially complicated DSSs to be developed[16]. From the perspective of this research a focus could be on providing a capability which assists stakeholders in:

[16]For example, degrees of decision concurrency and organisation design.

- Adopting diverse evidence to support gaining confidence in the software/system of interest.
- Making judgements on the characteristics of the evidence.
- Being provided with options and alternatives to assist with decisions regarding gathering/generating additional evidence.
- Allowing any decisions to be optimised so that suggested options maximise the efficiency and effectiveness of gathering/generating additional evidence.

Such a process could be adopted as part of a DSF. There are various types of decision makers which includes an individual (person or computer) to multi-participant (unilateral or negotiated) (Holsapple and Whinston, 1996). The potential environment for the adoption of such a DSF would be one which could consider individual perspectives or that of a number of stakeholders to gain consensus.

Turban, Sharda and Delen (2010) states that in a business context managers usually make decisions using a four-step process, namely to:

- Define the problem.
- Construct a model that describes the real-world problem.
- Identify possible solutions to the modelled problem.
- Compare, choose, and recommend a potential solution to the problem.

An approach to support such a process does not necessarily have to automate or provide technical solutions to all of the above steps. A framework which assists with such steps, or a partial number of steps, would be of value to the current domain. Likewise, Holsapple and Whinston (1996) states that a DSS can have a number of purposes (see bullet-list below). A valuable outcome of this research would be one which assists with at least one of these concepts.

- Increase decision makers productivity, efficiency, and effectiveness.
- Facilitate one of more of a decision makers abilities.
- Aid one of more of the three decision making phases: intelligence, design, and choice.
- Help the flow of problem-solving episodes proceed more smoothly and rapidly.
- Assist in the making of semi-structured or unstructured decisions.

- Help the decision maker manage knowledge.

The concepts of diverse evidence and the application of suitable attributes for judgements would benefit from a decision support process and as such this research will attempt to deliver a practical and proportionate[17] DSF.

Linked to the concept of assisting with decision support is that of optimising any results/decisions which are made. If there is the ability within a framework for evidence attributes to be reviewed and altered by a decision maker then there is a need to ensure that the values guide the efficiency/effectiveness to gather/generate the evidence. The use of optimisation would allow any decisions to be adopted with a minimal level of additional evidence gathering or refinement. Acknowledging that confidence in the software needs to be improved is a start to gathering/generating evidence but optimisation can determine *what* evidence is required and *how much improvement* is possible.

Approach (a) within Figure 4.3 shows the current methods which requires an iterative application of judgement to understand the evidence to gather/generate. Approach (b) within Figure 4.3 shows the benefits of an optimisation capability within a tool. Such a tool does not currently exist in the context of the software safety assurance domain.

The exact need and the method for such an optimisation process is obviously to be determined as the research progresses. However, from a review of existing research in this area there is a clear opportunity to provide such a facility.

4.4 Summary of the Scope for Further Investigation

For any intervention to be fruitful there is a need for the problem of interest to exist and for the *existing* research in this area to not sufficiently fulfil the identified need. From the analysis which has been conducted within this chapter, it is believed that there is a sufficient *gap* in the software assurance domain to benefit from intervention. This premise is based upon the following:

- Diverse evidence is used extensively within other safety-related domains and diversification is used within software design. Diversity can also be applied to specific evidence to gain confidence in the suitability of the software itself in relation to safety. Diverse evidence can legitimately be seen as a 'good thing'.
- The existing concepts for diverse software evidence and confidence measurement have clear shortfalls. The shortfalls are, in some cases, in the fundamental principles underlying the models and also in the end-to-end process which is assisting decision makers.

[17] *Proportionate* in balancing the theory of evidence diversity and the practicalities of a tool which is of value to decision makers.

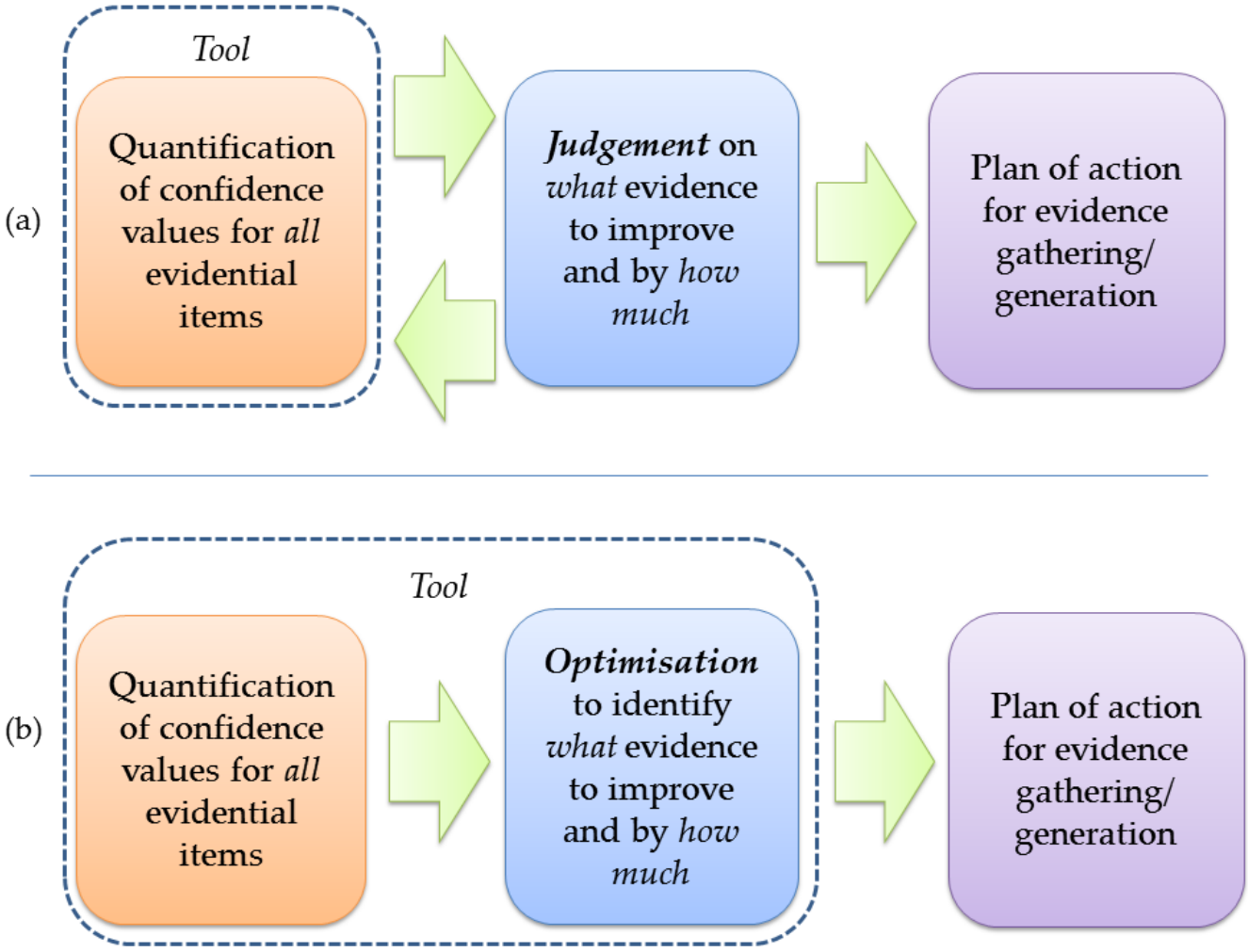


Figure 4.3: Differing Approaches to Determine What Evidence to Gather/Generate

The arguments put forward to date are inconclusive and there is scope to conduct further analysis of diverse arguments. The existing shortfalls which have been discussed in this chapter include:

  - Concepts lack sufficient case studies and adoption of approaches.
  - Concepts acknowledge further analysis is required.
  - Complicated concept to argue and resolve.
  - The need for subjective opinion and expert judgement.
  - Differing applications of the approaches.
  - Differing underpinning principles to the approaches.
  - Inconclusive results on approaches which reason under uncertainty.
  - Minimal existing approaches for visualisation of diverse argument structures and results.
  - Inconsistent (or lacking) information on how method outputs should be utilised.
  - Lack of simplicity to how evidence is captured.
  - Lack of scalability to how evidence is captured/assessed.
  - Lack of ability to perform optimisation and *what-if* analysis.
  - Lack of ability to inform decisions based upon multiple risks.
  - Myriad of attributes to measure confidence and diversity.

- There are DSSs adopted within business and management domains. The research conducted as part of this thesis can deliver value if it results in a revised and more effective approach to capture, judge, and optimise evidence decisions.

Chapter 4 has informed two research sub-questions:

- Sub-section 4.2 has partly responded to the sub-question: *What is the current permissible software safety assurance evidence within the UK defence domain and related domains?*.
- Sub-section 4.3 has partly responded to the sub-question: *What is a suitable structure for software safety assurance evidence and can mathematically derived approaches inform how judgements are made on the evidence and for proposing alternative/optimised solutions?*.

Chapter 5

A Review of the Use of Evidence Within Non-Safety Domains

This section reviews *what* evidence is valid and *how* evidence is used within different domains which do not have a focus on software safety. A verbose discussion on the *philosophy* of evidence, e.g. such topics as the Ravens Paradox[1], is purposefully avoided to concentrate on the *practical* implementations[2]. A focus on the *practical* use of evidence will help to identify lessons to enhance current software safety assurance practice.

This chapter will examine:

- *Evidence: A Definition.* Assessment of a number of definitions applied to the term *evidence*.
- *Evidence: How and Where is it Used.* A review of how evidence underpins decisions within a number of domains.
- *Evidence: A Discussion.* Discussion points from the review of the domains.
- *Lessons for the Problem of Interest.* Lessons which can be applied to the software safety assurance domain.

5.1 Evidence: A Definition

Within a number of domains, e.g. criminal justice and medicine, evidence plays a fundamental role to provide confidence in decisions. How evidence is assessed and adopted varies

[1]What appear to be irrefutable assumptions lead to a consequence that seems intolerable (DiFate, 2017).

[2]Philosophical concepts about evidence can assist with understanding practical perspectives; however, a detailed analysis of such concepts is not germane to the chapters purpose.

between these domains. There are a number of definitions for the term *evidence* which reflects the role it has within the decision making processes:

- Kim (1988) states that the *concept of evidence is inseparable from that of justification* with *evidence making a difference to what one is justified in believing or what it is reasonable for one to believe* (Stanford Encyclopedia of Philosphy, 2014).
- Evidence is a *premise for belief* (Stanford Encyclopedia of Philosphy, 2014).
- A general definition is that evidence is *the available body of facts or information indicating whether a belief or proposition is true or valid* (OED, 2018*d*).
- *Evidence is not proof* as this is gained via deductive or logical reasoning (Weinstock, 2007).
- Evidence is *data on which to base proof* or to *establish a truth or falsehood* (Collins Dictionary, 1995*l*)
- From a legal perspective it is *any matter of fact that a party to a lawsuit offers to prove or disprove an issue in the case* (Lehman and Phelps, 2005).
- Within the UK Government policy-making domain it is the process to *establish and bring together relevant facts, figures, ideas, analysis, and research* (HM Government, 2013).

From the definitions it can be concluded that evidence is a basis to *enable* a premise or a proposition to be believed or justified. There is a requirement to *reason* with, i.e. make sense of, the relevant data or the body of facts. The purpose of the decision influences the *relevance* of its supporting evidence; e.g. evidence can inform decisions which are post-event to *build on a belief* as is the case in a court of law. Evidence can also *create a belief* that a future event will or will not occur, e.g. with trials to learn that a medicine will not cause harm. What makes evidence *good* is based upon the context which it supports.

5.2 Evidence: How and Where is it Used

The following sub-sections review how a number of relevant domains assess and adopt evidence. The review spans law, healthcare/medicine, and policy-making, see Figure 5.1. A number of different domains could have been chosen, e.g. dentistry, teaching etc; however, the domains analysed are sufficient to allow observations and discussion.

Figure 5.1: Non-Safety Domains of Interest

5.2.1 Criminal Justice System (Prosecution)

Within the UK legal system evidence is assessed at a number of stages. This includes at the initial point of evidence collection, e.g. a crime scene, through to the critical evaluation and judgement on the evidence, e.g. within the court of law.

The CPS prosecute criminal cases[3] that have been investigated by the police and other investigative organisations, such as the Department for Work and Pensions (DWP). The CPS: decides which cases should be prosecuted; decides the appropriate charges and provides advice to the police; prepares cases and presents them in court; and provides information, assistance, and support to victims and prosecution witnesses (CPS, 2018*a*).

To proceed with a charge the prosecutors within the CPS must be satisfied that there is sufficient evidence to provide a *realistic prospect of conviction* and that prosecuting is in the *public interest* (CPS, 2018*a*). A Full Code Test is applied by prosecutors which has two stages: (1) Evidential Stage and (2) Public Interest Stage (CPS, 2018*c*)[4]. The defence case is also considered to determine how likely it is to affect the prospect of the CPS obtaining a conviction.

An *objective* assessment of the evidence is required from the prosecutor to determine the realistic prospect of conviction[5]. It is important to note that the test which the CPS applies to the decision to prosecute is not the same as that applied by the criminal courts. The CPS base their test on if a reasonable jury, magistrates, or Judge is *more likely than not to convict of the alleged charge*. A criminal court may only convict if it is *sure that the defendant is*

[3]Within England and Wales.

[4]It should be noted that CPS (2018*c*) is a significant source of the information contained within this sub-section.

[5]How such an assessment can be truly *objective* is questionable given that it is based on the opinions of experts. The issue of ensuring objectivity within decisions is also debated within the field of forensics, e.g. Dror and Cole (2010), with differing contexts resulting in different opinions on the *same* forensic evidence.

guilty.

The *sufficiency* of the evidence is part of the decision to prosecute. An assessment must be made on whether the evidence can be used in court. This is achieved by determining the likelihood that the evidence will be *admissible* and the *importance* of the evidence in relation to the evidence as a whole. The *reliability* of the evidence is considered which includes an assessment on the *accuracy* and *integrity*[6] of the evidence. *Credibility*[7] is a further consideration to establish the sufficiency[8]. Figure 5.2 illustrates the relationships of the attributes to determine evidence *sufficiency*.

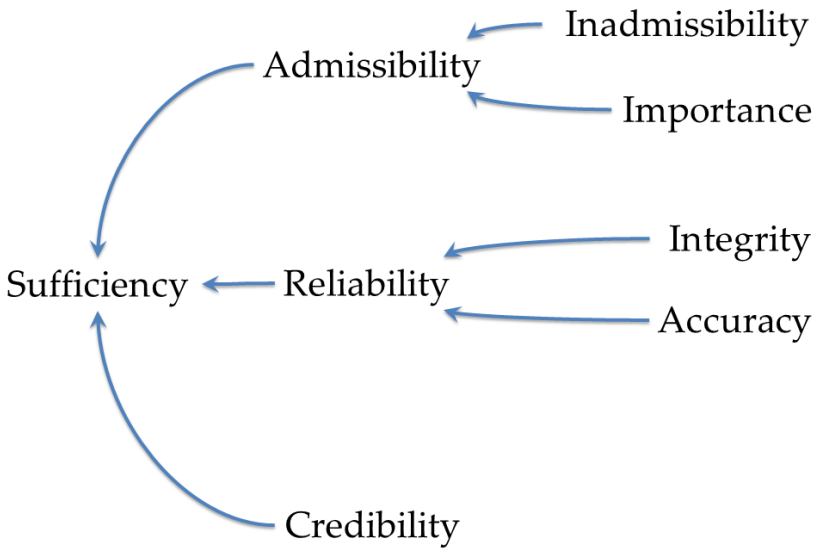


Figure 5.2: Establishing the *Sufficiency* of CPS Evidence (based upon CPS (2018*c*))

Having *sufficient* evidence does not automatically result in a decision to prosecute. There is also a need to assess if the prosecution would be in the *public interest*; this is the second stage of the Full Code Test. To determine if a prosecution passes the public interest test seven questions are considered (CPS, 2018*c*):

1. *Seriousness of the offence?* More serious offences increase the justification for the prosecution.

2. *Level of culpability of the suspect?* Considerations include: the level of involvement, extent of premeditation (if any), and past convictions etc.

[6] *Integrity* relates to having unimpaired judgement or an ethical code, for example.

[7] In this context, the term *credibility* relates to the *believability* of statements made by a witness, for example.

[8] The presentation in a court of law of the three core concepts to criminal evidence (*relevance*, *admissibility*, and *weight* (Hannibal and Mountford, 2016)) will be discussed in the next sub-section (5.2.2).

3. *Circumstances and harm caused to victim?* The vulnerability of the victim is considered and the impact that the event(s) has had on the victim, both physically and mentally.

4. *Suspect under 18 at time of offence?* The justice system implements different procedures for suspects (and victims) who are under 18 years of age. The adverse impact that a prosecution might have on the suspect would also be considered, e.g. future job prospects.

5. *Impact to the community?* If the offence has a more significant impact on the community then the greater likelihood of a prosecution being brought.

6. *Prosecution a proportionate response?* The severity of the alleged offence is a factor when considering a prosecution. Considerations include the cost of bringing the prosecution.

7. *Sources of information require protecting?* The circumstances of a case may mean a decision to prosecute could lead to harming sources of information, international relations, or national security. The risk of causing 'harm' needs to be considered by the prosecutor.

For decisions on domestic abuse charging an *evidence-led prosecution* is adopted. This ensures that a robust prosecution case is not solely reliant on evidence from the victim themselves as this would act as a single 'thread' of evidence. Wider evidence is used to remove the risks of this single 'thread'. Wider evidence includes: *res-gestae* statements[9]; bad-character evidence; understanding circumstances regarding victim reluctance or due to a victim becoming "hostile"; and even hearsay evidence (CPS, 2015*a*).

5.2.1.1 Salient Observations: Criminal Justice System (Prosecution)

The decision to prosecute or not, based upon evidence sufficiency and the public interest, is centred upon the *judgements* of the CPS prosecutor. There are guidelines, case law, and processes to enable a prosecutor to reach an informed decision but the key outcomes are determined via *judgement* and *interpretation*. Any conclusion formed by the prosecutor should be able to be defended to peers and within the court of law. As with many decisions within the safety domain, decisions regarding prosecution are based, in reality, upon *subjective judgements* and these will differ between CPS prosecutors. It could be argued that the aim is for a *defensible* decision to be made not necessarily a consistently repeatable one.

[9]A statement with limited potential for distortion or concoction due to the victim being "emotionally overpowered", e.g. recording of 999 calls.

Another point of interest is the attributes of the *evidence* and how these are judged to allow a conclusion to be reached. The importance of evidence is considered to determine the *admissibility*. Other attributes relate to *reliability* (*accuracy* and *integrity*) and the *credibility* of the evidence. The notion of assigning attributes to evidence and establishing values/relationships has applicability to a software safety judgement. An observation from the CPS public interest test is that it places the evidence within the context of the wider scope of society. This is relevant to a software safety judgement process, e.g. to ensure that the correct evidence and its attributes are considered.

The CPS consider the *cost* of a prosecution when determining how *proportionate* a potential prosecution is. This principle has similarities to the concept of As Low As Reasonably Practicable (ALARP) within the *general* safety domains[10]. ALARP involves weighing a risk against the trouble, time, and *money* needed to control it (HSE, 2018*a*). Cost is a *consideration* rather than a *key driver* for both the ALARP principle and the CPS review of prosecution proportionality. Within the software safety domain the efficiency of the assurance process could be supported by having a range of diverse evidence which offers *equivalent* safety findings.

The CPS also looks towards building a case using different 'threads' of evidence when considering domestic abuse charges. This is to limit the risk of an unsuccessful prosecution due to a single source of evidence, i.e. the domestic abuse victims testimony. There can be preferred sources of evidence but the benefits of having additional strands of evidence is recognised by the CPS process. The additional strands can allow a more robust argument to be achieved. There is clear cross-over to how software safety evidence is gathered.

5.2.2 Criminal Justice System (Court of Law)

The criteria used to establish whether to prosecute a case is different to that used within the court of law. Criminal evidence has three core concepts which are tested within the court of law: *relevance*, *admissibility*, and *weight* (Hannibal and Mountford, 2016). These concepts also form part of the decision to prosecute, as outlined by CPS (2018*c*).

- *Relevance*. For evidence to be put to the court it must be *relevant*. There must be a relationship between the evidence tendered and the fact to be proved; and it must also increase, or diminish, the probability that a fact in issue exists (this is supported by Davis (2018)). However, there are no distinct rules for deciding upon *relevance* and within some literature there is the notion of it being a matter of *common sense* (Hannibal and Mountford, 2016).

[10]For example, the oil and gas industry.

- *Admissibility.* Evidence to be presented to the court also has to be *admissible.* The concept of admissibility is to ensure that the defendant in the case is subject to a fair trial and to ensure that mandatory rules for evidence are adhered to. Evidence may be inadmissible, or become inadmissible, if the evidence is the opinion of a lay witness[11] or unlawful disclosure of evidence protected by legal privilege. Any disputes regarding the admissibility of evidence is decided by the judge or magistrate.

- *Weight.* The third core concept regarding criminal evidence is that of *weight.* Evidence should be convincing or persuasive to a Judge or jury. The type of evidence being presented will influence the weight which is applied to it; e.g. the weight of oral testimony evidence will be dependant on such factors as the demeanour and credibility of the witness. Less weight would be placed on the statements of a witness not perceived as independent. A jury or magistrate may place greater weight on some evidential items more than others, for example Deoxyribonucleic Acid (DNA) evidence might have significant weight (Hannibal and Mountford, 2016).

The *types* of evidence which are presented to the court is also relevant to the problem of interest. Evidence will be presented in one of the following forms (Hannibal and Mountford, 2016):

- *Oral testimony.* Evidence provided orally by a witness. This is deemed the preferred method for evidence to be put forward. Hannibal and Mountford (2016) states that it is likely that this type of evidence may be the most persuasive to a court.

- *Opinion evidence.* A witness should provide factual evidence and not offer an opinion on what was seen/heard. Generally, evidence such as this would be inadmissible. There are exceptions to this: where the opinion is that of an expert witness or where the opinion is of a lay person which does not require expertise.

- *Documentary evidence.* There are a myriad of documentary evidence types: photographs, plans, expert reports etc.

- *Real evidence.* Objects which are produced in court which may allow inferences to be drawn from them, for example Closed-Circuit Television (CCTV) footage of an incident.

- *Direct and circumstantial evidence.* The difference between these two types of evidence is that of the need for *inference.* Direct evidence can be accepted (or rejected) based

[11] 'Layman, (*noun*): amateur, civilian, non-professional, non-specialist, one who has no specialised training, unskilled practitioner, untrained person' (Burton, 2007).

upon the actual evidence presented. Circumstantial evidence is that which alludes to an event and requires an inference to be made by a juror or magistrate.

5.2.2.1 Salient Observations: Criminal Justice System (Court of Law)

The *admissibility* attribute raises a number of points. For evidence to be deemed inadmissible there needs to be an interpretation of guidelines and a judgement on the evidence itself. The judgement is made by the those that "control the proceedings... and decides questions of law or discretion" (Lehman and Phelps, 2005). Within a software safety domain it is feasible to have a number of SMEs with equal experience and knowledge. However, the point of decision will rest with the qualification/certification authority (the MAA in the case of the MOD), with relevant SME input.

The *weight* of evidence is also relevant to a software safety domain. Amongst SMEs there may be interpretations of the importance of evidence based upon their views of the guidelines/standards and beliefs in certain academic theories. This is an important issue when determining the *value* of evidence.

'Types of evidence' are used within a criminal justice system with each having rules and understood interpretations of how the evidence will be judged; e.g. circumstantial evidence needing inference on the part of the jury or magistrate. Within the software safety domain there are also 'types of evidence', e.g. process-based, but there are not the same forms of rules and interpretations to judge the evidence. Due to such features of software safety evidence there is a requirement to capture any decisions made by SMEs to allow a transparent judgement to be recorded.

5.2.3 Criminal Justice System (Expert Witnesses)

The CPS issues guidance on the use of expert witnesses and the evidence which can be put forward in these circumstances. Full guidance can be found within CPS (2015*b*). However, the point to note from CPS (2015*b*) is if there are a range of expert opinions on the matter in question (or value placed upon a type of evidence) then there is a need to understand where the expert's own opinion lies and for the preference(s) to be explained.

In a court of law, where expert witnesses have clear and valid differences in opinion the expert witnesses can be requested to write a joint report/submission. This highlights the agreed matters and to clarify any points which are disputed. There are direct parallels with the use of diverse software safety evidence in that the differing judgements of SMEs need to be understood and to have a mechanism for progress if there are conflicting views. It would be of value for a method to capture a consensus of SMEs judgements and to compare any disputed opinions.

5.2.4 Healthcare and Medicine

The term Evidence-Based Medicine (EBM) became popular in the early 1990s[12]. EBM is a process to adopt the "conscientious, explicit, and judicious use of current best evidence in making decisions about the care of individual patients" (Sackett et al., 1996).

EBM integrates the clinical expertise of individuals with external clinical evidence via systematic research. The *expertise* of the individual clinician is formed from their proficiency and judgements. *External clinical evidence* is research which is patient centred. This external evidence should be able to invalidate previously accepted patient treatments and have the ability to replace the treatments with more powerful and more accurate ones (Sackett et al., 1996).

Importantly, Sackett et al. (1996) states that EBM is not a *cookbook* as it requires a bottom up approach that integrates external evidence with individual clinical expertise. There cannot be a *slavish* cookbook approach to care.

This mix of clinical expertise and clinical evidence has an ordering which is informed by the significance and weight assigned to the evidence. Howick (2013) states a simplified hierarchy of evidence, shown in Figure 5.3. The hierarchy is based upon three central claims:

- Randomised Controlled Trials (RCTs) or systematic reviews of many randomised trials are stated to provide stronger support than observational studies[13].
- Comparative clinical studies (including RCTs) and observational studies offer stronger support than mechanistic-reasoning[14].
- Comparative studies in general offer stronger evidence than expert clinical judgement.

An expansion to the scope of EBM is the use of evidence within general science. It is useful to include this expanded definition here due to the overlap in the evidence of EBM. There are many references which include information on scientific evidential types and their potential hierarchies, e.g. NRC (2011); Cwik and North (2003); Parkhurst (2016); Perry, Potter and Ostendorf (2015); Hoffmann, Bennett and Del Mar (2013); and Imwinkelried (2014). A succinct illustration of a number of evidential types used within general science is shown within Compound Interest (2015). Figure 5.4 shows the evidential types which are particularly relevant to healthcare and medicine.

[12]Although whether it was a *new* paradigm is debated (Howick, 2013) with theories that the philosophical origins extend back to the mid-19th century (Sackett et al., 1996). This debate is outside of the scope of this thesis.

[13]Within observational studies inferences are derived from a sample to a population in which there is less control of the independent variable.

[14]This is where inferences are made from mechanisms to the claims that an intervention produces a patient-relevant outcome (Howick, 2013).

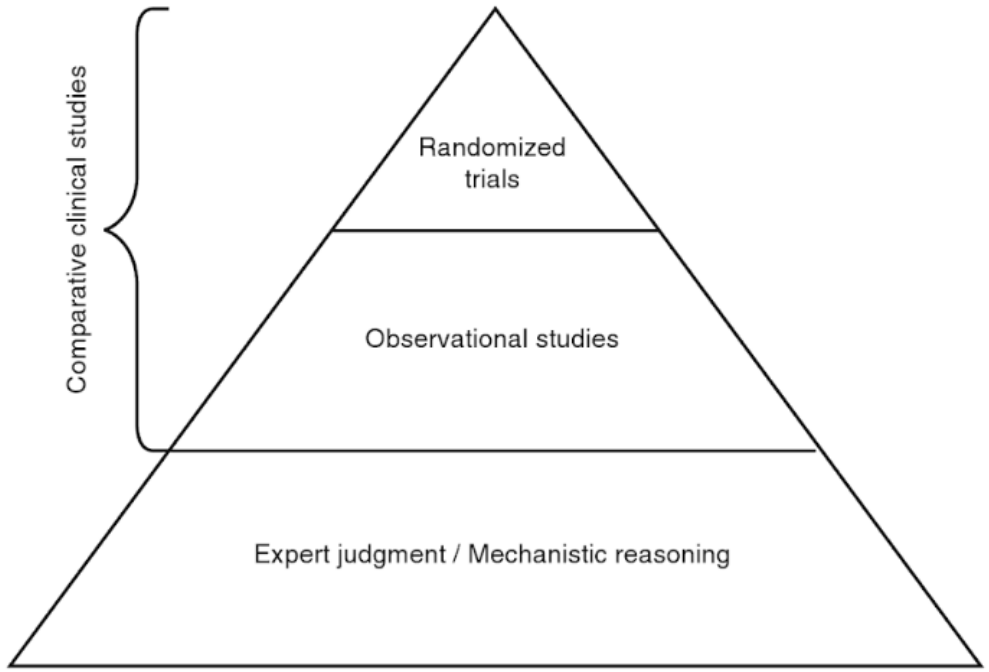


Figure 5.3: Simplified EBM Hierarchy of Evidence (Howick, 2013)

The University of Oxford Centre for Evidence-Based Medicine (CEBM) levels of evidence provides supporting metrics and quantitative information for the number of studies to be conducted for each of the evidential types within Howick et al. (2011). Howick et al. (2011) states that the evidence hierarchy and the quantity of the studies conducted is *not* intended to provide a definitive judgement on the quality of evidence. This is because lower-level evidence may be stronger than higher-level evidence which, for example, provides inconclusive results. NICE (2017*b*) allows clinical decisions to be structured based upon: guidance and policy (e.g. safety alerts and quality indicators); secondary evidence (e.g. systematic reviews and economic evaluations); primary research; ongoing trials; current awareness of medicines; practice based information; implementation; and patient decision aids.

NICE (2017*b*) advocates the use of patient decision aids to help determine medical treatments. The British Medical Journal (BMJ) reports that a patient decision aid should: improve knowledge of the options and help patients reach choices that are more consistent with their informed values (BMJ, 2013). A decision aid is defined as an approach to: describe the decision to be taken; the options available; and the outcomes of these options (including benefits, harms, and uncertainties) based on a careful review of the evidence (BMJ, 2013).

A remit of National Institute for Health and Care Excellence (NICE) is to assess the benefits of introducing medical interventions, e.g. a drug or treatment, which could improve the prognosis of a patient's condition. The assessment of these benefits is evidence-based with

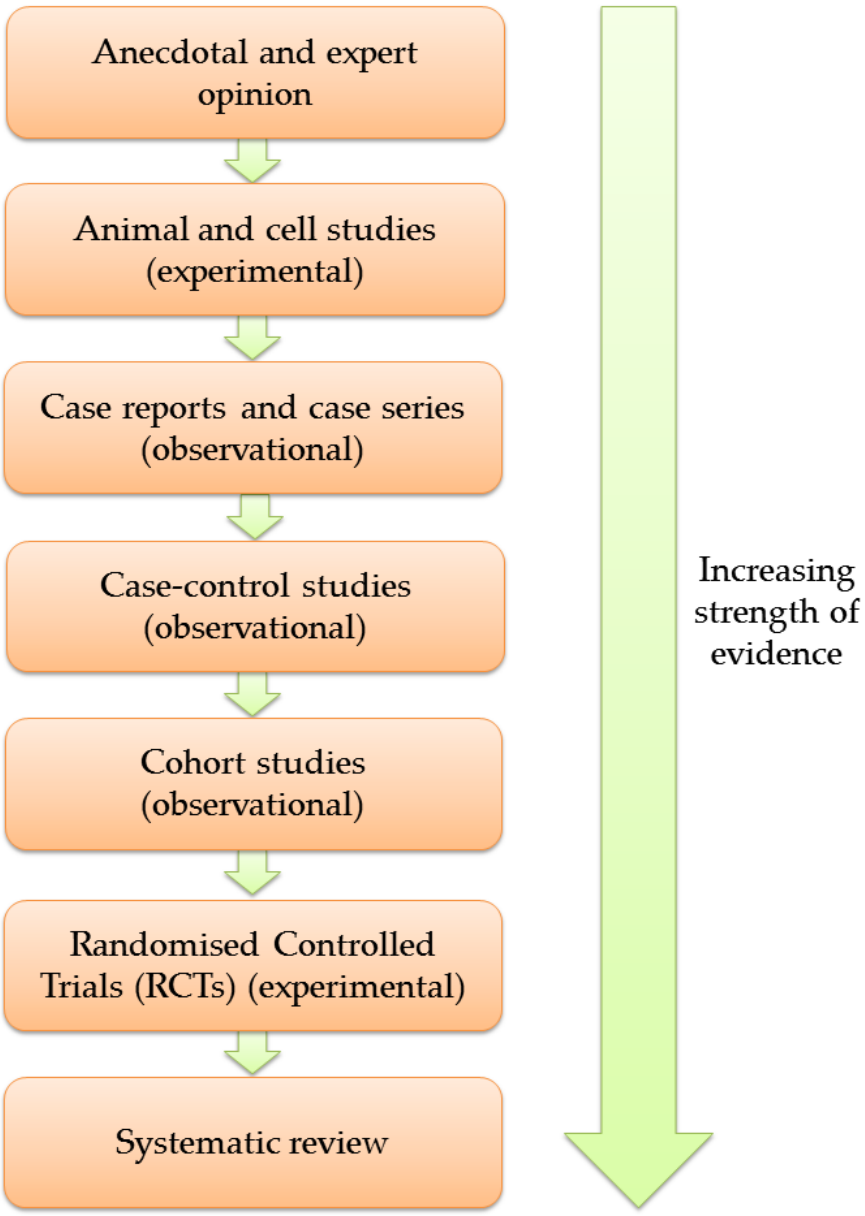


Figure 5.4: Common Evidential Types Relevant to Healthcare/Medicine (adapted from Compound Interest (2015))

a range of factors informing the decision making process. Factors include, the effectiveness of the intervention and any short- and long-term costs (NICE, 2017*a*). Other interventions are considered if they are more effective and/or efficient.

5.2.4.1 Salient Observations: Healthcare and Medicine

It is clear that the concept of EBM and the underpinning philosophy prior to the adoption of EBM is based upon a balanced appraisal of the evidence. The appraisal involves clinical expert judgement, in essence SME judgement, with scientific evidence underpinned via research. Wider than EBM is the more general adoption of evidence within science and there are clear rankings which are associated with the evidence presented as part of a clinical statement. There appears to be a domain-wide understanding of the types of evidence which need to be gathered to arrive at a conclusion which can be defended to peers. Also, the relative strengths of each form of evidence appears to be understood.

These observations are relevant to the software safety domain as even with a 'preferred' hierarchy of evidential types the evidence still needs to be judged on the context of the results and the support that the evidence provides.

Lessons can be learnt from the use of patient decision aids by the National Health Service (NHS). Providing stakeholders with information in a format which assists their knowledge of the options and for choices to be made which are consistent with their values, or *beliefs*, is a concept which has value to the software safety domain. The use of *visual* aids for decision making is recommended by BMJ (2013) and this concept could also be applied to the software safety domain.

Sackett et al. (1996) claims that there should not be a *slavish cookbook* approach to care within the EBM domain. Sackett et al. (1996) is advocating using evidence based upon its merits and the value that the evidence has towards reaching a justified conclusion. This concept has relevance to the software safety domain as an assurance claim can be supported by a range of relevant diverse evidence which is not focussed on a particular form of evidence.

The EBM domain acknowledges that there is a necessity to integrate individual clinical expertise with external clinical evidence to inform a medical treatment. There could be scope within the software safety domain for expertise to drive the choice of external evidence to inform decisions based upon relevance and value.

NICE and the NHS conduct budget impact tests to assess the affordability of introducing new drugs (NICE, 2017*a*). Considering *cost* is a necessity to ensure that any interventions are as efficient as possible. Within other domains, the decision making process may not necessarily include the cost of gathering diverse evidence as a key driver. However, the *efficiency* of gathering any evidence could act as a differentiator when selecting diverse

evidence when there are *equivalent* safety findings.

5.2.5 Government Policy Strategy

The concept of decisions being 'evidence-based' is one which became prevalent in UK politics during the 2000s through the Labour Government's stated commitment to Evidence-Based Policy Making (EBPM) (Rutter, 2012)[15]. This stated commitment has been subsequently supported by successive Governments and was referenced within the 2012 Civil Service Reform Plan with a stated requirement to be "building on evidence that works" (HM Government, 2012).

There are various stages at which evidence-based decisions should be made. Sense about Science (2018)[16] evaluated the Government's use of evidence from the following stages:

- *Diagnosis.* The issue that will be addressed.
- *Proposal.* The Government's chosen intervention.
- *Implementation.* How the implementation will be introduced and run.
- *Testing and Evaluation.* Assessment of if the policy has worked, or, in the case of consultations and further investigations, how the information gathered will be used.

The evaluation of the numerous Government agencies scrutinised for the Sense about Science (2018) study shows mixed results for each stage and for each Government Department. At present the UK Government is most transparent about the evidence which is adopted for the *diagnosis* stage of policy-making. The Government is least transparent on how it plans to conduct the *test and evaluation* of the policy (Sense about Science (2018) and Sense about Science (2017)).

The use of evidence to underpin advice for policy making is also part of a Government Office for Science (GO-Science) paper on "The Government Chief Scientific Advisor's Guidelines on the Use of Scientific and Engineering Advice in Policy Making" (GO-Science, 2010). GO-Science (2010) advocates the use of a range of evidence sources to support the advice which is provided. This includes using appropriate *expert* sources and recognising that SMEs will have differing viewpoints. GO-Science (2010) contains guidance that risk and uncertainties should be stated clearly. This is an acknowledgment of the limitations of certain evidence.

[15] Although it is recognised that an evidence-based approach predates this period (Panjwani, 2017).

[16] Sense about Science (2018) is a study which evaluated how evidence informed policy adoption for a number of areas within Government.

Panjwani (2017) states a number of challenges with EBPM which may impact its adoption or continued use within Government:

- *Actors in the policy making process.* Agreed terms of engagement are required due to the complexity created from policy making involving individuals/organisations with different incentives etc.
- *Types of evidence.* Scientific, economic, social, and cultural evidence may form part of a policy decision. Such evidence may take many forms, e.g. peer reviewed papers, user experience, SMEs. The mix and types of evidence is varied within Government.
- *Other factors.* Values, experience/judgement, information gaps, secrecy, need for expediency, funds, and timings etc are all factors which are part of the policy making process in addition to the evidence itself.
- *Matching supply and demand.* There needs to be a balance between the supply and the demand for evidence. To form part of a solution to a problem, the research must be balanced by *quality*, *credibility*, and *relevance*.

Within the UKs central and local Government the standards for applying evidence are adopted inconsistently. Some areas of policy making are more adept in making judgements on underpinning evidence; e.g. for UK social policy decision making there are a number of 'standards of evidence' used. Puttick (2018) analysed 18 standards of evidence which UK organisations use for *judging evidence*. Examples include: The Confidence Review (Catch-22, 2018), Evidence Principles (Bond, 2018), Standards of Evidence (Project Oracle, 2018), and the Teaching and Learning Toolkit (Education Endowment Foundation, 2018).

There are calls for EBPM to be more in keeping with EBM in terms of an agreed hierarchy of evidence. However, Cairney (2017) states that a more reasoned solution is to understand *how* and *why* policy makers demand information and to understand the complexity of the operating environment.

5.2.5.1 Salient Observations: Government Policy Strategy

Not all UK Government policy domains which implement EBPM use a hierarchical structure for assessing evidence. There is recognition that there are disparate sources of information, e.g. Bond (2018), but the EBPM domain in general has resisted applying preferences to certain types of evidence. This method is a valid approach but within EBPM there is a need to fully understand the *context* and the *limitations* to any evidence used to inform a decision or argument. For the software safety domain this is useful as it supports the validity of a

non-hierarchical evidence structure for decision making. The uncertainty regarding evidence could be captured via suitable attributes which *describe* characteristics of the evidence.

There is an acknowledgement within the EBPM domain that user experiences and SME judgements will inform the methods which are adopted for a decision making process. This recognition is two-fold: for the SME making a judgement they need to be cognisant of other SMEs valid perspectives; and that where SME judgements inform a decision there may be a level of bias, most likely unintentional, towards a certain process/outcome. Within the software safety domain there may be a need to allow the decisions of stakeholders to be formed via *consensus*, as can be the case with EBPM.

Other factors such as the time for the implementation and the available funds should form part of a decision making process. These may influence the success of a policies outcome. Decision making processes should be shaped by the wider influences/dependencies of the evidence, e.g. an *overhead* associated with gathering the evidence.

Panjwani (2017) cites attributes (*quality*, *credibility*, and *relevance*) to be considered for evidence and decisions which inform a solution. The use of such evidence attributes which have pedigree within other domains could assist a potential software safety decision making process.

5.3 Evidence: A Discussion

5.3.1 Need for Evidence-Based Decisions

For policy-making there is a need for decisions to be transparent and to be justified when subjected to the scrutiny of stakeholders, such as the media or the public. In addition, there is the need to do *more with less* due to the reduced funding which many policy-making organisations have when compared to the past. Policy-makers cannot afford to get it wrong (Nutley, Powell and Davies, 2013), although this is also the case for many (if not all) domains.

Evidence is also required to ensure the *effectiveness* of decisions, policy-based or otherwise. This is linked to managing perceptions and needing to have transparency but, more importantly, the need to *arrive at a legitimate decision* which is in the best interests of achieving a solution.

5.3.2 Understanding the Context

Understanding *context* is important for decision making and for the use of evidence. Decisions can be informed by evidence which is based upon 'past' events. In such cases, there may be a lessened degree of *inference* to make a decision as the evidence can potentially be

directly linked to an event. This is the case with criminal law[17]. Decisions which are based upon a *prediction* of a forthcoming event (e.g. clinical trial) or the *belief in a wider hypothesis* (e.g. the extensive release of a medical treatment) are of differing contexts. Therefore the decisions can be legitimately based upon different evidence and different judgements on the comparable evidence. A software assurance judgement on development/process evidence is one which is based upon a *prediction* (i.e. establishing a *prior* belief) as with clinical trials. An assurance judgement based upon in-service evidence is one based upon 'past events', as with criminal law[18].

Another consideration is that decision makers should review evidence "in the round" (Nutley, Powell and Davies, 2013). Evidence may support the judgements to form a conclusion; however an evidence-based process needs to consider the wider characteristics of the evidence. Considerations include cost and acceptability (Nutley, Powell and Davies, 2013).

5.3.3 What is 'Good' Evidence?

What counts as good evidence will vary considerably given the context (Nutley, Powell and Davies, 2013). The *quality* of the evidence, the rigour associated with capturing the evidence, and the subsequent judgements on the evidence will differ depending on "what we want to know, why we want to know it and how we envisage that evidence being used" (Nutley, Powell and Davies, 2013). Depending on this context then the concept of *quality* and the *acceptability* of the methods can be agreed upon with relevant stakeholders. For certain decisions and problem domains an overall consensus would be unreachable; however, for a problem domain which has a modest and manageable number of stakeholders then a consensus could be agreed.

Is there a need for the evidence to be *compelling* or just *good enough*? Does there need to be multiple sources of information/data to arrive at a justified decision? What weight and judgements can be applied by those applying the recommendations? Context will dictate. Project Oracle (2018) includes the concept of using criteria to determine the level of supporting research/evidence depending on whether the decision needs to be the *best* or only *good enough*. In essence, evidence which is deemed *good enough* should be fully sufficient to make the argument that a particular claim is true to a certain level of confidence. However, for decisions on software *safety* it would be difficult to defend the use of evidence which was only *good enough*.

[17]Not including particular nuances such as exceptions for the right to bail which are based upon a person being be denied bail if there are substantial grounds for *believing* that any of the exceptions in *Schedule 1 of the Bail Act 1976* are made out (CPS, 2018*b*). The *belief* is, in essence, based upon a *prediction*.

[18]Although in-service data is used to predict that the confidence can continue in future based upon the same context of use.

5.3.4 Hierarchies of Evidence?

As highlighted, there are a number of 'hierarchies of evidence' adopted in a number of domains[19]. The hierarchies determine the standard of evidence in support of a particular outcome. Within the science domain there are commonalities with the various hierarchies as randomised experiments with clearly defined controls (RCTs) have the highest preference. Case studies/reports usually with the least preference (Nutley, Powell and Davies, 2013). However, there are issues with the adoption of hierarchies. Hierarchies can neglect too many important and relevant issues around evidence and they can exclude all but the highest-ranking evidence. This can lead to a loss of useful and relevant information (Nutley, Powell and Davies, 2013). The value of the evidence synthesis can be weakened with such hierarchies (Ogilvie et al., 2005). Using evidence hierarchies as a technical filter prior to research synthesis is *wasteful* and can lead to *misleading conclusions* (Pawson, 2003).

Even 'accepted' evidence hierarchies are subject to debate with suggestions for wider evidence to be included (Bagshaw and Bellomo, 2008); e.g. the debate on if further evidence such as biological plausibility should be included within Grading of Recommendations Assessment, Development and Evaluation (GRADE). Matrices of evidence are seen as a way forward within some domains; however, there are conflicting views about the merits of the different forms of evidence (Nutley, Powell and Davies, 2013).

5.4 Lessons for the Problem of Interest

A number of lessons have been stated in the *salient observations* which accompany each subsection in this chapter. This section draws upon these observations to provide a high-level view of the lessons for the software safety domain.

The definition of *evidence* is not straightforward. There are a number of philosophical elements that are of interest, such as the concept of evidence being a *premise for belief* and evidence *not being proof*. Any concept which provides the ability for evidence to be gathered is, in essence, supporting *beliefs* to be formed via *reasoning* which allows *proof*[20]. The context is key to ensure that the types of evidence adopted and the judgements that are captured via relevant attributes are proportional and adequate for the problem.

Within a number of domains the processes used to gather and form the basis for decisions is underpinned by the *judgement* of the decision makers. Hierarchies of evidence and standards of quality for evidence exist to allow a decision, or judgement, to be made

[19]See Bagshaw and Bellomo (2008) and Petticrew and Roberts (2003) for a study design for the medical domain.

[20]The term *proof* is used in the context of the establishment of a fact by the use of evidence (Lehman and Phelps, 2005) rather than an inferential argument for a mathematical statement (Cupillari, 2012).

using defensible approaches. However, the key principle to this process is that *judgements* are being formed. This is the case for domains which appear to apply 'robust' evidential approaches, e.g. EBM, and the strategic policy-making domain. The concept of 'hierarchies of evidence' clearly has it's merits but there are issues to consider, e.g. wasteful use of data Pawson (2003).

The attributes associated with evidence allows it to be judged. This means that the *belief* of a proposition being true is based upon the understanding of the evidence itself. Within the CPS there are the concepts of *reliability*, *accuracy*, and *integrity* (CPS, 2018*c*). Policy-making refers to attributes such as *quality*, *credibility*, and *relevance* (Panjwani, 2017). These examples illustrate that evidence attributes can play a fundamental role in understanding the *value* of evidence and also as a mechanism to *measure* evidence via judgements. There are various forms of evidence, e.g. documentary and real (CPS, 2018*c*)[21], with each having levels of *weight*. This concept is worthy of consideration for any judgments on software safety.

Evidence needs to be considered on the basis of what would lead to a successful outcome. Relevant factors include implementation cost and risk. Cost is a key factor within the NICE and the NHS decisions on interventions but other domains use cost as a *consideration* rather than a *key driver*. These factors are not always part of the decisions which determine a technical solution; however, evidence which is the most *suitable* is not always the most *obtainable*.

The NHS advocates the use of patient decision aids to provide information in a format which assists their knowledge of the options and for choices to be made which are consistent with their values, or *beliefs*. There are clear advantages to adopting such an approach within the software safety domain. Visualisations and models can allow stakeholders to *comprehend* the range of options which diverse evidence can provide.

Caution is needed when comparing the domains which have formal evidence-based decisions due to the differences in the use of evidence and the contexts. An example is with the acceptance of evidence within the justice domain as the principle of case law[22] has cross-over to the software safety domain. There is an acceptance of *precedence* within the legal and safety domains. Evidence or techniques from *previous* safety assurance arguments can be put forward for the *current* arguments. An example, is a claim that reliability models may not be explicit objectives within extant standards but they may still have had acceptance as part of a prior software safety assurance argument. This *acceptance* sets a precedent. However, within a legal domain the case law acts as the *confirmed* precedence which other future judgements *must* consider. Therefore it has a greater degree of weight. This could

[21] See sub-section 5.2.1 for further information.

[22] Case law is a past ruling which can be cited as a precedent (Apple and Deyling, 2012).

be analogous to process-based evidence within the software safety domain as process-based evidence is the *confirmed* precedence and therefore has greater weight than evidence with *perceived* precedence, e.g. reliability models. However, with the weight of process-based evidence not always considered as the strongest form of evidence, e.g. as stated by Menon, Hawkins and McDermid (2009*b*), there may be scope to increase the weight of other evidence (which has precedence or not).

Within the safety domain there should be varying *degrees* of conformance to a particular standard or objective. However, within the legal system the outcome is based upon a binary decision, i.e. guilty or not guilty, but the sentencing can reflect the degrees of culpability.

The review of a number of non-software safety domains has provided valuable observations which can be considered for any enhancements to the software safety domain. This ranges from the types of evidence, how such evidence can be measured, the non-technical factors which could differentiate evidence, and the value in allowing stakeholders to visualise data to make informed decisions.

Chapter 5 sub-sections 5.2 and 5.3 have partly responded to the research sub-question: *What is the current permissible software safety assurance evidence within the UK defence domain and related domains?*.

Chapter 6

Current Permissible Evidence for Safety-Critical Software Assurance

Chapters 4 and 5 assessed a number of non-software domains to ascertain how evidence can be used to form judgements. The use of evidence for decision making is complicated[1] with a number of approaches adopted to manage the challenges of evidence assessment, e.g. establishing the *relevance* of evidence.

This chapter will examine:

- *MOD Software Assurance.* It is important to understand the evidence which is currently adopted for software assurance arguments within the MOD.
- *Software Assurance Within Other Safety-Critical Domains.* There are lessons to be learnt from the observations made on other safety-critical domains. Figure 6.1 shows the safety-critical domains of relevance to understand the potential permissible evidence.
- *Literature to Inform Software Assurance Evidence.* In addition to the relevant observations from the domains there is also literature and guidelines which can inform the permissible evidence to inform a safety argument.

The information related to *airborne* software and CEH assurance will undergo a greater level of analysis by the RE in comparison to other domains. This is deliberate as the focus of the thesis is on enhancements to *software/CEH* airborne assurance arguments.

[1]With the definition of *complicated problems* being those which are "hard to solve but they are addressable with rules and recipes" (Kinni, 2017).

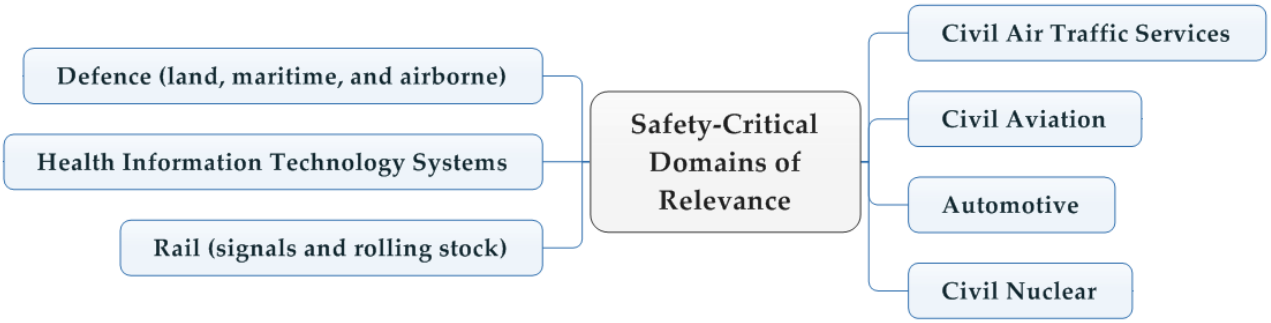


Figure 6.1: Safety-Critical Domains of Interest

6.1 MOD Software Assurance

A high-level overview of the MOD safety assurance process was described in the previous chapter. This section will look at how the current *software* assurance processes deal with evidence and how *diverse* evidence is currently captured and managed.

6.1.1 Airborne Platform Software Assurance

DS 00-970 (UK MOD, 2014*a*), specifically requirement 1.7 - Safety Related Programmable Elements, contains considerations for determining the airworthiness of PEs. However, it is recognised via the MOD Military Certification Review Item (MCRI) process that rigid considerations, as part of the DS 00-970 default airworthiness code, are not always applied (MAA, 2017*d*).

Within DS 00-970 the PE considerations relate to:

- System-level safety considerations.
- Airworthiness related cyber security assurance.
- SRS assurance.
- Safety-related CEH assurance.

For this thesis the *system-level safety considerations* and *airworthiness related cyber security assurance* will not be reviewed as, although important considerations for safety assurance, they are out of scope of the *immediate* research focus on software/CEH. However, any diverse evidence approach should be able to take the assurance of such aspects into account when measuring the overall safety assurance confidence.

6.1.1.1 Safety Related Software (SRS) and Safety Related Complex Electronic Hardware (CEH) Assurance

The AMC for SRS is DO-178C (RTCA, 2011*a*) which has a focus on life-cycle process. It can be argued that the process-based stages lack variety in terms of the assessment methods and hence the level of evidence diversity[2].

DO-178C contains details on additional considerations to life-cycle evidence. This includes a reference to software reliability models and PSH. However, the reference to software reliability models within DO-178C is only included within the guideline to state that such models *do not* provide results in which confidence can be placed. This position on reliability modelling is arguable for some models and applications and is worthy of debate in itself; however, the arguments will not be discussed further within this thesis.

Positively, DO-178C supports the use of PSH as it indicates that it is a viable form of evidence (which, interestingly and as contradiction, is a form of reliability model). DO-178C states that it is possible for an equivalent level of software safety to be demonstrated by the use of PSH. There are a number of dependencies which need to be met for the method to be accepted (e.g. configuration management of the software); however, there is a clear statement of support for PSH.

It is not clearly stated within DO-178C what *level* of credit that can be claimed via PSH, i.e. partial or full compliance. DO-178C states that "some certification credit may be granted" which indicates that full credit may not be gained. However, DO-178C also indicates that any use of PSH should be included in the Plan for Software Aspects of Certification (PSAC). DO-178C goes on to direct that any inclusion within the PSAC should state which objectives in sections 4 to 9[3] of DO-178C are to be addressed through the use of PSH. This indicates that all objectives are valid for a PSH argument if an equivalent level of confidence can be demonstrated. However, the guidance is not definitive in relation to the use of PSH and is very much open to interpretation.

Discussion Paper #4 within DO-248C (RTCA, 2011*b*) contains additional guidance on the DO-178C terminology and the intent of PSH, i.e. the service history rationale for DO-178C. However, within the guidance it is not clear if PSH can be used fully in-lieu of processed-based evidence. However, there should be a recognition that there are overlaps in the evidential requirements for PSH and process-based assessment, mainly CM.

DO-254 (RTCA, 2000) refers to the use of PSE[4], stating that service experience may

[2]This is based upon the premise which was determined in Chapter 3 (and throughout this thesis) that diversity is a *good thing* (Littlewood and Wright, 2007).

[3]Software planning process, software development process, software verification process, software Configuration Management (CM) process, software Quality Assurance (QA) process, and certification liaison process.

[4]Section 11.3 of RTCA (2000).

be used to substantiate design assurance for previously developed hardware and for COTS components. The use of any PSE in the context of DO-254, should:

- Be assessed against a number of acceptability criteria, e.g. actual failure rates in operation.
- Make an assessment of the PSE data to satisfy the criteria and to also meet the PSE data requirements.

As with DO-178C this indicates that there is an acceptance that PSH can be a valid form of evidence in-lieu of process-based evidence. There are a number of criteria which must be met to validate any PSH data. It should be noted that although DO-254 is supportive of the use of PSH the section which defines the use of PSH is far shorter and less comprehensive than DO-178C. SMEs in the field of CEH assurance[5] have stated that PSH is actively used for CEH assurance; however the level of service data required to provide equivalent confidence in-lieu of design information is very subjective. Also, there can be belief gained in the PSE of the *component* itself and of the component's *design* (Fulton, 2017). There is a clear preference within the MOD software and CEH assurance regulations for judgements to be based upon the use of process-based evidence.

It is clear that in two of the most recognised guidelines for software and CEH assurance, i.e. DO-178C and DO-254 respectively, there is support for the use of PSH. PSH can form an equivalent safety argument if certain criteria are met for the supporting data. However, the use of DO-178C and DO-254 by the regulatory authorities is to determine the *process-based* life-cycle objectives and not for the use of PSH. This is also the case for DS 00-55 which outlines a number of requirements and objectives with references to RGP and open standards. These are referenced in the context of the *process-based* life-cycle guidance.

PSH evidence is seen as a method to be employed if the preferred, process-based, evidence cannot be gathered. However, there is debate to the value of process-based evidence; e.g. Menon, Hawkins and McDermid (2009*b*) states that "there is no evidence that a good process will result in a good product (although there is a correlation between bad processes and bad products!)".

CAST-1 (CAST, 1998)[6] provides information on the attributes which could be assessed to gain a level of assurance confidence for a system based on PSH. The attributes are focussed on broad categories which link to DO-178B (RTCA, 1992). The categories are (CAST, 1998):

[5]For example, Fulton (2017).

[6]CAST Position Paper (CAST-1) - Guidance for Assessing the Software Aspects of Product Service History of Airborne Systems and Equipment.

- Means of compliance.
- Service history duration.
- Product quality.
- Problem detection and reporting.
- Modifications and control.
- Evidence of compliance.

A number of attributes are considered for a PSH argument. CAST-1 provides attributes for consideration such as: service duration length, change control during service, and error detection/reporting capability. It is expected that these attributes will be judged, using linguistic terms, to determine the acceptable level of the PSH evidence. The judgements are based upon *scales* of acceptance. An example is within Table 6.1.

PSH Attribute	**Scale**
Service duration length	*Short* ↕ *Moderate* ↕ *Long*

Table 6.1: PSH Attribute and Scale (CAST, 1998)

The individual attributes are judged by a level of *acceptability* to provide credit towards the software assurance confidence. This is shown in Table 6.2. Once these judgements are made then CAST-1 provides a matrix to judge if the evidence associated with an attribute can be linked to a DAL.

PSH Attribute Acceptability
Credit allowed
↕
Credit allowed based on engineering judgement
↕
Engineering judgement for no or some credit allowed
↕
Little if any credit allowed
↕
No credit allowed

Table 6.2: PSH Attribute and the Acceptability Scale (CAST, 1998)

The CAST-1 concept is very much subjective, e.g. what is a *long* level of service duration?, and is open to conflict between stakeholders. However, there are benefits to the approach as it allows a commonality of language and a method for stakeholder dialogue. The use of PSH via the CAST-1 method indicates that PSH can be used as a valid approach to gain assurance confidence and not necessarily in-lieu of process-based evidence.

The MAA undertake reviews to ensure that DTs seek *counter-evidence*[7] as part of their safety argument. Evidence that the DTs have sought counter-evidence is just as important as the counter-evidence itself (as stated in the interview with the MAA)[8].

Also stated within the interview with the MAA is that there needs to be an underpinning philosophy to the standards that are applied by a DT for the software safety argument. Some standards have different approaches and principles but the terminology used within a safety argument has to be consistent and in keeping with the standards that are being applied. It would not be possible to take the definition of a Software Integrity Level (SIL), for example, from one standard and apply it using a differing context. In essence, DTs cannot "*cherry pick*" from standards as there is a need to retain the context to the original intent of the objectives (as stated within the MAA interview).

The MAA allows the clearances of aircraft which may be subject to limited evidence which has not been derived from a fully substantiated safety assessment (MAA, 2016*b*)[9]. These clearances are termed Clearances with Limited Evidence (CLE). In addition, the MAA also has the concept of Operational Emergency Clearances (OEC) for when the equipment does not satisfy the project safety standards. These will be subject to Special Conditions whilst ensuring that full clearances are granted as soon as possible (MAA, 2016*b*)[10].

6.1.1.2 Military Aviation Authority (MAA) Design Approved Organization Scheme (DAOS)

RA 1005 (MAA, 2018*d*)[11] is focused on providing guidance to DTs for contracting with competent organisations. The RA is relevant for organisations within the Defence Air Environment (DAE) who may be contracted to conduct design, maintenance, contractor flying, or air traffic management activities. RA 1005 is clear that without competent organisations contracted to conduct activities there may be a *compromised* level of air safety.

The MAA have a number of approval schemes to establish the competencies of organi-

[7] Counter-evidence refers to the provision of an item of evidence which has the potential to undermine a claim (Menon, Hawkins and McDermid, 2009*a*).

[8] A list of the research interviews are contained in Appendix B.

[9] RA 1300 - Release to Service. For further information see the following: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/637458/RA1300_Issue_3.pdf.

[10] *ibid.*

[11] RA 1005 – Contracting with Competent Organizations (*sic*).

sations which are to be involved in military platforms, e.g. to conduct maintenance. This allows consistency when comparing the organisations and allows an evidential threshold to be established.

Organisations which provide air systems (including products, parts, appliances), airborne equipment, air launched weapons, and post-design services are classed as Design Organisations (DOs) within RA 1005. There is a MAA requirement that only competent DOs will be contracted, and a DO needing to be approved under the Design Approved Organization Scheme (DAOS). Where a DO also holds a relevant EASA approval then there are related evidential artefacts that should be submitted to support any competence case.

RA 5850 (MAA, 2018*c*)[12] contains guidance on the responsibilities of a DO, information on the approvals scheme processes, and instructions on sustaining type airworthiness. Annex B of RA 5850 includes information on the DO exposition requirements. Granting of approval under DAOS to a DO is only relevant for given systems, in stated locations, within certain contexts. For DAOS approval the following DOs areas of business are assessed.

- Organisation structure.
- Human resources.
- Management of staff.
- Certifying personnel.
- Independent system monitoring.
- QMS certification evidence.
- Design process information.
- Design documentation control.
- Subcontractor selection process.
- Continuing airworthiness.
- Process to collect and investigate failures, malfunctions, and defects.
- Statement of qualification and experience.

[12]RA 5850 – Military Design Approved Organization (MRP 21 Subpart J).

Once a DO has gained DAOS approval they are included within a consolidated list of approved organisations which is published by the MAA[13]. Any approval is time-bounded with elements of the scheme being repeated periodically to maintain confidence in a DO.

The DAOS process has a number of benefits to the MAA and those organisations which wish to become a contracted DO. The process allows the MAA to gain suitable satisfactory confidence in an organisation, which is consistent across different organisations. From an industry perspective the DAOS process is a tried and tested approach (Kritzinger, 2017). However, the DAOS process has been classed as inflexible and difficult to navigate for software specific evidence (Kritzinger, 2017).

There is a level of confidence gained in the DO via the DAOS process. However, the use of direct evidence captured as part of the DAOS process could be enhanced to form mitigations to non-process compliance for the software. There is scope to include such evidence to form part of a direct and consistent narrative to support a software safety argument.

6.1.1.3 Military Aviation Authority (MAA) Mutual Recognition (MR)

The MAA have a process in place to evaluate other military regulators for their airworthiness artefacts. This can potentially support the DT air system safety argument submissions. The MAA states that the granting of Mutual Recognition (MR) status to an authority does not mean that the evidence and activities of the recognised organisation can be taken at "face value" as the context of the recognition must be assessed and understood (MAA, 2017*a*).

There are clear benefits to MR, e.g. with a reduction in costs. This is due to reduced duplication of effort and bureaucracy across the DAEs of the nations. In addition, a shared understanding of airworthiness information and principles can be gained as well as a shared certification approach, e.g. a major change[14] certified by one nation could be accepted by others (Robinson, 2016). The MAA has traditionally been committed to a recognition activity with other competent authorities where direct benefits may be realised (MAA, 2013).

As at December 2017, the last update to the publicly available MAA list of recognised military regulators, there are 7 organisations that have gained MR (MAA, 2017*a*). This includes regulators from:

- France (Direction de la Sécurité Aéronautique d'État (DSAÉ) for Airbus A400M).
- Spain (Dirección General de Armamento y Material (DGAM) for Airbus A400M and Eurofighter Typhoon).

[13]For further information refer to MAA (2018*a*).

[14]RA 5820 contains guidance on changes to the type designs of air systems. In essence, a change to an air system which has "*no appreciable* effect on the mass, balance, structural strength, operational characteristics, or other characteristics affecting the Airworthiness of the Air System" should be classed as a minor change. All other changes should be classed as a major change (MAA, 2017*e*).

- Germany (Luftfahrtamt der Bundeswehr (LufABw) for Eurofighter Typhoon).
- Italy (Direzione degli Armamenti Aeronautici e per l'Aeronavigabilità (DAAA) for Eurofighter Typhoon).
- United States of America (USA) (e.g. Aviation and Missile Research, Development, and Engineering Center (AMRDEC) Aviation Engineering Directorate (AED) for various platforms).

The number of MAA mutually recognised organisations is encouraging. It could play an important element in supporting the MOD procurements and subsequent certification activities in the future, e.g. in the post-Brexit era with European Union (EU) procurements and redefined EU directives (Butler, 2016). In addition, this will potentially support the procurement efforts with suppliers which have traditionally received significant portions of MOD procurement expenditure[15]. Examples include Finmeccanica SpA (Italy), Airbus Group (trans-European), and Hewlett-Packard (HP), Lockheed Martin Corporation (LM), and Boeing (all US)[16].

The MR process assesses regulator responsibilities in the following areas (MAA, 2017*a*):

- General functions of an airworthiness authority.
- Airworthiness inspection regulations.
- Production oversight regulations.
- Aircraft certification regulations.

The European Defence Agency (EDA) have a number of approved Military Airworthiness Authority (MAWA) documents which provide requirements for European member states to implement within their own military airworthiness regulations. These are the basis for the MAAs MR. EDA MAWA Forum (2018) states the requirements for such processes as:

- Aircraft maintenance training (EDA MAWA Forum, 2014).
- Continuing airworthiness (EDA MAWA Forum, 2015).
- Military flight test permit procedures (EDA MAWA Forum, 2016*b*).
- Information on the recognition process itself (EDA MAWA Forum, 2016*a*).

[15]This can have an advantage in that a consistent capability can be maintained.
[16]Company information in relation to MOD expenditure informed by Utterly and Wilkinson (2016).

EDA MAWA Forum (2016*a*) provides information on a number of areas which should be considered as part of establishing MR, either one-way or two-way. The *critical* elements, as they are termed within EDA MAWA Forum (2016*a*), includes:

- Primary aviation legislation.
- Specific operating regulations.
- Aviation system and safety oversight functions.
- Technical personnel qualification and training.
- Technical guidance, tools, and the provision of safety-critical information.
- Licensing, certification, authorisation, and approval obligations.
- Surveillance obligations.
- Resolution of safety concerns.

There is a degree of overlap in terms of the evidence which is requested from a DO as part of the DAOS process and that which is required for regulatory authorities within the MR activities.

6.1.1.4 Salient Observations: MOD Software Assurance (Airborne Platforms)

There are a number of observations that can be made on the current use and exploitation of evidence. There is a rich set of evidence which can form an initial judgement on specific PEs due to the broad set of evidence captured via DAOS and MR processes. The evidential requirements at a PE level will need to be much more detailed and focussed to understand specific PE risks; however an initial level of *confidence* can be established. Evidence from DAOS and MR processes could be evolved to account for PE specific evidence. This could remove the need to re-establish confidence via process-based means, e.g. life-cycle artefacts. The underpinning *principles* of the evidence captured as part of DAOS or MR procurement could form additional evidence in the event of PE level evidence shortfalls, for example. This could occur even if the direct evidence captured from the initial DAOS and MR activities is at a higher level of abstraction.

The process-based standards for software (DO-178C) and CEH (DO-254) which are traditionally adopted within MOD airborne domain procurements *do* make reference to the fact that PSH can play a role in establishing *confidence* in a given system. Indeed, the RE has been actively engaged in projects and explored a number of MOD airborne procurement

case studies. These case studies have adopted wider non-process based evidence as part of the software and CEH assurance arguments made to the MAA. The report in question (Standish, Hadley and Lennon, 2017) is releasable only within MOD; however the lessons from these case studies can be found within the research output titled *Use of Diverse Software Evidence within a Safety-Critical Software Airborne Qualification Strategy*[17] as well as supporting concepts within this thesis[18].

CAST-1 provides a method to judge PSH attributes and can be used in the event of shortfalls in software and/or CEH process-based evidence. However, the CAST-1 method does not form part of any formal MOD assurance guidance and is not necessarily fully adopted. Despite this the concept of assessing evidence attributes applied via engineering judgement is an approach which has a merit which is accepted by the FAA[19]. Many of the domains, e.g. medical, have formal structures in place to assist with gathering evidence types/weightings[20].

6.1.2 Land and Maritime Platform Software Assurance

The MOD airborne regulator[21], the maritime regulator[22] and the land regulator[23] are teams within the DSA (DSA, 2018*a*). Within the DLSR it is the Land Systems Safety Regulator (LSSR) which regulates the acquisition and use of equipment within the land domain (DSA, 2018*b*) with DS 00-56 (UK MOD, 2014*c*) being applied for the contracting of safety (LSSR, 2017). DMR02[24] (DMR, 2016*a*) and the Naval Authority Notice (NAN) Software Integrity Policy (SIP) (DMR, 2016*b*) is the focus for software assurance within the MOD maritime domain.

Within the land and maritime domains the context and environment of the platforms influences the severity of any hazards which can occur, e.g. the severity of a hazard associated with an engine failure within the air domain is greater than that within the land domain. Due to the differing contexts and environments, the guidelines and standards which are applied vary. Therefore, the approach to mitigate any perceived shortfalls in software

[17]The case studies were generated/reviewed as part of this research and subsequently published within the customer deliverables.

[18]For example by supporting the potential permissible evidence for MOD airborne safety-critical software assurance (see sub-section 7.1).

[19]Noting that CAST-1 is a FAA *guidance* paper.

[20]There is an argument that the medical domain is now becoming more pragmatic due to necessity, e.g. drugs to counter Ebola had a *very* limited level of testing before being adopted, although in reality there were very few alternatives! See the following for further information: https://jme.bmj.com/content/44/1/3.

[21]The MAA.

[22]The Defence Maritime Regulator (DMR).

[23]The Defence Land Safety Regulator (DLSR).

[24]DSA02-DMR - MOD Shipping Regulation for Safety and Environmental Protection.

evidence also differs. Interviews with Dstl colleagues and a MOD Naval Authority Group (NAG) representative has informed the REs understanding of the MOD land and maritime domains[25].

[Sub-section text redacted]

If there are known risks associated with any evidential shortfalls, within any domain[26] and with any assurance item[27], then the operational environments and capabilities of the system may be limited. Within the MOD air domain these limitations can result in a CLE[28].

6.1.2.1 Salient Observations: MOD Software Assurance (Land and Maritime Platforms)

The environments and contexts of the domains can shape the evidential requirements for assurance. The adoption of continuous operator mitigations, e.g. via Standard Operating Procedures (SOPs), can be generally sufficient for systems which are within environments which allow timely Human-in-the-Loop (HITL) or Human-on-the-Loop (HOTL) action(s)[29]. This reaffirms that the context to the use of software and CEH is paramount as the assurance requirements and evidence needs to be proportionate and sufficient.

Within the air domain there is a need to gain suitable confidence of a software's dependability prior to the adoption of the software into service. HITL, HOTL, and the fail-safe[30] options are significantly less tolerant within the air domain. However, it should be noted that non-air domains undergo significant assurance on focussed hazards, e.g. those that could lead to harm of the vehicle occupants. The aim, reasoning, and adoption of the 'proven in use' concept and the context of the environments are significant observations from the MODs land and maritime domains.

[25] A list of the research interviews are contained in Appendix B.

[26] Including non-MOD domains.

[27] Including non-software items.

[28] See MAA (2016*b*) for further information.

[29] HITL actions are those which "require a positive affirmation from the human operator for the machine to proceed" whereas with HOTL "the operator need not approve of the action beforehand but retains the ability to veto it before the execution of the machine's action or abort the action once it has begun" (Schaub and Wenzel-Kristoffersen, 2017). The exact distinctions and variations of these relationships is more complicated but for the purposes of illustrating the concept these definitions suffice.

[30] The concept of to *fail-safe* is in the event of a breakdown or malfunction the machinery reverts to a safe condition (OED, 2018*e*).

6.2 Software Assurance Within Other Safety-Critical Domains

There are a number of domains which operate within safety-critical environments, i.e. hazards may lead to *catastrophic* events. Each domain must ensure that the software performing the safety-critical functions is *robust* and *dependable*. Due to the various contexts in which the software is used the methods and principles that are adopted will differ. These will also be influenced by the political and technical environments. There are certainly lessons that can be learnt from these domains to inform the possible forms of permissible evidence and how they can be judged.

Within this section each of the domains are described[31]. At the end of the section there is discussion on the salient observations of the software assurance conducted within other safety-critical domains[32].

6.2.1 Civil Aviation Software Assurance

It should be noted that a key source of information for this section is the REs interview with a Civil Aviation Authority (CAA) representative[33]. Additional information is sourced from references including, but not limited to: CAA (2019*a*)[34], EASA (2017*a*), DfT (2018), CAA (2018), and CAA (2019*b*).

EASA have the remit for the regulation and assurance of airborne software for LRUs and platforms which operate within UK airspace. This is in keeping with the EU single Single European Sky initiative[35]. The CAA have a remit for the regulation and assurance of Air Traffic Services (ATSs) and the broader Air Traffic Management (ATM) (CAA, 2018, 2019*b*).

The CAA regulatory objectives for software safety assurance of *bespoke* software within ATM equipment are contained within CAP670 SW01[36] (CAA, 2019*a*). The guidance can also be used in circumstances where the published guidance does not fully address the needs of software within the scope of Article 5 of Regulation (EC) No. 482/2008; e.g. COTS software and changes to legacy software (CAA, 2019*a*). The document does not prescribe how the assurance evidence is to be produced or its adequacy argued, this is due to guidelines

[31] See Appendix B for further information on the *open/probing* questions adopted for the semi-structured interviews used to gather this information.

[32] This allows the various strands to be described and assessed in a single coherent sub-section.

[33] A list of the research interviews are contained in Appendix B.

[34] This sub-section has been amended to reflect version 3/2019 of the CAP670 regulations.

[35] See the following for further information: https://www.eurocontrol.int/dossiers/single-european-sky.

[36] CAP670: Air Traffic Services Safety Requirements. Specifically, Part B, Section 3, "SW 01: Regulatory Objectives for Software Safety Assurance in ATS Equipment".

such as DO-178B (RTCA, 1992) being able to be used in conjunction with the document. The prime software safety objective for ATS systems which contain software is "to ensure that the risks associated with deploying any software used in a safety related ATS system have been reduced to a tolerable level" (CAA, 2019*a*). To achieve the objective CAA (2019*a*) states that it is necessary "for arguments and assurance evidence to be available which show that the risks associated with deploying any software used in a safety related ATS system are tolerable".

CAP670 SW01 provides general requirements for the evidence of requirements satisfaction, e.g. that software safety requirements are satisfied. Differing sources of evidence are permitted for the different software safety requirements and the same evidence may be used for different software safety requirements if it valid to assess the requirements *independently*. The guidance itself only considers evidence from testing, field service experience, and analysis.

Within CAP670 SW01 the concept of Assurance Evidence Levels (AELs) are used to relate the safety criticality of the software safety requirement to the depth and strength of evidence required for the assurance of the correct implementation. The purpose of an AEL is to define the *minimum* set of assurance evidence required for a given software safety requirement for any proposed system. The AEL safety criticality level for a given safety requirement can be reduced with the consideration of *architectural* and *operational* defences within other parts of the system. For each type of evidence (i.e. test, field service experience, or analysis) the level of rigour is commensurate with the AEL. As an example, for field service experience AEL 1 requires relevant *statements* with AEL 5 requiring the completion of analysis, justification, and verification by an independent organisation. The evidence to satisfy the AELs should be formed of *direct*[37] and *backing*[38] evidence. CAP670 SW01 also provides guidance on the *behavioural* attributes of a software safety requirement and the expected supporting evidence. As an example, analysis and testing for AEL 5 timing properties with worst case timing analysis and performance modelling forming part of the analytical evidence.

The AMC document to CAP670 SW01[39] (CAA, 2010) aims to provide guidance to Air Navigation Service Providers (ANSPs) and their suppliers on addressing the objectives of CAP 670 SW 01 (CAA, 2019*a*) when deploying COTS equipment.

The AMC for COTS (CAA, 2010) states that there are at least four types of integrity

[37]With *direct* evidence being that which is produced by an activity taking place or software behaviour occurring, which is directly related to the claim being made (CAA, 2019*a*).

[38]With *backing* evidence being that which shows that the *direct* evidence is both credible and soundly based (CAA, 2019*a*).

[39]Acceptable Means of Compliance to CAP670 SW01. Guidance for Producing SW01 Safety Arguments for COTS Equipment.

assurance evidence which can be adopted to claim a suitable level of assurance confidence. The evidence is to gain a direct level of confidence in the *software* itself. Therefore, the approach is not reliant on a process-based review. The use of varied evidence strands to inform the direct software confidence is positive in supporting the thesis arguments to enhance the software assurance for the defence domain.

- Testing.
- Field-service information.
- Supplier experience and reputation.
- Supplier software design and development.

The AMC attempts to quantify the judgements on the evidence to allow the *impact* of the evidence to be measured. This is achieved by a points based system which has a defined target dependent on the required fault free hours, e.g. set number of integrity assurance points for no worse than 1 x 10^{-4} occurrences per hour. There are varying levels of confidence (scaled 1-3) which can be attributed to the evidence, e.g. if the evidence is *below* the expected requirements. The levels have a range of 'points' associated with them. To assign the level of confidence each form of evidence states the artefacts expected to be available and reviewed. The AMC for COTS provides information on the range of points which each evidential type can provide towards the target. An example is shown in Table 6.3[40].

Evidence	**Points**
Appropriate software development process	*10* for a level *below* the recommendation ↓ *25* for a level *meeting* the recommendation ↓ *30* for a level *above* the recommendation

Table 6.3: Example of Points Associated to Software Evidence (CAA, 2010)

In essence, the points-based approach is cognisant of the *quality* of the evidence as it's being measured against a *recommended* level. Also, the *relevance* of the evidence is captured as it's being judged in relation to the other forms of evidence.

The CAA adopts the EU2017/373 regulation[41] (EASA, 2017*b*) which supersedes the

[40]Extracted from Table I.4 within CAA (2010).

[41]Regulation (EU) 2017 373 - the Air Traffic Management Common Requirements Implementing Regulation (ATM IR).

EU482/2008 regulation[42] (EASA, 2008). EU482/2008 was an attempt to focus on a software process assurance approach whereas EU2017/373 ensures that software is one element of many for safety judgements. This indicates that non-process based software evidence can be legitimately considered.

[Sub-section text redacted]

The context in which the CAA software assurance is adopted should be noted. As with other domains, there are a number of mitigations that can be enacted if a software failure occurs; there is a very high level of HITL within a ATS for example. Therefore, the very limited set of mitigations within an airborne software domain does not apply to an ATM environment.

6.2.2 Civil Air Traffic Services Software Assurance

It should be noted that a key source of information for this section is the REs interview with a National Air Traffic Services (NATS)[43] representative[44].

[Sub-section text redacted]

For ground-based systems NATS adopts the European Organization for Civil Aviation Equipment (EUROCAE) ED 109 standard. In essence, EUROCAE ED 109 has the same requirements as the DO-278A[45] guideline (RTCA, 2011*c*). DO-278A calls out DO-178B (RTCA, 1992)[46].

6.2.3 Automotive Software Assurance

It should be noted that key sources of information for this section are RE interviews with two Vehicle Certification Agency (VCA) representatives[47].

[Sub-section text redacted]

[42]EU 482/2008: Establishing a Software Safety Assurance System to be Implemented by Air Navigation Service Providers.

[43]The acronym *NATS* is now only the company name although the name was derived from National Air Traffic Control Services (NATCS) originally. See the following for further information: https://www.nats.aero/about-us/our-history/.

[44]A list of the research interviews are contained in Appendix B.

[45]DO-278A. Guidelines for communication, navigation, surveillance and air traffic management (CNS/ATM) systems software integrity assurance.

[46]DO-278A is unofficially termed the "DO-178B of the ground".

[47]A list of the research interviews are contained in Appendix B.

6.2.4 Rail Software Assurance

It should be noted that a key source of information for this section is the REs interview with a rail domain representative[48].

Within the rail sector the predominant standard adopted for software assurance is an equivalent to International Electrotechnical Commission (IEC) 61508, called IEC 62279[49]. There are two communities within the rail domain: signalling and then the other elements, including rolling stock. Edition 2 of IEC 62279 will move from a focus on the signalling to include other aspects (such as rolling stock).

[Sub-section text redacted]

RSSB (2017) is a UK Rail Industry Guidance Note which aims to assist with providing information for a number of defined processes for High-Integrity (HI) software and software-based systems[50]:

- Procurement.
- Preparation of specifications.
- Contractual arrangements.

RSSB (2017) is aimed at companies which procure HI software. The guidance states that there are a number of methods that can be used to provide a suitable level of rigour, for example: V&V; clear documentation and traceability; appropriate consideration of organisation; and personnel competency issues. The guidance recognises that software testing, in all its forms, is only part of the validation of the software. The guidance refers to the need to have a number of requirement categories (e.g. safety, security such as protection from cyber-attacks) but the guidance does not elaborate further on the security requirements[51]. Animation and model simulation are also seen as a useful methods of verification.

The rail industry reviews a broad level of evidence, including simulation, to *directly* inform the confidence in the software safety assurance. This is in keeping with other safety domains.

[48] A list of the research interviews are contained in Appendix B.

[49] IEC 62279: Railway applications - Communications, signalling and processing systems - Software for railway control and protection systems.

[50] HI systems encompass those which can be mission, security, or safety-critical. They are designed to have a high probability of conducting their intended behaviour.

[51] Rail industry guidance on cyber-security is available from https://www.rssb.co.uk/Library/improving-industry-performance/2016-02-cyber-security-rail-cyber-security-guidance-to-industry.pdf.

6.2.5 Civil Nuclear Software Assurance

It should be noted that a key source of information for this section is the REs interview with a ONR representative[52].

[Sub-section text redacted]

A common position report[53] contributed to by a number of Nuclear Regulators aims to provide a consensus of eight international regulators (1 from Canada and 7 from Europe). The report consists of two major parts:

- Generic licensing issues; e.g. pre-existing software, security, and formal methods.
- Life-Cycle phase licensing issues; e.g. software implementation and verification.

The report states that software assessments cannot be limited to verification and testing of the end product. Other factors should be considered, e.g. process quality. The view is that there are three basic independent types of evidence that must be produced to support the safety demonstration of a "computer based digital system":

- Quality of the development process.
- Adequacy of the product.
- Evidence of the competence and qualifications of the staff involved in all of the system life-cycle phases.

The report supports the use of pre-existing software as it can increase confidence that the system is *safe*. International Nuclear Regulators (2018) contains recommendations such as: need for a well understood life-cycle approach, e.g. requirements, design, and implementation; the need for quality assurance; and for graded requirements compared to classes of software etc. Wider considerations such as organisational requirements, e.g. safety culture and staff competencies, are also included.

A ONR guide on Computer Based Safety Systems (ONR, 2017) aims to advise and inform ONR inspectors for when they perform their professional regulatory judgements. The guidance draws upon a number of supporting standards, e.g. IEC 62340[54]. ONR (2017) contains a large guidance section on software for computer based safety systems. This covers the:

[52] A list of the research interviews are contained in Appendix B.

[53] International Nuclear Regulators (2018). Licensing of safety critical software for nuclear reactors. Common position of international nuclear regulators and authorised technical support organisations.

[54] IEC 62340:2007 - Nuclear power plants - Instrumentation and control systems important to safety – Requirements to Cope with Common Cause Failure.

- Need for defensive programming[55].
- Need to run in a fixed sequence pattern, rather than employ interrupts.
- Coding standard requirements, both prohibited practices and those which are encouraged.
- Use of static analysers by the designers with an independent tool used by the assessors.
- Checks that should be adopted by V&V teams, e.g. appropriate use of fault tolerant and defensive programming features.
- Testing principles, extended period of testing to establish reliability.

6.2.6 Health Information Technology (IT) Systems Software Assurance

It should be noted that a key source of information for this section is the REs interview with a representative with the company Consultants to Government and Industries (CGI)[56].

[Sub-section text redacted]

6.2.7 Salient Observations: Software Assurance Within Other Safety-Critical Domains

The review of a number of domains which undertake the assurance of safety-critical software has allowed a range of trends and observations to be made. These can significantly inform the evidence which can be permitted for a software assurance argument. Obviously, the contexts in which the domains operate influences both the methods and evidential requirements.

There are a number of process-based standards which are adopted within the domains, with many based upon IEC 61508. These standards set an initial benchmark in terms of the expectations on suppliers by the regulators, Certification Authorities, and assessors. However, there is also an acknowledgement within a number of domains that the preferred process-based evidence is not guaranteed to be provided in all instances. Therefore, it is common for wider, more diverse, evidence to be adopted to support the software safety assurance argument. Evidence includes, but is not limited to: SQEP data, in-service use, formal methods, organisational competence, test plans/results, and safety hazard analysis.

[55]Defensive programming aims to ensure that software will function under unforeseen circumstances. The method restricts the functionality of the software to increase the probability that it will perform as intended. This has safety and security benefits and can be adopted within the airborne domain.

[56]A list of the research interviews are contained in Appendix B.

Such evidence forms part of a standard argument within some domains. However, the level of weighting for the evidence is applied inconsistently across the domains. The ability for such diverse evidence to significantly support other evidence strands is also applied inconsistently.

Evidence requested by regulators, Certification Authorities, and assessors can vary within a single domain to ensure that the evidence is proportionate to the context of the system[57]. Other domains do not have the nuances or sophistication of the integrity levels and so the range of evidence sought is reasonably consistent from system to system.

One of the original arguments for this thesis was that there is a need to significantly reassess how diverse evidence is adopted to expand the system solution space (see Figure 1.1). This concept is not confined to the MOD airborne software assurance domain. Other domains have similar challenges which require adopting a wider set of evidence to support and mitigate perceived evidence shortfalls. In some instances, the concerns regarding access to supporting artefacts can be mitigated via the use of Crown Servants, such as Dstl.

All of the domains discussed must assess evidence for brownfield and greenfield projects. Each has its own challenges and a greenfield project does not necessarily lead to full initial compliance with the software assurance requirements. The supplier may not necessarily adhere to the initial standards and the issues regarding release of information are the same for brownfield and greenfield projects. An approach which embraces diverse evidence is required for both types of projects.

Within some domains there is an acceptance that there are benefits to allowing suppliers to use an in-house development process rather than a defined standard prescribed process. There are merits to this approach as long as the in-house development is suitably robust. Adopting such a view further supports a need to provide a holistic view of the evidence. Such evidence presented for a system needs to have the supporting evidence ascertained on a case-by-case basis.

The levels of review conducted to gain sufficient confidence varies between the domains and within some domains this is further dependent on the criticality of the software. Within one domain there is a requirement to gain full objective-by-objective evidence and to mitigate where required. Other domains take a suitable sampling approach. As identified previously, the context and the environment shapes these requirements and the ability to tailor the evidence.

Some domains can have no preference for the diverse evidence which is judged, based upon having limited supplier information and a lower severity of hazards. Whereas other domains have clear hierarchies of evidence and strong concepts of which evidence is valid. Interestingly, some of the hierarchies are from a stringent regulatory perspective, based

[57]Such as the environment of the operations or the integrity level of the software.

upon standards. Others are based upon SME judgement to inform the software assurance argument. The preferences for evidence to be reviewed is based upon the practicalities of receiving the supporting information and the legal requirements which are adopted.

Counter-evidence is an important element to the software assurance processes of a number of domains, as with the MOD airborne software assurance requirements. The ability to gather counter-evidence requires strong supporting legislation. There is a necessity for data owners to allow such information to be reviewed and assessed.

COTS evidence is a common thread within the domains. Legislation may support the review of suitable information; however, for COTS systems the available evidence may be limited and increasing the legal requirements will not assist in such circumstances. Again, this is a supporting case for the use of diverse evidence to mitigate such a lack of sufficient evidence.

There are differing levels of rigour and depth of software evidence reviews, and therefore there are varying levels of *trust* which is placed in the suppliers. This *trust* is based upon the suppliers being able to suitably develop the software in the initial and continuing phases, e.g. for software fault resolution. This *trust* can only be established by a firm understanding of the organisational processes and, where possible, the pedigree of those processes. *Trust* is not a concept that can be granted blindly and therefore continual review and assessment is required.

The differing software architectures and contexts of use results in terminology and definitions changing between the domains which have been discussed. As an example, the concept of *simple* and *complex* hardware/software differs between domains and this alters the expectations of the evidential artefacts/results and the ability for sufficient reviews to be undertaken. This supports MAA comments with regard to the need for consistency, not only with the adoption of standards but in terms of the methods in which cross-domain evidence is assessed. This is to be noted for this research and for scenarios in which supporting evidence is fed from one domain to another, for example any testing requirements/results from the rail domain to the airborne domain.

There is a clear preference within the discussed domains for simple and non-novel designs/concepts to be adopted. Traditionally it is easier to gain suitable assurance confidence in these types of systems. The concepts can be 'easily' understood with designs adopting solutions with pedigree. However, as stated within the argument for this thesis, there are cases where the use of novel technology is becoming less optional and more a hard requirement. Therefore, there is a need for the domains to adapt. This can be in terms of the interpretation of how existing standards can be achieved but also in terms of how the standards reflect the changing technology landscape.

A number of domains provide a consensus on a number of aspects which are in keep-

ing with the current MOD software assurance policy. However, the evidential requirements/realities of the domains does prompt the question as to what additional evidence could be used to inform a safety argument beyond the traditional life-cycle elements; e.g. animation and model simulation and the ramifications of the evidential requirements for this. A review of the domains leads to a conclusion that additional evidence could be used to inform a direct software safety argument (e.g. staff competencies, common cause failures etc) and also the need to consider how such evidence will be measured.

6.3 Literature to Inform Software Assurance Evidence

The sub-sections which follow are *additional* to the activities which are conducted in current safety assurance arguments; however, they are potentially suitable to be considered as evidence strands.

6.3.1 Software Architecture Complexity

Process-based standards would normally require a *suitable* architecture to be designed. However, a deeper understanding of the software architecture could act as a form of supporting diverse evidence. This is due to not all software being of equal complexity.

Complexity can drive the maintainability of software and also the ability for a system to be *extensively* tested. Therefore, there is an argument that the complexity of the software and the associated assurance activities should be assessed in their context. Does a *very simple* system require full process compliance compared to *very complex* software? This approach is quite nuanced and is, as with other evidence, subject to SME judgements.

Clements, Kazman and Klein (2001) propose a number of metrics which can allow code to be located within larger code-bases which may be challenging to maintain or which does not adhere to software design RGP . The Clements, Kazman and Klein (2001) metrics are as follows:

- Number of events (synchronous and asynchronous calls) to which an object reacts.
- Number of synchronous calls made by an object to other objects to get or set some data/resource.
- Number of asynchronous calls made by an object to other objects.
- Number of component clusters of which a component is composed.
- Depth of structure (layers of encapsulation).

- Depth of Finite State Machine (FSM).
- Number of data classes used or referenced by an object.
- Number of extended state variables.
- Depth of inheritance tree.

The ability to capture such information may be limited due to the level of source-code detail required for this. However, the software complexity illustrates that *relevant* evidence can inform a judgement on the diverse software safety assurance confidence. In theory, the more simple the software architecture the less likely errors are to have been introduced. Complex software architectures may have a higher risk of errors occurring due to the increased probability of an incorrect implementation, for example. Also, understanding the complexity of the architecture and the safety-critical function it provides will allow focussed reviews of relevant artefacts.

6.3.2 Data Safety Working Group: Data Safety Guidance

DSIWG (2018) states that the role of data is becoming more prominent and therefore, data needs to be considered as a "first class citizen" within any system safety analysis. This concept is supported by the fact that DSIWG (2018) provides information on over 25 incidents and accidents, from numerous domains: naval, air, space etc, where data "failures" can be considered to be a contributory factor.

The Data Safety Guidance provides the concept of a Data Safety Management Process which is based upon four phases:

- Establish context.
- Identify risks.
- Analyse risks.
- Evaluate and treat risks.

Each of the phases have clear objectives and outputs, e.g. to identify key stakeholders within the "establish context" phase, in addition to a set of activities to achieve the objectives. The adoption of such principles could provide additional evidence to support the perceived level of rigour of the software.

Wider evidence such as that associated with data safety can inform a perceived software safety assurance confidence level. This is due to the potential *relevance* of the evidence and the *contribution* it has according to the SMEs forming the judgements.

The data safety concept and that of software architecture complexity is concerned with the *context* of the software. It attempts to establish the role that the software plays in the wider system and the actual nuances regarding the software itself. It is intended for these evidence types (data safety and software architecture complexity) to be considered within any framework which is developed as part of this thesis.

6.4 Summary of Current Permissible Evidence for Safety-Critical Software Assurance

This chapter has reviewed a number of software safety domains which adopt a range of evidence to support the safety assurance confidence. The diverse evidence is not only used to support the wider system safety review but it also *directly* informs the *software* assurance. Evidence can also be judged by applying a quantified measurement to gather 'points' to obtain an assurance argument, e.g. in the civil airborne domain. From a military airborne software assurance review this is positive as within other domains diverse evidence frequently judges process-based and non-process evidence.

There is also information from wider guidance and literature which can inform an assurance argument. This ensures that the software is being considered as a *unique* entity which is being judged on its own particular characteristics.

The information reviewed within this chapter allows assertions to be made on the evidence which may be permissible to a software safety assurance argument, see Chapter 7.

Chapter 6 has partly responded to the research sub-question: *What is the current permissible software safety assurance evidence within the UK defence domain and related domains?*.

Chapter 7

Potential Permissible Evidence, Underpinning Principles, and Stakeholder Engagement

The previous chapter reviewed a number of safety-critical domains to explore how they gather and assess software evidence. Each of the domains have challenges in gaining a level of confidence and assurance[1] of safety-critical software. The domains differ in context and this impacts on the acceptable tolerances for certain hazards. This is due, in part, to which mitigations can be implemented[2], which in turn impacts the acceptable evidence and its weight within any assurance argument. This chapter will explore the forms of potential evidence in further detail.

This chapter will examine:

- *Potential Permissible Evidence for MOD Airborne Safety-Critical Software Assurance.* Evidence which *could* inform a diverse assurance argument.
- *Underpinning Principles for the use of Evidence.* Information on a number of fundamental principles to consider when using diverse evidence for a software and/or CEH assurance argument.
- *Importance and Unintended Consequences of Metrics.* The metrics for software assurance need to be chosen intelligently. An insight is provided into the unintended consequences of using incorrect metrics in the software supply chain.

[1]The distinction between the two terms in this context is that *confidence* provides "trust in a thing" and "showing [a level of] certainty" (Collins Dictionary, 1995*b*) with *assurance* being "a positive declaration intended to give confidence" (OED, 2018*a*).

[2]For example, HITL and HOTL.

- *Communicating Evidence with Stakeholders.* How can stakeholders comprehend and debate a varied set of evidential data and sources? Is there a way to drive *better* decision making?

7.1 Potential Permissible Evidence for MOD Airborne Safety-Critical Software Assurance

The reviews of the software assurance domains in the previous chapter show there are a range of evidential items which could inform a diverse software argument for airborne PEs. The *strength* of such evidence would be subject to stakeholder dialogue, e.g. between the MAA, DT, and the ITE. The concept of evidence being able to *inform* an argument to a *degree* is one which should be exploited. Figure 7.1 shows the potential evidence strands which can form a part of a diverse evidence argument; each will have varying levels of *weighting*[3]. When constructing a diverse software assurance argument the features of both the evidential types (e.g. software life-cycle) and the supporting evidence of the types (e.g. software requirements planning) will need to be considered[4].

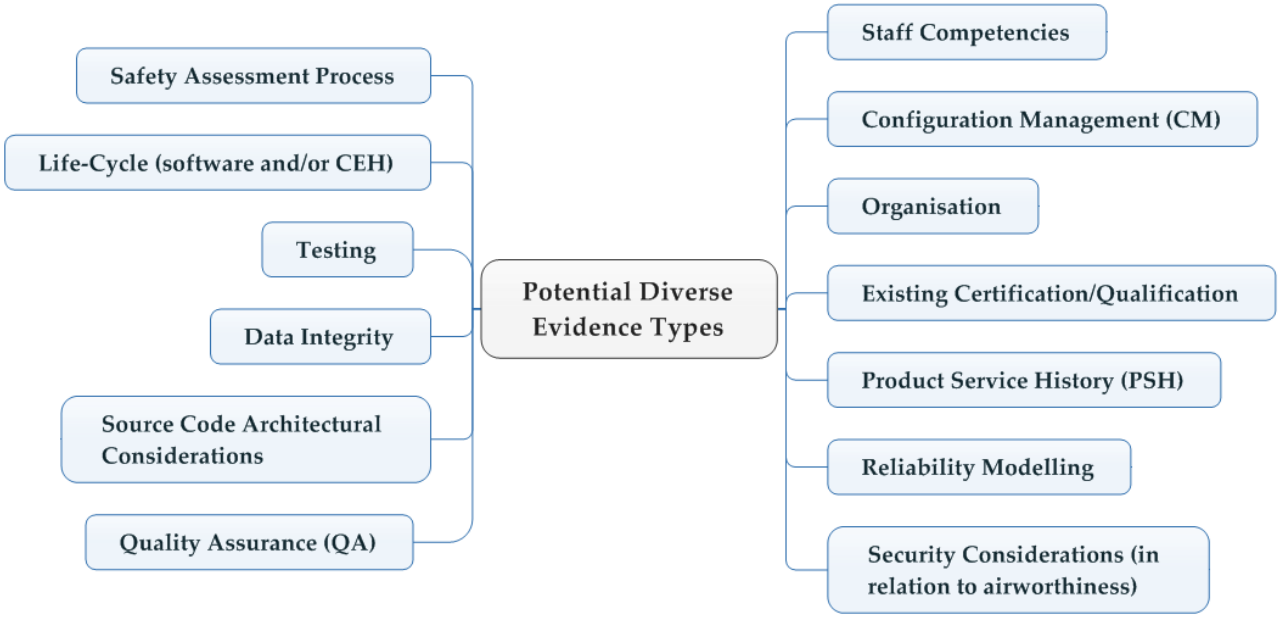


Figure 7.1: Potential Diverse Evidential Types

[3]The sub-sections which follow are stated within Figure 7.1 in an anti-clockwise direction.
[4]See sub-section 8.2.2.

7.1.1 Safety Assessment Process

The creation of, or amendment to, safety-critical systems needs to consider the safety functions of a system, or sub-system, and how these functions interact. The safety assessment process defines the required DALs. This then underpins the software and/or CEH development. The safety assessment process includes, but is not limited to, a number of methods such as Functional Hazard Assessment (FHA), System Safety Assessment (SSA), Fault Tree Analysis (FTA), and Common Cause Analysis (CCA). An example of a suitable safety assessment process is outlined within Aerospace Recommended Practice (ARP) 4761[5] (SAE, 1996). Figure 7.2 shows a simplistic flow of the DAL *determination*, *assignment*, and *implementation*[6].

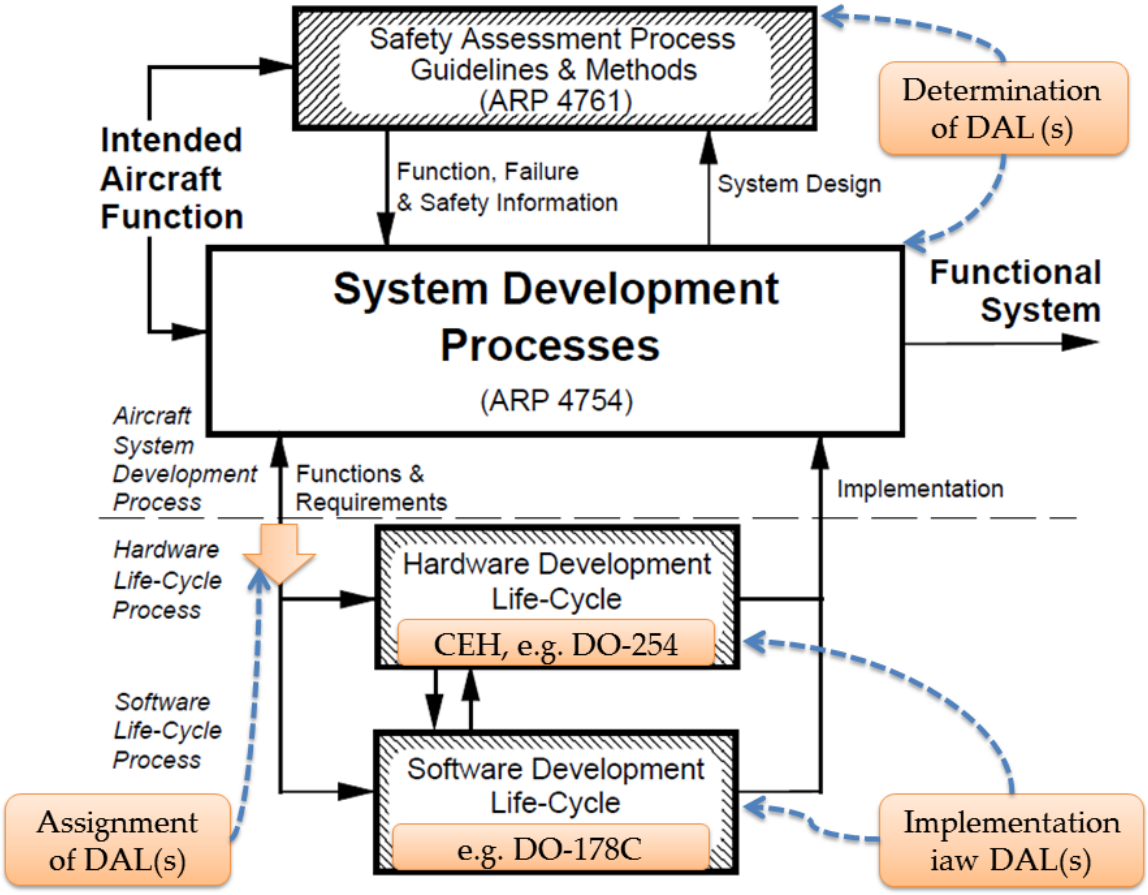


Figure 7.2: Flow of the DAL Determination, Assignment, and Implementation (adapted from SAE (2010))

[5]ARP 4761. Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment.

[6]Figure 7.2 contains the abbreviation *in accordance with (iaw)*.

7.1.2 Life-Cycle (Software and/or Complex Electronic Hardware (CEH))

The life-cycle activities for software and CEH consists of a number of stages. These result in a solution which is adopted within the final system build. The stages traditionally include: requirements capture, design, implementation, testing, deployment, and maintenance. During the life-cycle process the product matures and builds upon the decisions and outputs from the preceding stages. It could be argued, that the requirements are fundamental to any development as any errors and omissions propagate into the later life-cycle stages. This can be termed a 'snowballing' scenario. Figure 7.3 shows a simplified V-model with the flow of the stages stemming from the *requirements*.

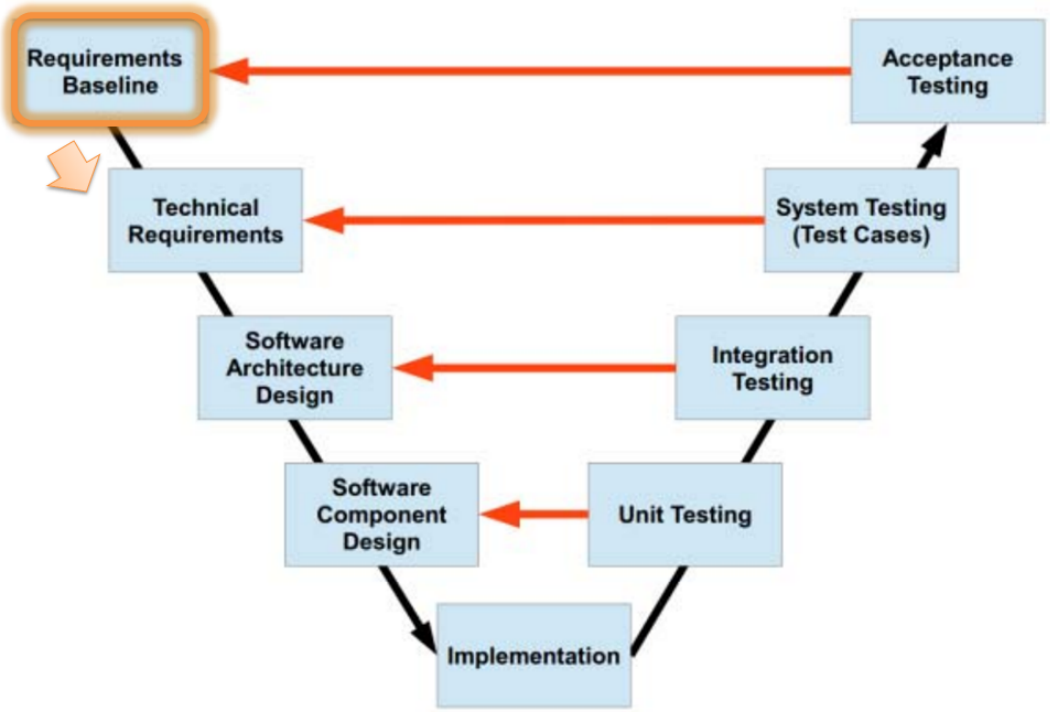


Figure 7.3: Simplified Software Life-Cycle V-Model (adapted from Ghanbari (2016))

There is also a need to consider the *pedigree* and *novelty* of any life-cycle approaches which are adopted. A 'waterfall' approach is a traditional method with each stage instigated after the completion of the preceding stage. However, there is a move towards adopting more modern life-cycle approaches within the domain, e.g. agile methodologies with iterative patterns of development[7]. It is important to consider the pedigree of use within the

[7]There are challenges to adopting *agile* methods within the safety-critical domain due to there being a

wider domain but also the pedigree of use *within* the organisation itself. How experienced the organisation is at implementing the life-cycle approach will have implications on the confidence of such evidence. Figure 7.4 shows a simplistic view of the differences between *plan-driven* (e.g. waterfall) and *agile* methodologies.

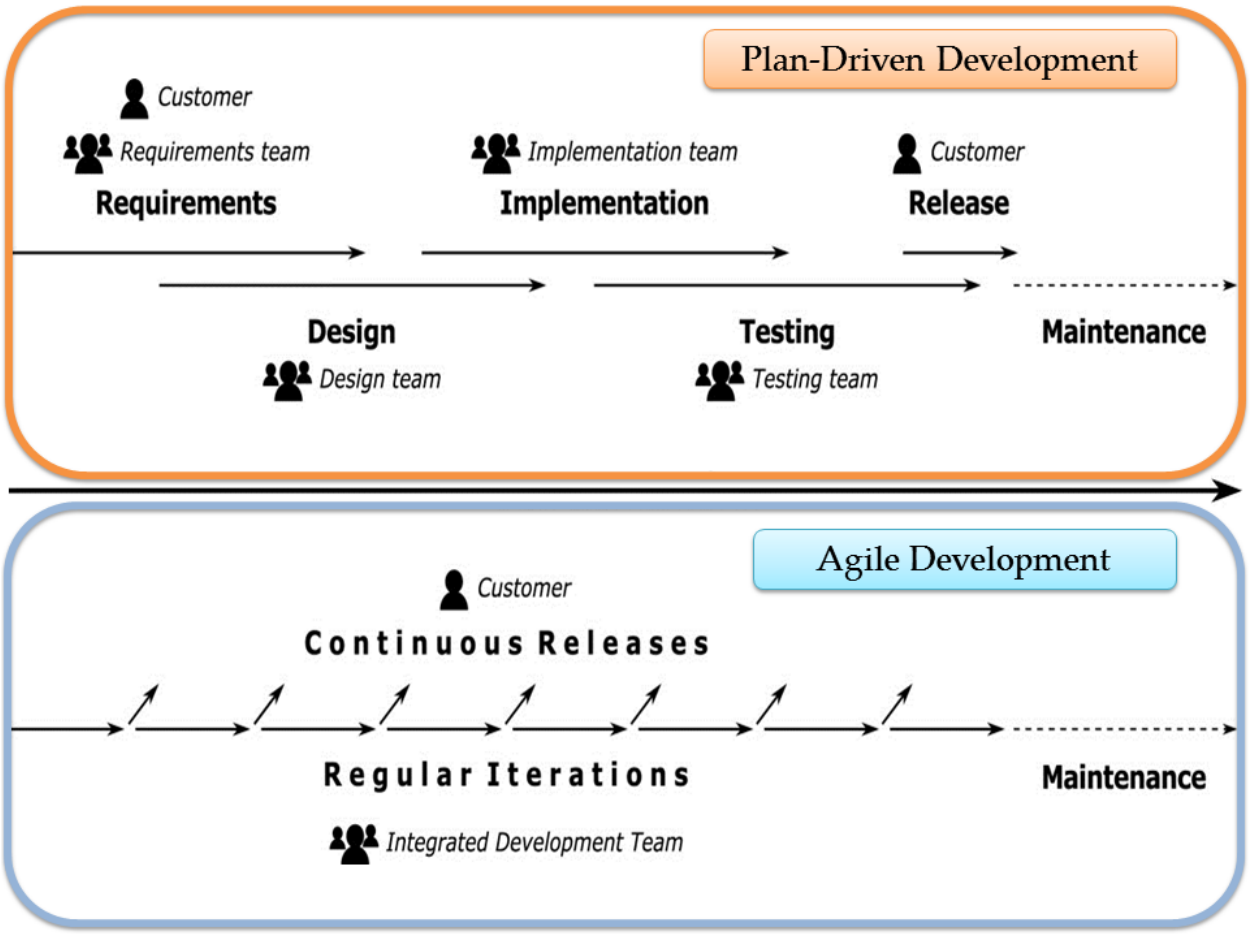


Figure 7.4: Differences Between the Plan-Driven and Agile Development Methodologies (adapted from Kaisti, Rantala and Mujunen (2013))

7.1.3 Testing

As stated by Myers (2004), "software testing is a process, or a series of processes, designed to make sure source code does what it was designed to do and that it does not do anything unintended. Software should be predictable and consistent offering no surprises to users". This description is also valid for CEH in the context of DO-254. There are a range of test

traditional focus on documentation and the level of upfront planning which is required, for example (Kasauli et al., 2018).

strategies, e.g. structural coverage methods, which allow confidence to be gained in the implementation. Figure 7.5 shows a number of typical software testing approaches.

Capturing test information should not be limited to the code created/revised *during* the development activities. Information should also be captured on the testing from the *ongoing* use of the implementation itself, in essence the BITs, e.g. during system initialisation. These BITs, although requirements driven, provide additional confidence in relation to the system being able to capture and correct errors. Figure 7.6 shows a number of the typical types of software BIT approaches.

7.1.4 Data Integrity

The accuracy and consistency of the data used by a system is vital to ensure that the system performs in a *correct* and *predictable* manner. Data integrity issues can have *catastrophic* safety implications[8]. Data integrity errors can manifest themselves via a number of channels such as data recording or data transfer. The design and development of the airborne software can exacerbate or mitigate such data integrity issues. Vulnerabilities in the design and/or development of the software can also allow exploitations via cyber-attacks.

The data itself can take a number of forms, including that used by an application and the data about a system, e.g. configuration data file. Therefore, there is a need to have confidence in the types of data used, e.g. application or system, and also in the methods adopted to retain and transfer such data.

There are methods to minimise the impact of low data integrity via activities such as initial risk assessments to understand the data of concern and also via the implementation of checks and controls. Any assessment of the data integrity would take into account the risk assessment and subsequent actions. Confidence in the underpinning data allows confidence in the accuracy of the system.

7.1.5 Source Code Architecture

An assessment of the source code architecture requires access to low-level information regarding the structure of the code itself. The architectural considerations can provide an indication of the complexity of the code which can impact the code's maintainability. Complexity can also potentially introduce errors during the code development. Architectural considerations include, but are not limited to, the number of data classes and the number of synchronous calls.

[8]As described in sub-section 3.1.1 with the Airbus A400M accident in 2015 (Gibbs, 2015, Gallagher, 2015).

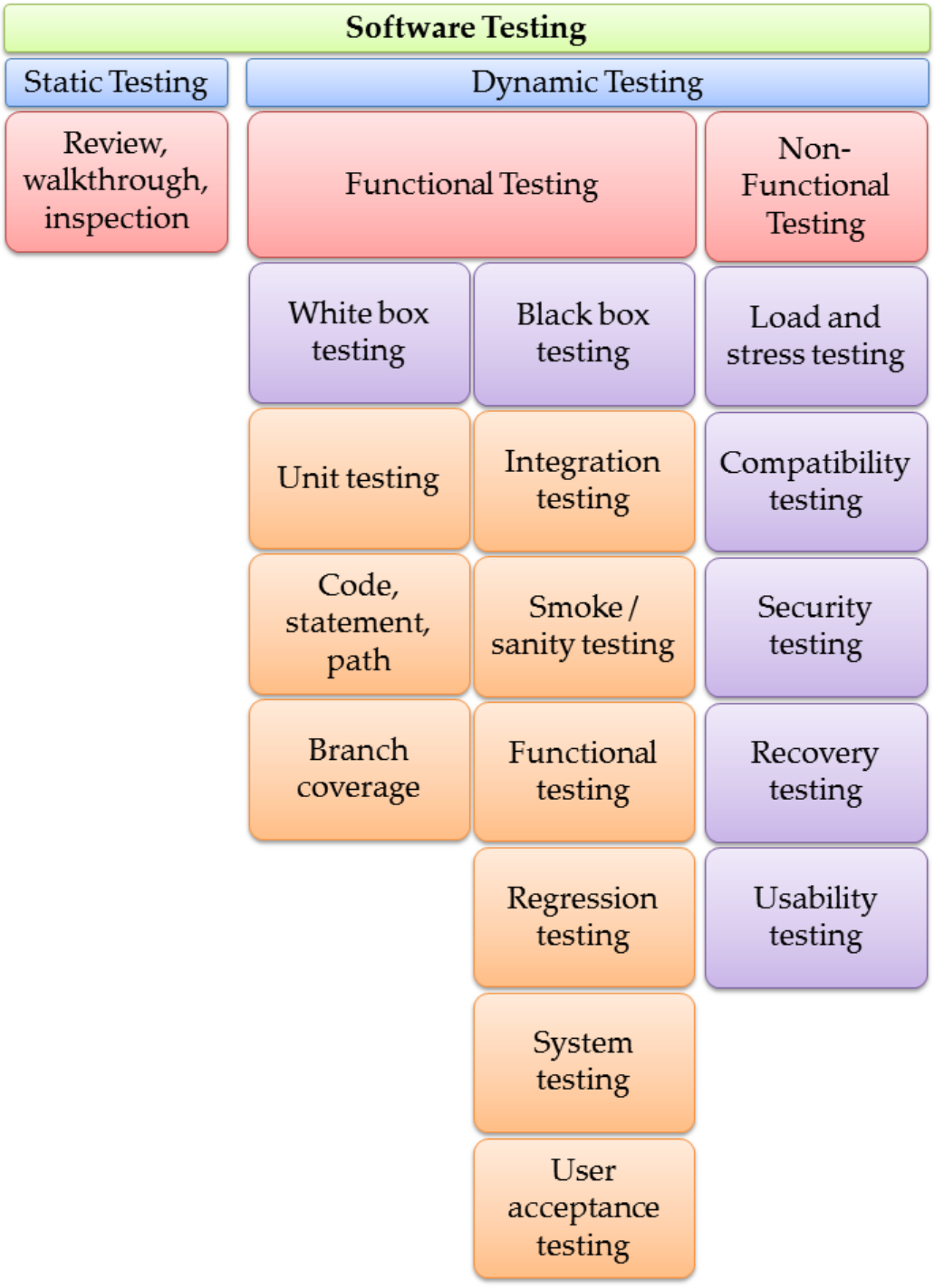


Figure 7.5: Examples of the Types of Software Testing Approaches (adapted from Functionize (2018))

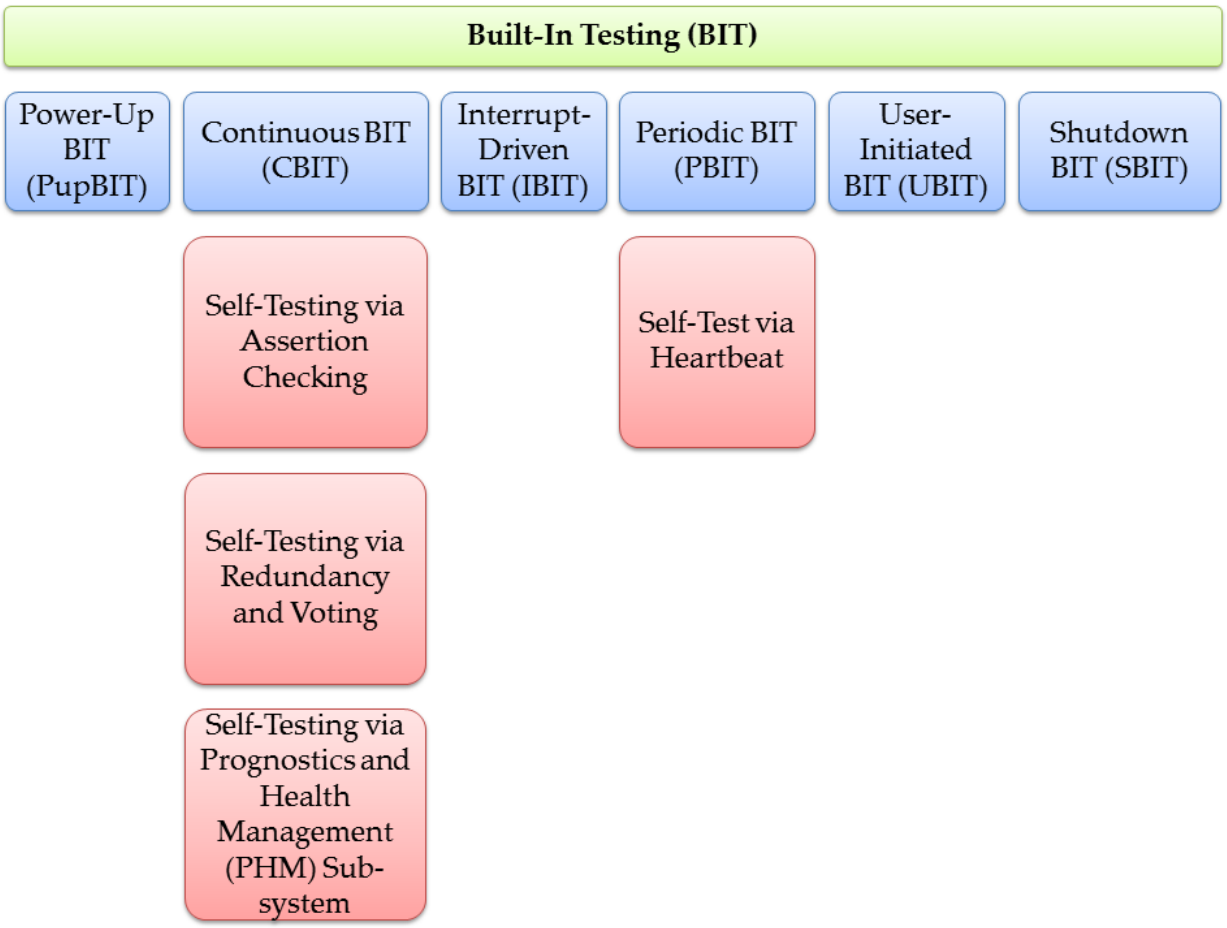


Figure 7.6: Examples of the Types of Software BIT Approaches (adapted from Firesmith (2015))

There are a number of attributes which can be used to judge the confidence in the source code architecture. The confidence can be formed via an analysis of the in-service system, e.g. performance and reliability. Confidence can also be gained via the development principles for the source code, these are part of a group of characteristics classed as 'design for attributes', e.g portability and modifiability.

It is possible for such information to be obtained via evidence of compliance to a detailed coding standard. However, access to such information can be an issue due to restrictions on the release of such data by the vendor. This may be the case if there are broader limitations on accessing life-cycle data. Source code architecture information is a significant element of the Intellectual Property (IP) owned by vendors. However, methods can be used to obfuscate the low-level information to ensure that only the metrics of interest are captured.

7.1.6 Quality Assurance (QA)

Assessing evidence which is part of a diverse argument requires a number of subjective and independent judgements. This independence plays an important part in allowing confidence to be gained in any system assessment. The concept of *independence* is also relevant to the development process itself to ensure that there are sufficient measures in place to provide ongoing confidence in the activities conducted. Confidence from the QA process comes from knowing that a QA process exists, that the QA process is conducted, and that the QA has been applied to the project.

The QA process covers a substantial number of the development stages for software which is an 'in-house' product. There is also a need to consider the QA processes which are adopted by third-parties; either for COTS or via subcontracting any element of the life-cycle stages. The prime vendor needs to ensure that there is sufficient supplier management and oversight in place as this forms part of an integrated and effective QA process.

7.1.7 Staff Competencies

It is recognised within the literature[9] that the competencies of those involved in the development life-cycle stages, including the QA, is crucial to achieving a satisfactory system. The competencies are not only linked to education but also the level of experience, especially if gained on equivalent projects. Considerations should also include any membership of professional organisations, such as chartership status, and the associated code of ethics/professionalism of members.

[9]For example, Acuna, Juristo and Moreno (2006), UK MOD (2012), SEBoK (2018*a*), IEEE (2018), and UK MOD (2018).

Confidence in the vendor's staff competencies can be gained via a number of metrics. There are the initial qualifications and level of experience expected when staff are recruited by the organisation. The ongoing training and development activities are other considerations. These include the prolonged mentoring and 'buddy' systems that may be in place. For the safety-critical software domain relevant staff should have the requisite skills to *perform* the life-cycle tasks. In addition, relevant staff should have the right skills to *judge* the competencies of other staff members where required, e.g. for QA.

7.1.8 Configuration Management (CM)

CM is an important overarching premise. It is not only relevant to providing confidence in the life-cycle stages[10] but also in allowing confidence to be gained in the evidence related to in-service use[11]. The CM process is one that has significance within DO-178C and other software life-cycle guidelines.

The CM process covers a range of activities within the software development life-cycle. The process allows the identification and management of artefacts, including documentation and source code, as well as ensuring that any changes are managed in a consistent manner. CM also assists with post-development activities with expectations for the archiving, retrieval, and release of software/documentation. This is in addition to the processes adopted for the reporting of any errors discovered during the in-service phase of the software, traditionally via *problem reports*.

7.1.9 Organisational Competencies

Staff competency assessments are focused on the skills and abilities of those *directly* involved in the software/CEH development. Organisational competency processes are focussed on having suitable methods embedded in the organisation which allow staff to perform their roles effectively. There is also a need to ensure that any associated activities, e.g. QA, are performed using rigorous and recognised approaches. Gaining insight into the organisational processes allows a more holistic view of the capabilities as a range of areas can be assessed, e.g. procedures to meet customer requirements and overall process improvement methods. This can increase the confidence in the software being developed.

Organisations should adopt suitable quality management processes, e.g. ISO 9001:2015, for any activities that may impact on the software/CEH development.

[10]To ensure suitable software versioning etc. The software version is also a fundamental enabler to demonstrate that evidence is valid for the version of the software in the safety argument.

[11]To ensure that any credit for in-service use is attributed to the correct software build.

An additional factor when assessing evidence for organisational competency is the pedigree and capabilities of the organisation in producing safety-critical software/CEH. A view on the pedigree and historic capabilities can give additional weight when assessing staff competencies and their experience.

7.1.10 Existing Certification/Qualification Evidence

Pre-existing qualification evidence from recognised bodies such as EASA or the FAA can prove useful. The issuing of a TSO for a particular part signifies that the design and production has been approved by the FAA. Part of the TSO process is to gain a level of confidence in any development activities that have been conducted by the equipment vendor.

The MOD has recognised a number of bodies for mutual recognition (see sub-section 6.1.1.3), such as the US Army Airworthiness Authority[12]. If there is an agreed mutual recognition then information should be gathered on the TSO, or similar form of approval, to understand its relevance and suitability to support the item under consideration. A TSO, or similar form of approval, which confirms full life-cycle conformance may require additional evidence; however, it *may* also reduce the level of effort for any direct life-cycle assessment process.

7.1.11 Product Service History (PSH)

It could be argued that the strongest form of evidence provides confirmation or confidence in any prior established beliefs, e.g. as stated by McDermid (1998). Current assessments of life-cycle evidence are a means to establish a prior-belief, e.g. that the software provides suitable confidence for compliance to a DAL. PSH can act as a method to further support this premise. By gaining confidence via PSH, e.g. the review of the error reporting processes, confidence can be established in any prior belief placed in the life-cycle. There must be continual monitoring of any system to maintain the belief in any prior confidence (McDermid, 1998). This can be achieved by PSH arguments and/or establishment of a method to conduct error reporting analysis. Figure 7.7 illustrates the relationship between the development and in-service phases of a system.

The methods to judge the suitability of PSH is subjective; there is a recognised and adopted guideline: CAST 1 (CAST, 1998). However, the guideline is open to interpretation with potential variances in the acceptable evidential thresholds. It acts as a means to establish a common language to assess confidence and to compare PSH assessments. EASA (2012) attempts to apply a quantitative figure to the level of hours required to support the

[12]See the following for further information: http://maa.tools.mod.uk/linkedfiles/20140512-maa_mutual_recognition_amrdec.pdf.

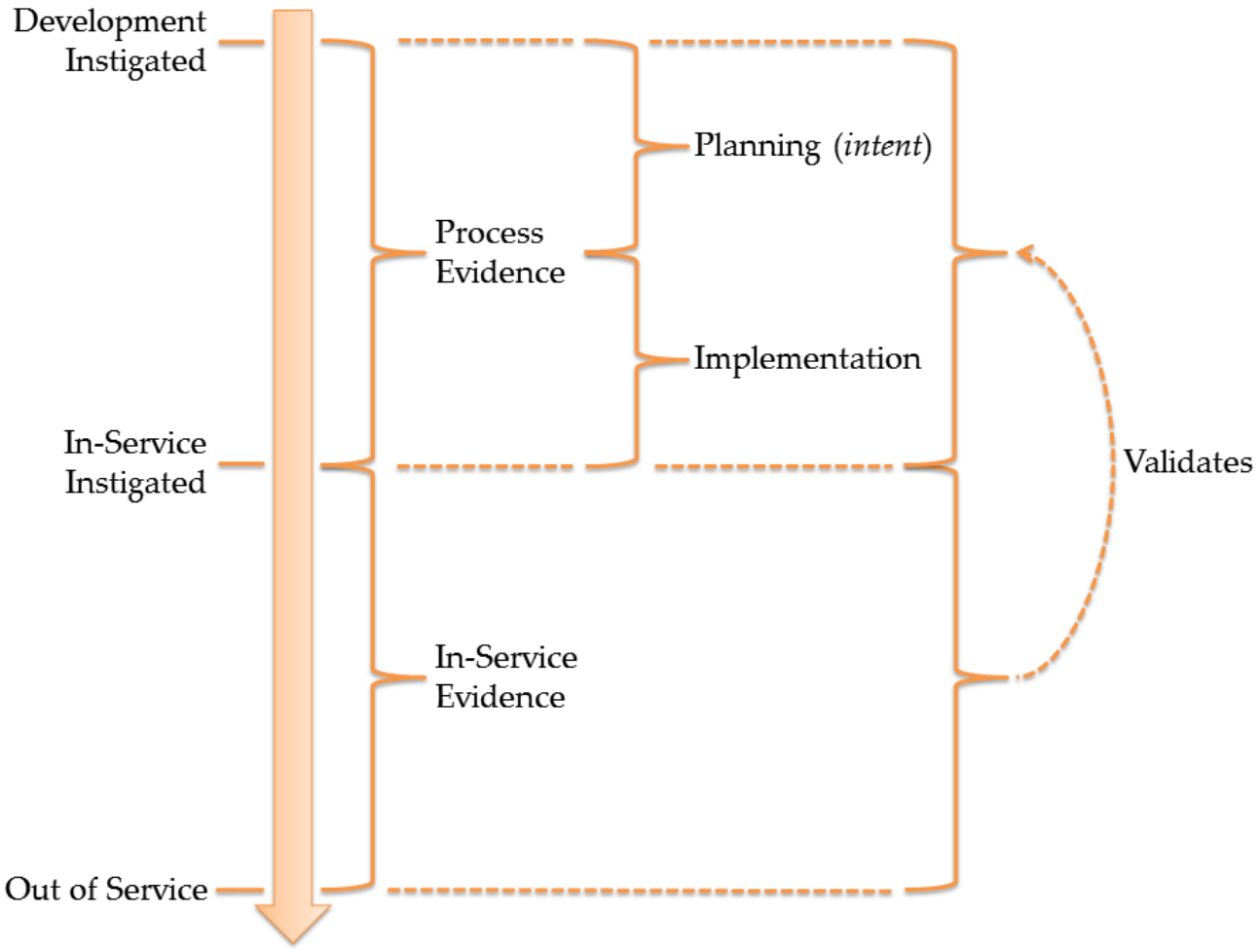


Figure 7.7: Relationship Between Development and In-Service Phases

DALs of a component. The EASA guidance also illustrates that there is an acceptance to mix the PSH from a number of domains to support an in-service argument of a specific LRU, for example. However, the reasoning for the quantitative values is not clear and is open to debate with SMEs. Table 7.1 provides an EASA opinion on the *sufficient* PSE required for DALs A-C.

The applications of use stated within Table 7.1 have the following definitions:

- *Aircraft.* Aircraft operation in flight or on ground and board/LRU/system/aircraft tests.
- *Safety.* Space, airborne military, nuclear, medical, railway, automotive.
- *Other.* Bank, computer, telecom, etc.

The benefit of any PSH is also dependent on the context of *how* the software/CEH is

Design Assurance Level (DAL)	**Product Service Experience (PSE)**
DAL A	*Sufficient* PSE if: • At least 2 years of use **with** [hours of aircraft applications + safety applications] $> 10^6$. • At least 2 years of use **with** [hours of aircraft applications + safety applications] $> 10^5$ **AND** [hours within other applications] $> 10^7$.
DAL B	*Sufficient* PSE if: • At least 2 years of use **with** [hours of aircraft applications + safety applications] $> 10^5$. • At least 2 years of use **with** [hours of aircraft applications + safety applications] $> 10^4$ **AND** [hours within other applications] $> 10^7$.
DAL C	*Sufficient* PSE if: • [hours of aircraft applications + safety applications + other applications] $> 10^5$.

Table 7.1: DAL and Associated PSE Requirements (based upon EASA (2012))

used. Adhering to the design envelope of the system allows data to be collated from an anticipated set of environment variables. A security context is different due to the evolving level of sophistication and the types of threats to a system/software. Therefore, PSH would have more validity in the context of providing *safety* assurance rather than *security*. The frequency of the software/CEH function usage is also important and this is considered within sub-section 7.2.

7.1.12 Reliability Modelling

The aim of reliability modelling is to predict future performance and it can be determined via two broad types of modelling. The first is based upon data collected during the *development* effort, i.e. before delivery of the software/CEH to the customer. The second method uses data based upon *faults* captured once the software/CEH has been provided to the customer. The collection methods to do this could be via end-users reporting the errors or via routine reviews of software fault logs. In both instances (post- and pre-delivery of the software) the aim is to ascertain a *prediction* of the future behaviour of the software/CEH. The models can be implemented via a range of methods such as Binomial-Type[13], Poisson-Type[14], etc.

A number of metrics should be captured before a suitable reliability prediction can be

[13]Binomial-Type models assume that there is a fixed number of faults remaining in the program at any given time (Musa, 1987).

[14]Poisson-Type models *do not* assume that there is a fixed number of faults remaining in the program at any given time (Musa, 1987).

made. Product metrics, e.g. SLOC, would be an input within any model based upon the software prior to a customer delivery. However, such metrics would still be establishing a *prior-belief* in the development effort and may be problematic to obtain from the vendor, e.g. due to data access limitations. Post-delivery the metrics and models would be more likely to be captured and would be based upon data which is established by PSH.

It should be noted that the issue of metric collection is contentious as it can influence system behaviour detrimentally. This can impact other areas within the software domain such as software sustainment. See sub-section 7.3 for further information on the merits and dangers of using metrics.

7.1.13 Security Considerations (in Relation to Airworthiness)

The topic of cyber security in relation to airworthiness became a key consideration for the MOD with the up-issue of the DS 00-970 Part 13 to Issue 11 (UK MOD, 2014*a*). This introduced the consideration of cyber assurance to airworthiness. Prior to the DS 00-970 up-issue the concept of cyber-assurance was conducted via other forms of analysis. In essence, the up-issue stated that for airworthiness related cyber security the guideline DO-326A[15] (RTCA, 2014*a*) should be adopted as an AMC.

The DO-326A approach aims to embed the consideration and resolution of any security-related issues through the development of the software/CEH product. The processes adopt an approach similar to that of ARP 4754A (SAE, 2010) with efforts to identify risks[16] to the verification of the adopted approaches[17]. The security related elements to airworthiness, and the subsequent assurance, are highly related to that of safety and therefore this lends itself to adopting a similar safety and security assessment process. Figure 7.8 shows the link between the *safety* and *security* processes.

Other forms of AMC may be required in-lieu of direct DO-326A evidence due to the different methods which vendors adopt to address airworthiness related cyber security. Suitable levels of confidence can be achieved, for example, by the review of the software architecture for relevant assurance features or direct static analysis of the source code (Hadley and Standish, 2017). Additional activities could cross-reference the DO-326A objectives to a representative Risk Management Framework (RMF) such as the National Institute of Standards and Technology (NIST) standards; e.g. 800-53 (NIST, 2013) and 800-37 (NIST, 2018). The results of such an approach can be found within Lennon, Standish and Hadley (2018).

[15]DO-326A. Airworthiness Security Process Specification.
[16]Via various methods such as preliminary system security risk assessments.
[17]Within a system security verification context.

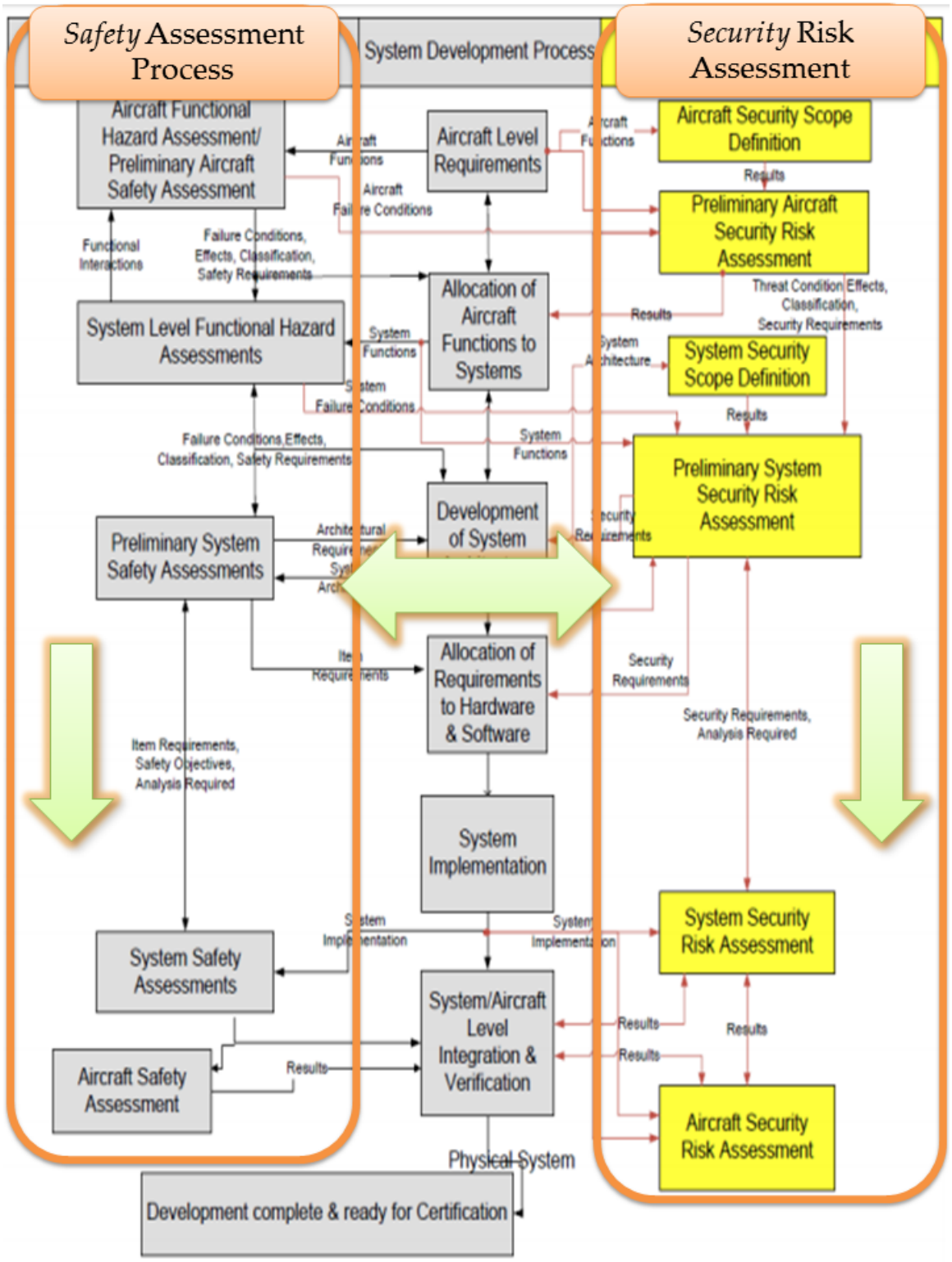


Figure 7.8: Airworthiness Security Process as Part of the Aircraft Certification Process (adapted from Paul et al. (2016))

7.2 Underpinning Principles for the use of Evidence

The following sub-sections provide information on a number of fundamental principles which need to be considered when any diverse evidence is to be used for a software and/or CEH assurance argument. Figure 7.9 shows the potential underpinning principles to be adhered to.

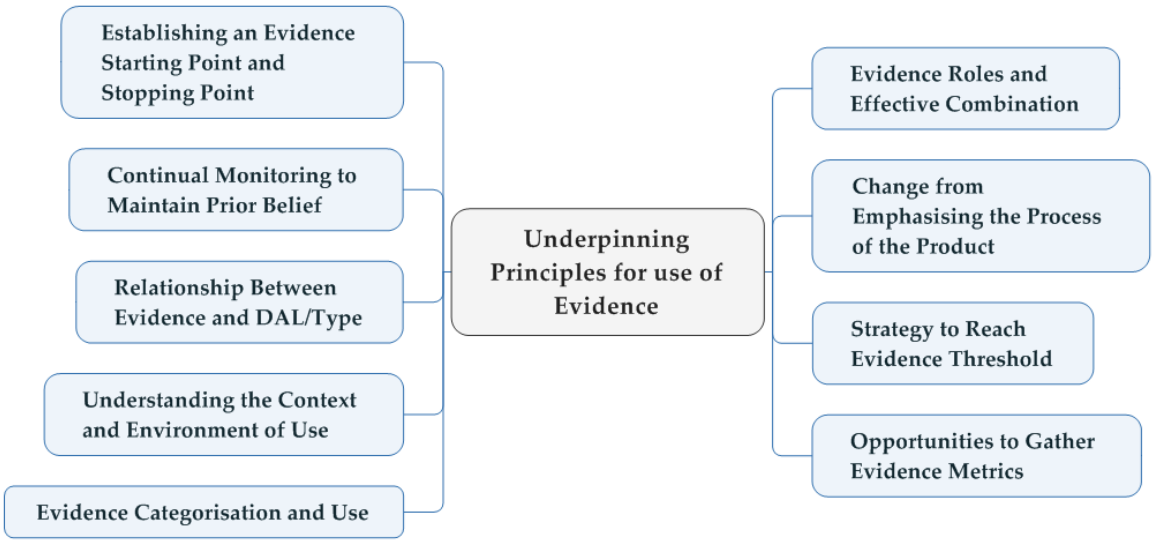


Figure 7.9: Underpinning Principles for the use of Evidence

7.2.1 Establishing an Evidence *Starting Point* and *Stopping Point*

Evidence *starting* and *stopping* points can establish the *validity* of any evidence, e.g. the evidence which is supported via in-service hours. Any significant software or CEH modifications to an existing development may result in the in-service starting point being *reset*. This *reset* may occur if key aspects of the software or CEH[18] have undergone a change which results in not being able to justify the continuation of the in-service data calculations. Figure 7.10 shows (a) the continued credit for in-service hours due to *insignificant* software changes and (b) the impact of *significant* software changes with the reduced credit for in-service hours.

There are a number of considerations for the evidential *starting* and *stopping* points. It is not always realistic to expect evidence to be fully compliant to make a full assurance judgement. The challenge is to understand a sufficient level of evidence and how any potential *primary* evidence can be reinforced with other strands. This will support the decision on

[18]The key aspects may be measured via the amount of change to SLOC or the *significance* of the software functionality.

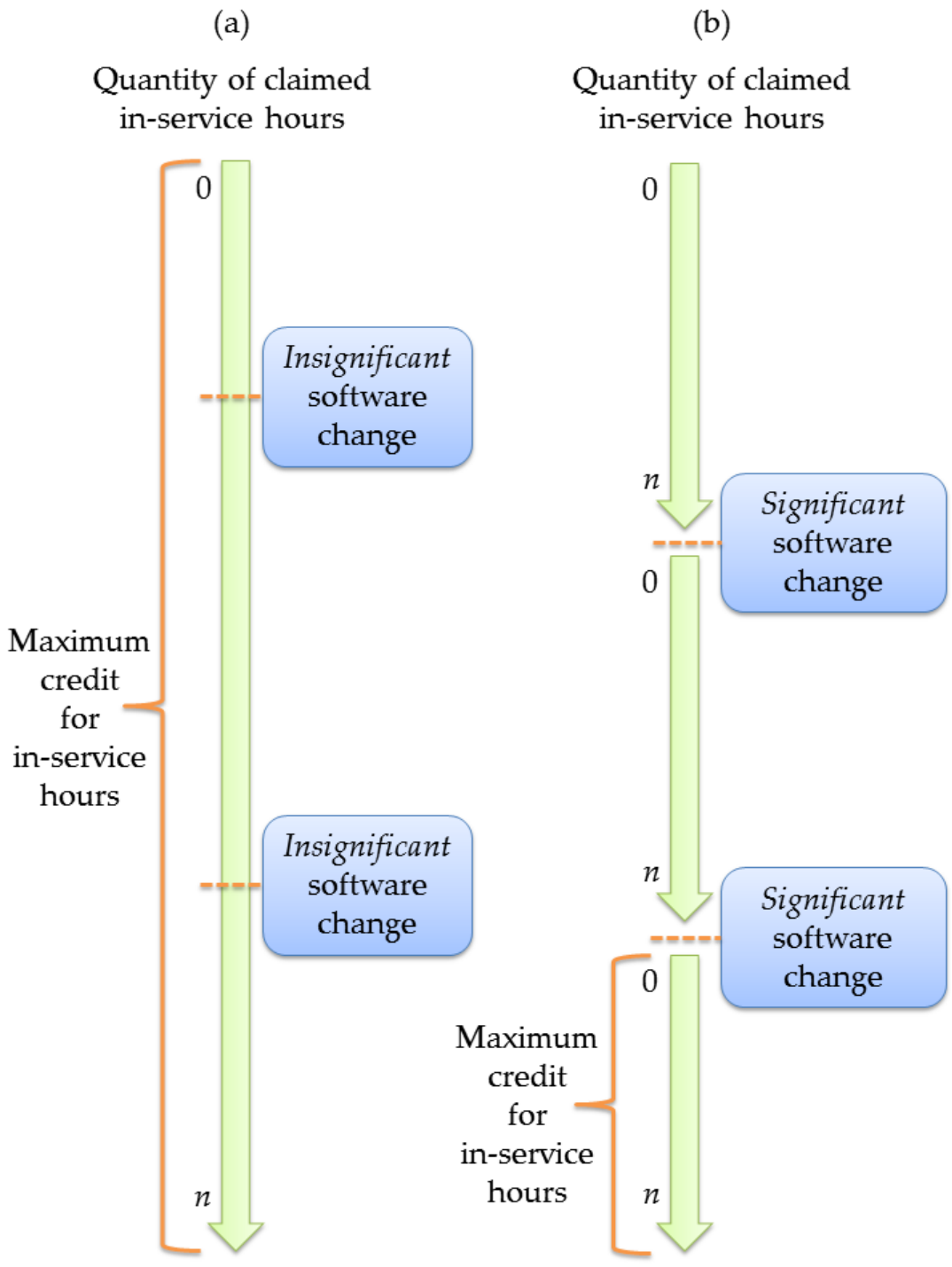


Figure 7.10: Impact of Significant Software Changes on PSH In-Service Hours

whether to collect further evidence of the same type or to select alternative evidence to reach the assurance confidence threshold.

7.2.2 Continual Monitoring to Maintain Prior Belief

Any evidence gathered and assessed prior to the in-service use of the system is, in essence, establishing a *prior belief*. This prior belief is based upon process evidence, testing, and reliability modelling which is focussed on the development activities. However, it is important to gather evidence via continual monitoring to maintain and support any prior belief arguments that have been established. In addition, Hadley and White (2008) claim that statistical testing allows development test data to be *quantified* to support a robust argument for the reliability of the software and hence of the software safety.

The use of PSH, or operational data, is a further method to 'validate' the prior-belief, see Figure 7.7. This validation should be a planned and continuous activity. This premise is supported by McDermid (1998) and Hadley and White (2008)[19]. Any monitoring should be fed back into any reliability modelling which is being conducted for the through-life assessment of the software.

7.2.3 Relationship Between Evidence and Type/DAL

A number of the evidence types, e.g. PSH, and underpinning principles, e.g. continual monitoring, are applicable to both software and CEH. This is due to both being developed via a life-cycle to establish a prior belief with the in-service period being able to *validate* the belief. Evidence specific to the software and CEH will need to be established, e.g. the safety assessment process information. The evidence may have elements which have direct read-across between the software/CEH, e.g. in-service hours. Alternatively, inferences may be made from the software/CEH evidence, e.g. system integration testing, (although having a focus on software confidence could allow CEH failures to be observed).

The evidential types and the underpinning principles are valid for software/CEH for any DAL or safety impact level. The same types of evidence can be used to gain confidence in a safety-critical context, i.e. DAL A, and that which has a very limited safety impact, e.g. DAL D. An example is a SDP with the threshold rigour for the evidence needing to be in keeping with the safety impact level.

[19]It is essential that the evidence collected can be shown to be derived from the in-service product rather than a pre-production model.

7.2.4 Understanding the Context and Environment of Use

For a 'brownfield' development and those subject to periodic refreshes the software may not have been developed to known processes or have known safety-related properties. This type of software is known as Software of Unknown Pedigree (SOUP). If the software has been developed to a *known* development processes it may meet the new requirements in the extant guidelines. DO-178C refers to this as Previously Developed Software (PDS) and the expectation is that the DO-178C guideline would be used to close any gaps between the PDS and the new assurance requirements. Other evidence may be sufficient to close the shortfall.

Systems, especially at a platform level, are complicated with various supporting sub-systems. Each have their own development and in-service history. The sub-systems should be considered independently to understand their own evidential provenance. They must also be considered as an integrated whole to assess the interfaces and the environments in which the system will operate.

It is important to understand how the sub-systems operate and the context in which they are used. As an example, some systems such as a flight management system will operate on a *continual* basis as the functionality is required throughout the stages of flight. Whereas, other systems may only operate *on–demand* and only during certain flight stages, e.g. landing. This concept is illustrated simplistically within Figure 7.11 which shows the phases of flight for (a) Weight-on-Wheels (WOW) functionality and (b) continuous engine power-on functionality. The theory is that the software in continuous use can claim a greater level of confidence due to it being exercised more frequently. The on-demand and continuous use considerations will impact the confidence claimed for the evidence, e.g. the level of PSH.

7.2.5 Evidence Categorisation and Use

There has been a range of literature which has attempted to establish a broad categorisation of evidence. McDermid (1998) proposed that evidence could be generally described as one of three forms of evidence:

- *Direct.* Evidence which is quantified evidence from testing, operational experience or analysis indicating that the software meets its safety targets.
- *Backing.* Indirect evidence that shows that the direct evidence is sound (e.g. that historical data has been kept properly, and that it reflects the future operational environment of the software).
- *Reinforcement.* Indirect evidence which enables arguments to be made to claim failure rates beyond those which can be evaluated through direct means.

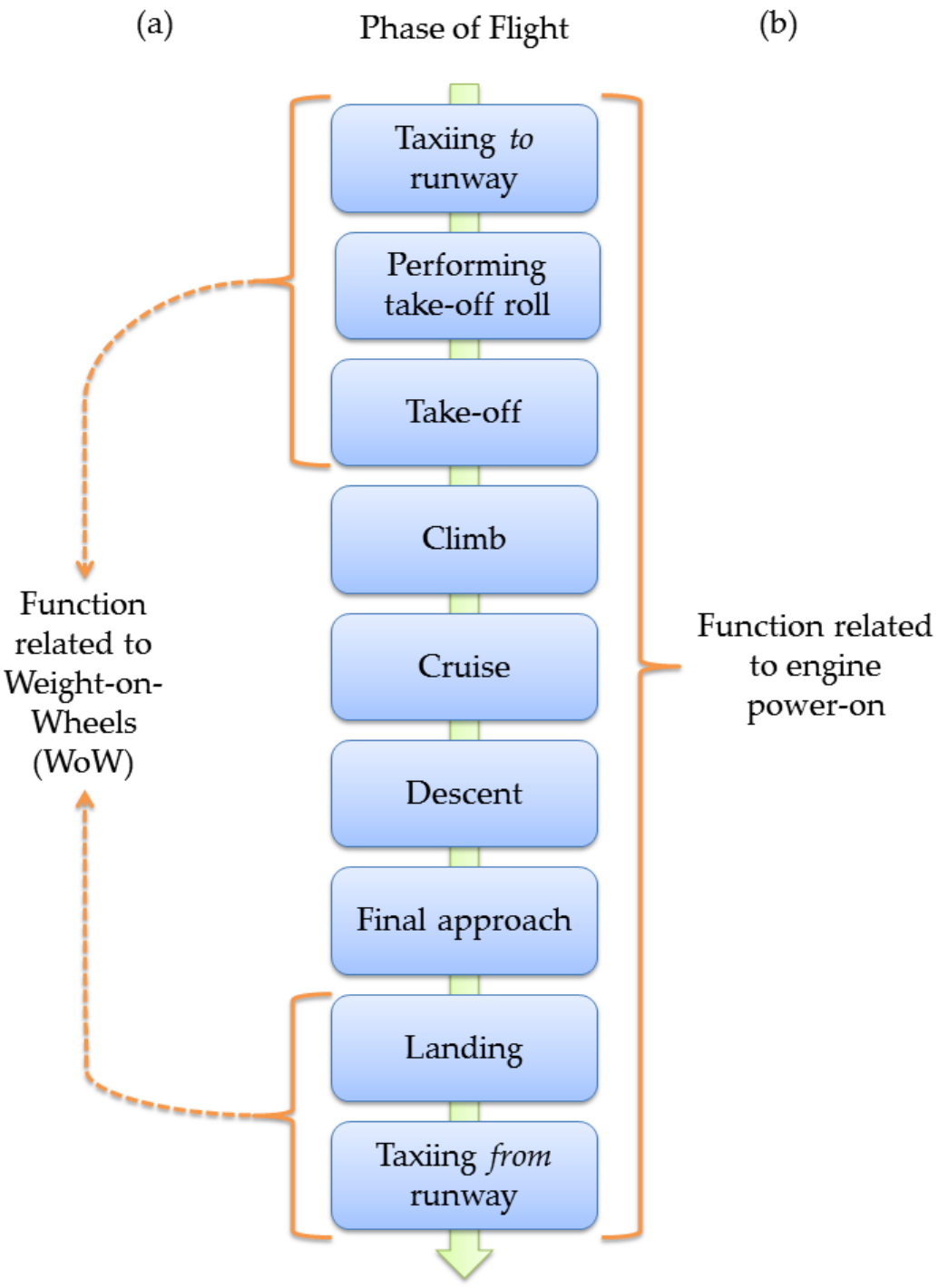


Figure 7.11: Relevance of Functionality During Different Phases of Flight

Caseley, Tudor and O'Halloran (2003) proposed that evidence could be one of four forms:

- *Process.* In essence, process evidence is described as an indirect qualitative measure, e.g. software developed by a specific technique.
- *Historic.* Historic evidence is that which is based upon failure date, maintenance records, or such techniques as reliability modelling.
- *Test.* Test evidence is based upon a quantitative measure of the product, e.g. qualification testing.
- *Proof.* Proof which is a direct mathematical qualitative measure of the product.

Hadley and White (2008) proposed a modification to the established 'three pillars' of evidence to embed the concept of PSH into the software assurance arguments for an airborne platform. The 'three pillars' were at the time traditionally classed as: (a) process evidence, (b) verification, and (c) validation. The revised 'pillars' were defined by Hadley and White (2008) to be:

- Process.
- Test Processes.
- In-Service History.

Any software safety assurance review, including the use of the underpinning principles, must establish clear distinctions in the evidential types and must judge the weightings of such evidence. How the evidence supports or refutes other forms of evidence should also be considered.

7.2.6 Evidence Roles and Effective Combination

The varying forms of evidence that can be included as part of a diverse assurance argument have differing relationships. These impact the claims that can be supported or refuted. Each evidential strand has fundamental elements; e.g. the requirements elicitation/generation stage is an imperative stage of the development life-cycle. It informs the design and the measurable criteria for the testing. For in-service data, the CM plays a significant role as it establishes software baselines which enable in-service flight hours to be validated. Evidential strands cannot, generally, be considered in isolation. At least two forms of evidence should be provided in support of any claim (McDermid, 1998). Evidence should be provided that

the techniques used are diverse, e.g. one is static and the other dynamic, and thus are likely to identify different limitations in the software and CEH.

As claimed by McDermid (1998) and reinforced within this thesis, there is an argument that for an assurance argument to be made the process evidence should only be utilised where the direct product evidence cannot be provided. In practice this means that a combination of process and product evidence will be required in most cases due to the limitations of gaining direct product evidence.

7.2.7 Change from Emphasising the Process to the Product

[Sub-section text redacted]

Further support is contained within Hadley and White (2008) which states that to understand the value provided by process evidence the authors "know of only a negative measure: the highly qualitative sound-bite that [evidence for] good processes cannot guarantee a good product but [evidence of] bad processes will almost certainly guarantee a bad one". This led to Hadley and White (2008) stating that "good processes are necessary but not sufficient". Robust processes play an important role as they assist with supporting the premise that a "stronger argument is that a hazard has been mitigated by being designed out rather than it has been mitigated by being tested out" (Hadley and White, 2008).

The concept of utilising additional evidence strands is also outlined within a report for an airborne platform DT co-written by the RE and the EngD Industrial Supervisor[20] (Standish and Hadley, 2014). The report cited three review components:

- Software processes.
- Product service history.
- Software reliability claims as sources for additional evidence.

7.2.8 Strategy to Reach Evidence Threshold

The assignment of a DAL to a system will result in an evidence threshold being established to gain a suitable level of confidence. It is not always appropriate to meet the evidence threshold as it may be sufficient to be aware of the evidential gap and to understand any risks. If the risk is understood it can be *tolerated* or *mitigated*. There may be no further action if a balanced assessment of the impact determines that the risk can be tolerated. The challenge is to be aware of the evidential gap and to make an informed decision.

[20]Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems.

Technical and procurement avenues will need to be explored to reach the evidence threshold. The evidence would not be solely based upon the cost impact. Where the risk associated with the evidential gap it too great or cannot be mitigated then there may be a need to *buy knowledge*. This provides further evidence to reach an appropriate evidential threshold.

[Sub-section text redacted]

An alternative approach is to ensure that the assurance requirements and the technical design/solution achieves a *balance*. This maximises the compliant assurance evidence available to be judged whilst implementing a feasible technical solution of value for the capability. Such an approach was proposed by the RE and the EngD Industrial Supervisor[21] with a stepped process for MC processor qualification[22].

The concept of assurance/capability balance is simplistically illustrated within Figure 7.12 which shows:

- (a) The assurance requirements limiting the technical solution (and a potential reduction in capability).
- (b) The technical solution limiting the assurance confidence via design considerations which are not commensurate with the defined qualification approach.
- (c) An imbalance between the assurance requirements and the technical design/solution so that an unsatisfactory capability is delivered.
- (d) A balance between the assurance requirements and the technical design/solution so that an achievable capability can be delivered.

7.2.9 Utilise Opportunities to Gather Evidence Metrics

Throughout the development phases of software or CEH there are opportunities to gather suitable metrics. These can act to provide confidence in a particular system, or equally important, to provide counter-evidence. Traditionally, the implemented development stages are compared to relevant standards to judge compliance and therefore to ascertain the level of confidence in the development. However, throughout the development stages there are additional metrics which can bolster the assurance confidence, e.g. results from independent reviews. As an example, Hadley and White (2008) propose that the number of technical comments generated from a Technical Interface Meeting (TIM) can be assessed to measure how many of the comments lead to *actual* changes. This would ensure that the significance

[21] Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems.

[22] As illustrated within the research output titled *Multi-Core (MC) Processor Qualification for Safety Critical Systems*.

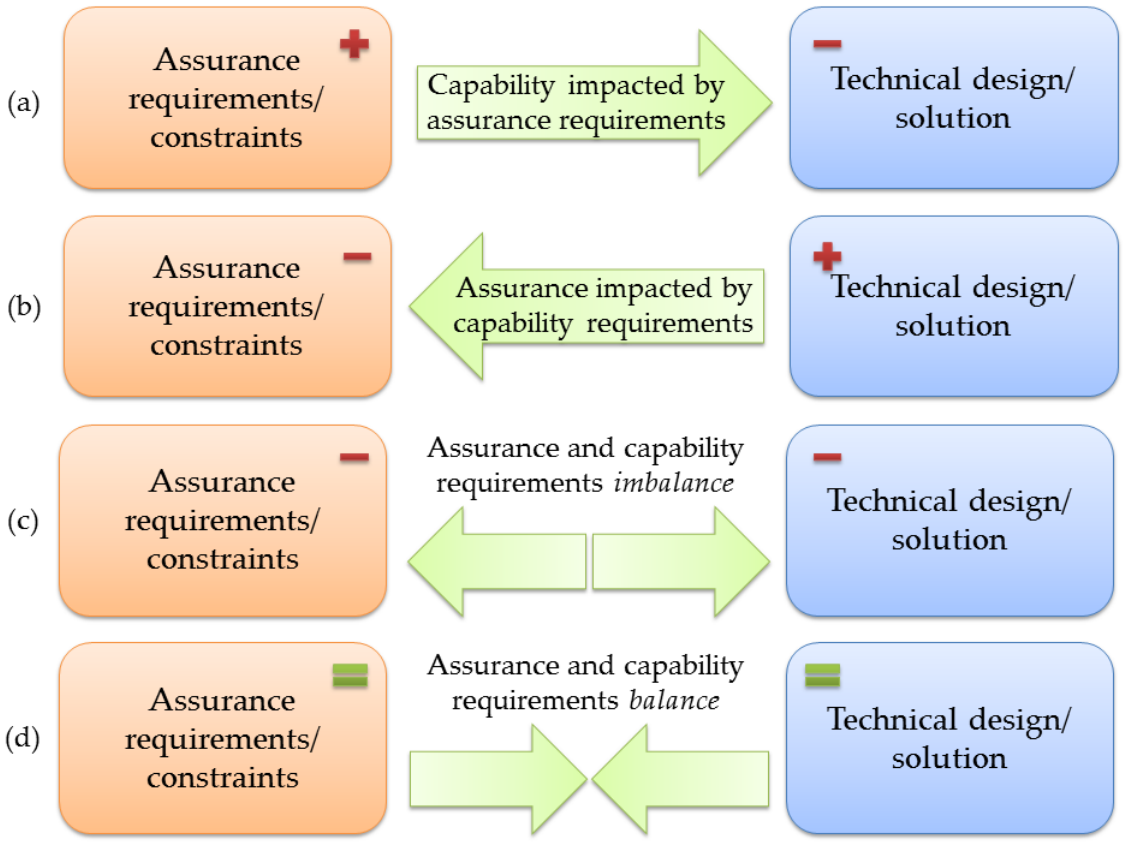


Figure 7.12: Trade-off Between Assurance Requirements and Technical Capability

and validity of the comments are measured rather than the number of comments made. Such a measure could also be applied at the formal verification stages.

Gaining evidence/confidence throughout the stages of the platforms development and in-service period is predicated on suitable metrics being identified, justified, and recorded. See sub-section 7.3 for further information on the merits (and dangers) of the use of metrics.

7.3 Importance and Unintended Consequences of Metrics

Metrics can provide valuable information to decision makers and those that wish to draw conclusions from the data. However, metrics that are not chosen well can have negative consequences if incorrect decisions are made on the results. They can also lead to incorrect interpretations which then inform a safety assurance argument.

The research output *The Measurement of Software Maintenance and Sustainment: Positive Influences and Unintended Consequences*[23] illustrate the unintended consequences of incorrect metrics in the context of the software supply chain. The particular focus of the output is on Performance Based Logistics (PBL) which is commonly implemented for large and complex software systems, e.g. F-35 Lightning II (Huff and Novak, 2007). The information within this section is based upon the journal output.

7.3.1 Potential Unintended Consequences

Smith (1995) is concerned with the unintended consequences of publishing performance data for UK public sector organisations and the lessons are relevant to a PBL. Smith (1995) states that the paper's findings can assist with understanding how performance data can play a *significant* part in guiding the activities of an organisation. Smith (1995) highlights a number of negative unintended consequences of using performance data to influence system behaviour:

- *Tunnel vision.* When management focuses on quantified aspects of performance rather than overall quality.
- *Sub-optimisation.* Where narrow, local objectives are prioritised over the wider objectives of the organisation as a whole.
- *Myopia.* Which involves the pursuit of short-term targets at the expense of legitimate long-term objectives or outcomes.

[23]Written by the RE and a Dstl colleague: Rob Ashmore - Dstl, Senior Fellow.

- *Measure fixation.* Where managers focus on the metric, rather than the objective for which the metric was developed.
- *Misrepresentation.* Where the reported metrics do not match the behaviour on the ground.
- *Misinterpretation.* Where those to whom the metrics are reported make incorrect or inappropriate decisions.
- *Gaming.* Where behaviour is deliberately altered to exploit loopholes in the measurement system.
- *Ossification.* Where an overly rigid measurement system prevents innovation.

7.3.2 Concept of Technical Debt

Within the research output the concept of *technical debt*[24] is used to illustrate how the measurement of performance data can lead to unintended consequences such as those highlighted by Smith (1995). There are perceived benefits to incurring technical debt, for example it allows a new software release to be produced sooner than otherwise would be the case. However, this usually comes at a longer-term cost, as indicated by the *debt* metaphor. In particular, as this level of debt grows, it becomes more difficult to make changes, slowing down future releases. Ultimately, an unchecked growth in technical debt is likely to shorten the lifespan of the software, hastening the need for its replacement.

7.3.3 Mitigations to the Risk of Unintended Consequences

There are several strategies that can be used to mitigate the risk of unintended consequences. However, the most comprehensive mitigation strategy involves gaining a system-level understanding of the process that is being measured and using that understanding to identify likely responses to different measurement choices. The system-level understanding should also be used to monitor the measurement-induced effects so that, if necessary, corrective action can be taken.

The key conclusion is that any proposed set of software maintenance and sustainment metrics should be accompanied by the following:

- A system-level description of the process that is being measured.

[24]Technical debt refers to code that is known to be "not quite right" but a decision has been made to postpone making it right.

- A description of how the metrics are intended to influence the system toward the desired behaviour, including how they might interact to generate unintended consequences.
- An explanation of how the risk of unintended consequences will be mitigated. This should include a description of how the effects introduced by the metrics will be monitored and how the selection of metrics will be altered if necessary.

Although the research output has a focus on software maintenance the principle of being cautious with the use of metrics still holds. Any measure of assurance must take into account the unintended consequences which could arise. Metrics adopted to make judgements must be used with an understanding of their limitations[25].

7.4 Communicating Evidence with Stakeholders

7.4.1 Principles to Allow Understanding

Whether small or large levels of data are being assessed, there is a need to make sense of it so that it can be interpreted. As Nussbaumer-Knaflic (2015) states: being able to visualise data and tell stories with it is key to turning it into *information* that can be used to drive *better* decision making.

The science and philosophy of appropriate visualisations is not within the scope of this thesis; however, a number of *rules* have been applied to create a visualisation. Kirk (2016) provides guidance for getting a balance between various attributes which a visualisation could possess. Understanding the attributes and the aim of a visualisation allows the message to stakeholders to be articulated clearly and with purpose.

Kirk (2016) also provides a definition for data visualisation: the representation and presentation of data to facilitate understanding. This concept is further defined with three stages of understanding:

- *Perceiving.* What does it show?
- *Interpreting.* What does it mean?
- *Comprehension.* What does it mean to me?

[25]There may also be unintended consequences of adopting diverse evidence for qualification purposes on the actual software development processes themselves. An example is with *gaming* due to a perception that diverse evidence will mitigate any deficiencies anyway. However, such a study is considered out of scope of the current research.

These principles are key and they are enforced via good visualisation design principles, such that a design should be: *trustworthy*, *accessible*, and *elegant*. With the Kirk (2016) principles (and the underpinning attributes) and further guidance obtained from Nussbaumer-Knaflic (2015) a concept visualisation was created (see next sub-section). The visualisation was generated by the RE and the EngD Industrial Supervisor[26] to assist with engaging stakeholders effectively. The visualisation is proportionate in portraying the right level of information and sufficient to allow stakeholders to be informed to make decisions.

The adoption of diverse evidence provides decision makers with an increased *solution space* to inform the assurance confidence. This increased space could lead to issues for the: (a) *perception*, *interpretation* and *comprehension* of the evidence, (b) the source of the evidence, and (c) how it relates to specific LRUs. Figure 7.13 illustrates (a) the process-based review of evidence and then (b) with it being a subset within the increased solution space containing the wider sets of diverse evidence. However, the increased solution space can lead to challenges with understanding and comprehending suitable evidence.

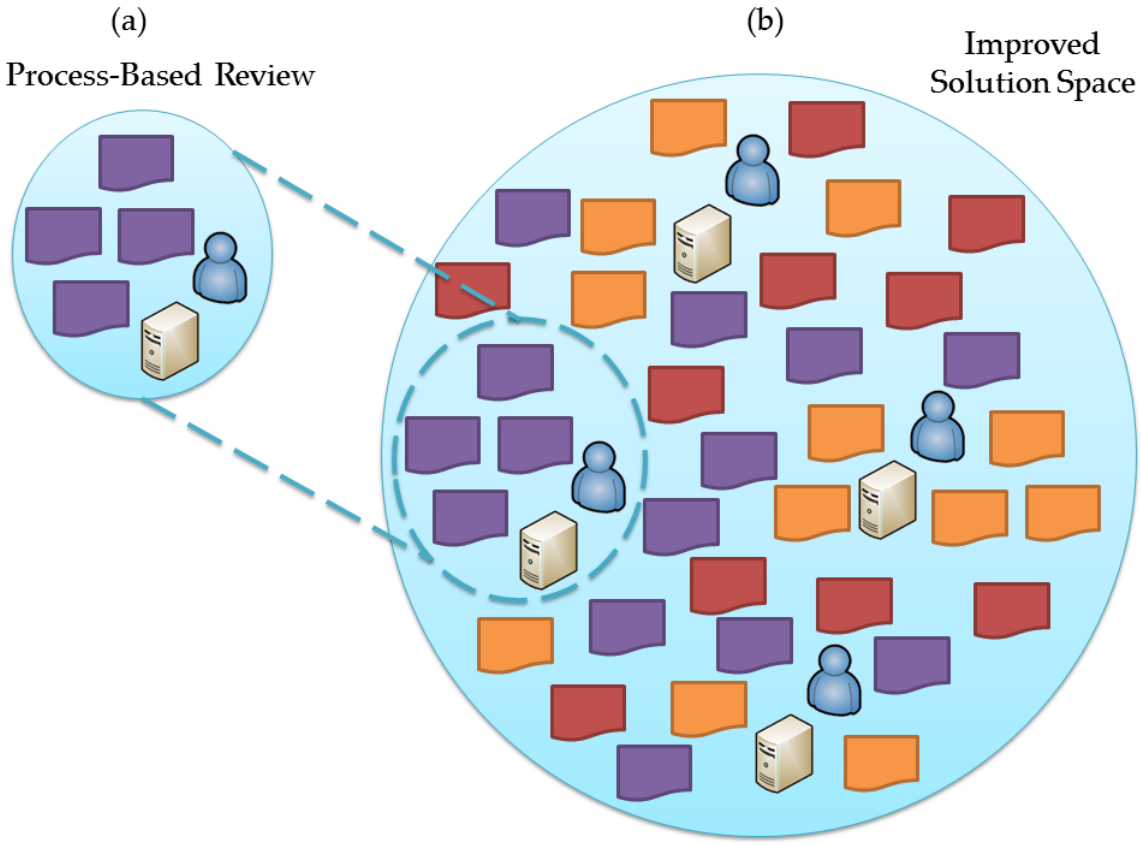


Figure 7.13: Adopting Diverse Evidence Expands the Solution Space

[26]Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems.

7.4.2 Wheel of Qualification: A Model to Assist Understanding

7.4.2.1 Wheel of Qualification: Overview

A visualisation was created which was termed the *'Wheel of Qualification'* by the RE and EngD Industrial Supervisor[27]. The visualisation aimed to allow informed dialogue with a number of stakeholders (e.g. MAA, DT etc) who require an insight into the software/CEH evidence for the individual LRUs of a platform. The 'Wheel' consists of a number of tiers (or layers):

- *Tier 1 (the outer layer).* Evidence associated with an active ITE which provides judgements on additional activities which may be undertaken by mutually recognised bodies (e.g. FAA) or via a Coordinating Design Authority (CDO) (e.g. V&V testing)[28].
- *Tier 2 (the middle layer).* Evidence associated with a DO which undertakes integration activities or develops software for LRUs[29].
- *Tier 3 (inner layer).* Traditional core elements which would be assessed as part of a DS 00-970 review process with additional product evidence (PSH) included[30].

The 'Wheel' consists of *sections*, e.g. Domain Awareness. Each section contains a number of *segments* which represent the associated LRUs for that section, or evidence[31]. As an example, each section within Figure 7.14 has 7 segments to represent each of the LRUs (or sub-systems) of interest for the platform. Those LRUs (or sub-systems) which are relevant for the evidential item are coloured appropriately, e.g. evidence which is relevant for *LRU1* is coloured *dark blue*. LRUs which are not suitable/valid for the evidence have a segment coloured *light grey*.

For example, the 'Wheel' within Figure 7.14 shows the availability of the evidence for *LRU3*[32] within tier 3[33].

- PSH: In-service data is available for *LRU3* as the PSH section has a *pink* segment within it[34].
- Software Life-Cycle (Process): Software life-cycle data is available for *LRU3* as the section has a *pink* segment within it[35].

[27] Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems.
[28] *Tier one* is shown as (a) within Figure 7.14. The figure contains the abbreviation: Sub-System (SS).
[29] *Tier two* is shown as (b) within Figure 7.14.
[30] *Tier three* is shown as (c) within Figure 7.14.
[31] *Segments* are shown as (d) and a *section* shown as (e) within Figure 7.14.
[32] Coloured *pink* and shown as (f) within Figure 7.14.
[33] Traditional core elements - coloured *green* and shown as (c) within Figure 7.14.
[34] Shown as (g) within Figure 7.14.
[35] Shown as (h) within Figure 7.14.

- CEH Life-Cycle (Process): CEH life-cycle data is *not* available for *LRU3* as the section has a *light grey* segment within it to replace that of *LRU 3*[36].

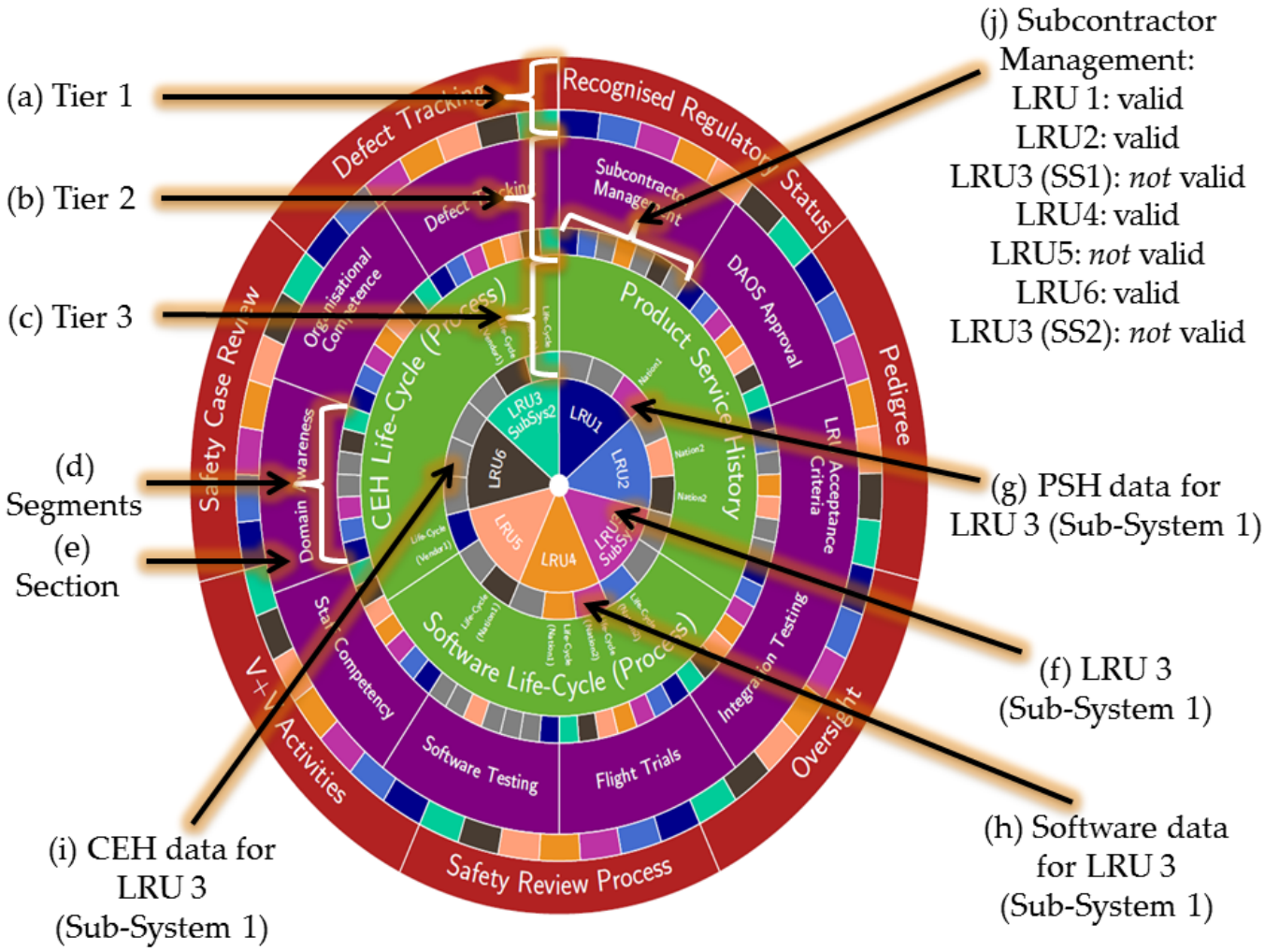


Figure 7.14: Elements of the 'Wheel of Qualification'

This representation is repeated throughout the 'Wheel' with each section detailing the LRUs (or sub-system) as segments which the evidence relates to. As a further example, "Subcontractor Management" (within tier 2 - coloured *purple*) is *valid* for the LRUs (or sub-systems): *LRU1*, *LRU2*, *LRU4*, and *LRU6*. "Subcontractor Management" is *not valid* evidence for *LRU3* (sub-system 1), *LRU3* (sub-system 2) or *LRU5* as the associated segments are shown as *not valid/available* (coloured *light grey*)[37].

The legend to the 'Wheel of Qualification' is shown in Figure 7.15 with a full example of the 'Wheel' visualisation within Figure 7.17.

[36]Shown as (i) within Figure 7.14.
[37]The example is shown as (j) within Figure 7.14.

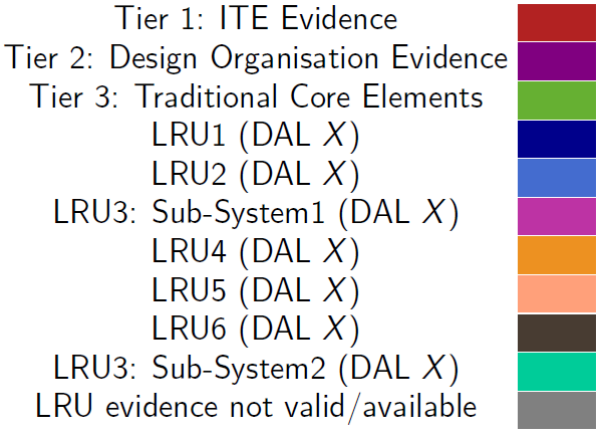


Figure 7.15: Example of a 'Wheel of Qualification' - Legend

7.4.2.2 Wheel of Qualification: Benefits of the Visualisation

The adoption of a visualisation model to communicate information has proved to be exceptionally useful as part of the REs activities with the sponsoring organisation.

For a diverse and layered software/CEH argument there is a tendency to have a number of evidential items which have varying degrees of relevance to the system. As an example, a system integrator using third-parties to develop software would mean that the system integrator's *own* development and software testing regime would *not* be of relevance. Software developed in-house by the system integrator for specific systems would result in the development and software testing regime *being of relevance* and part of a safety argument. The 'Wheel' allows a clear picture to be gained of the important relationships between the systems and the relevant evidence. This can show which systems are currently *weak* in evidence and the particularly important evidence can be made obvious.

The visualisation helps to hide the complexity of the assurance activities to assist engaging with multiple stakeholders who have an interest in the qualification approach. This allows the Kirk (2016) stages of understanding to be adopted with stakeholders able to gain a *comprehension* by determining what the system and evidence relationships mean to them. The key systems (LRUs) of interest can be viewed by Project Managers (PMs), e.g. for contracting purposes. Also, key evidence and the sources of the evidence can be ascertained. This is useful for PMs and also for roles which hold the assurance risk as they can view the *totality* of the evidence.

Simplifying the *complicated* solution space provides the ability to holistically view: the evidence, the evidence sources, the systems, and the relationships. Stakeholders can make informed decisions on the *evidence* to be gathered/generated and to interpret the *consequences* of any shortfalls. The visualisation has allowed stakeholders to request further evidence in

support of a particular LRU which has provided additional safety assurance confidence. Figure 7.16 shows the benefits of the 'Wheel of Qualification' moving from (a) a *complicated* view of the solution space to (b) an *elegant* representation of the data.

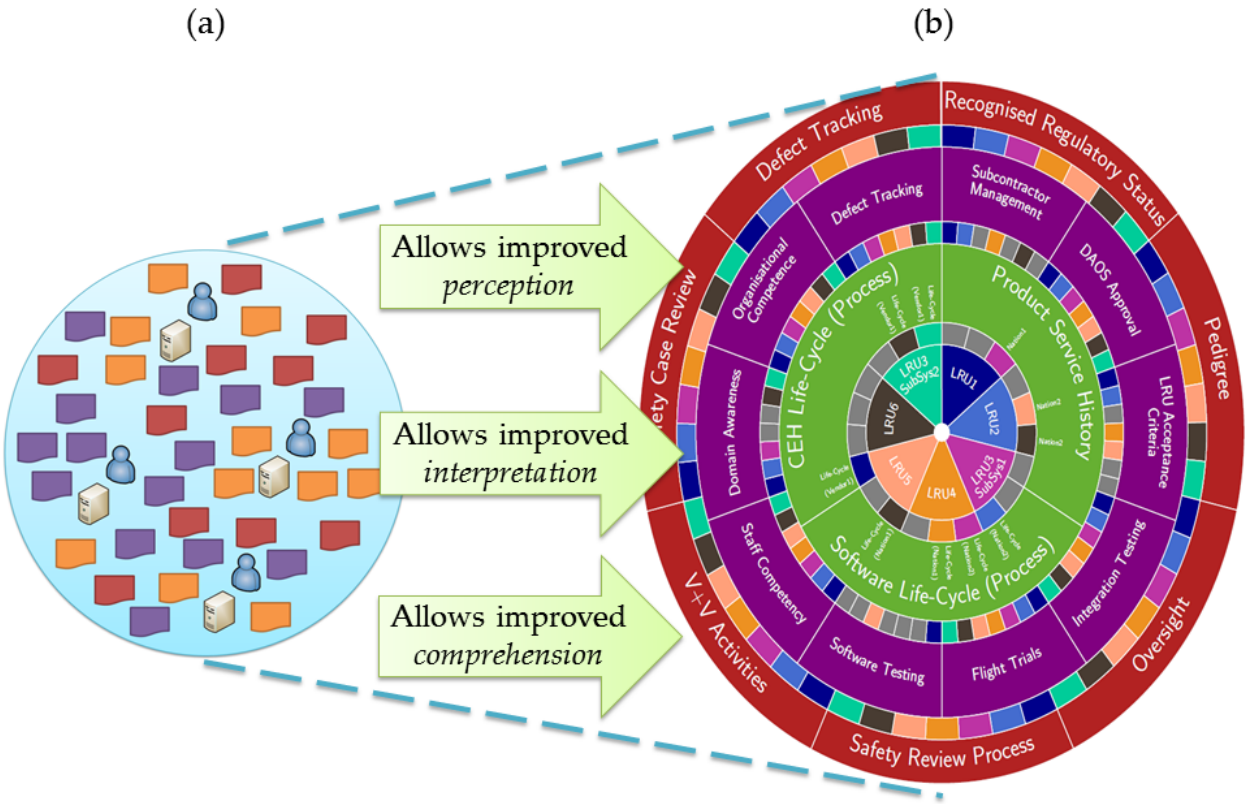


Figure 7.16: The 'Wheel' Simplifies the Visualisation of a Complicated Solution Space

An example of the use of the 'Wheel' has been when a lack of process-based evidence (which supports Tier 3) for a particular LRU required stakeholder engagement to understand the wider assurance evidence needing to be gathered. For the particular project within this example the LRU lacked direct evidence to support making a full process evidence based claim. The 'Wheel' was presented (including the project specific LRUs of interest) within a dedicated software assurance workshop with attendees such as equipment purchasers, SMEs, and ITEs. The 'Wheel' allowed the necessity of the wider evidence to be exposed by highlighting the absence of process-based evidence and illustrating *what* evidence could inform which LRUs. It showed the direct links that could be established between the LRU and the Tier 1 and Tier 2 evidence, e.g. platform integration testing and the subsequent oversight activities. The 'Wheel' supported the creation of a dialogue based upon a common under-

standing of the relationship between the LRU and the wider evidence. Due to the workshop discussions, supported by the 'Wheel', further supporting evidence was contracted for within the project. The priority and requirements for the evidence, in part, were justified by the 'Wheel' visualisation and subsequent informed dialogue.

Within another software assurance project the 'Wheel' has been of benefit to highlight to stakeholders the assurance evidence required to support a *future* LRU procurement. The 'Wheel' visualisation and the discussion allowed firm evidential statements to be placed within the project contractual requirements. These were to ensure that the supporting evidence was funded and made available for review by the safety assessors (e.g. formal platform integration test results). The 'Wheel' allowed workshop attendees to have informed dialogue to understand *why* and *what* evidence would support a future assurance judgement to align separate software assurance stances.

7.5 Summary of the Potential Permissible Evidence, Underpinning Principles, and Stakeholder Engagement

The reviews of a number of safety-critical domains such as the civil nuclear sector allowed a range of potential evidence items to be identified. These forms of evidence could inform a diverse software argument for airborne software. However, the weighting of such evidence would be subject to SME judgement. The evidence identified has prevalence within other domains and therefore the inclusion is based upon supported conclusions. The identified evidence includes:

- Safety assessment process.
- Life-cycle (software and/or CEH).
- Testing.
- Data integrity.
- Source code architectural considerations.
- Quality assurance.
- Staff competencies.
- Configuration management.

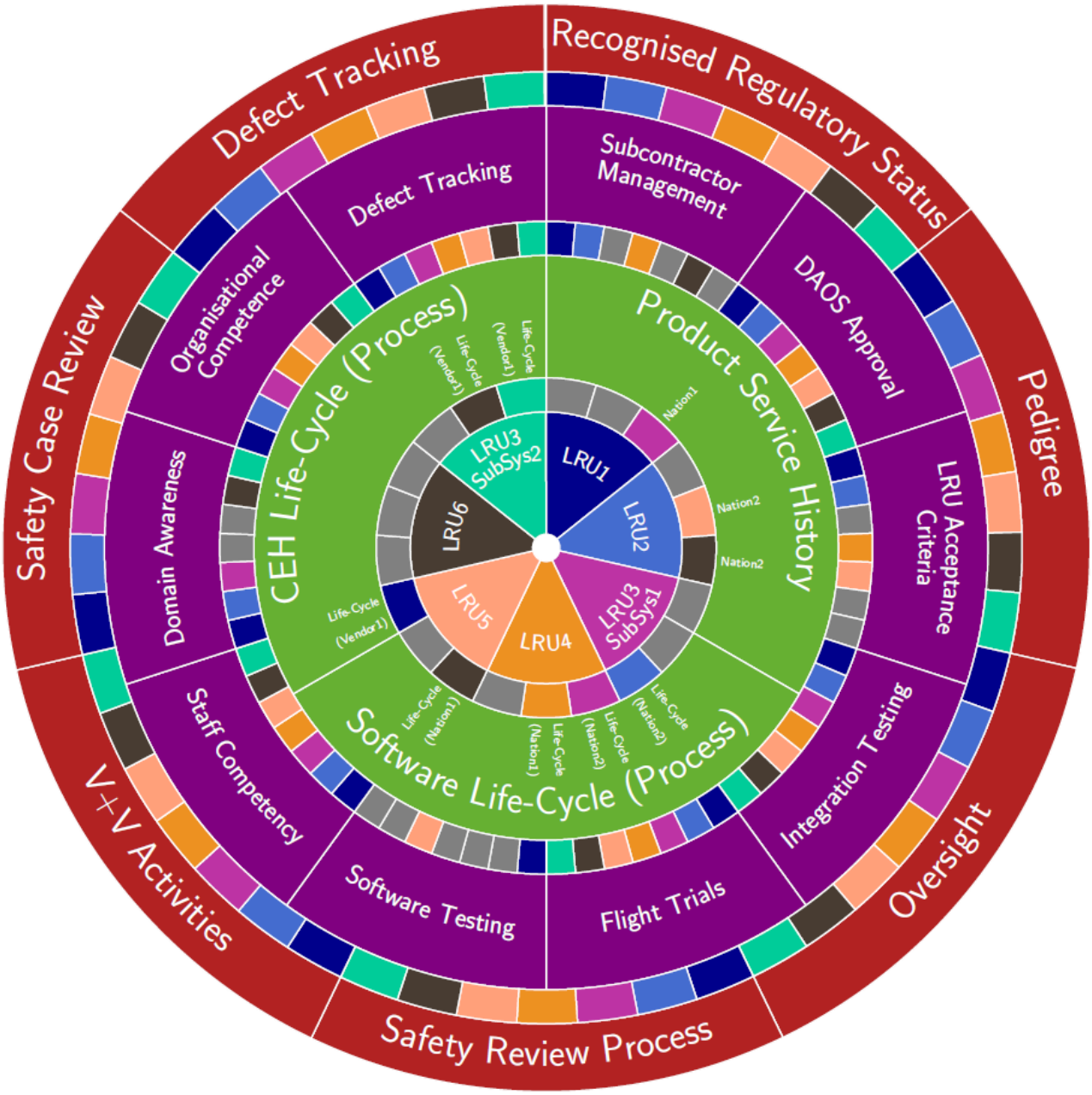


Figure 7.17: Example of a 'Wheel of Qualification'

- Organisation.
- Existing certification/qualification.
- Product Service History (PSH).
- Reliability modelling.
- Security considerations (in relation to airworthiness).

A number of underpinning principles for the use of evidence have also been identified. The principles were stated as the adoption of diverse evidence cannot be undertaken without a clear understanding of the *context* of the evidence. There are a number of factors which may invalidate or limit the value of the evidence being presented if fundamental principles are not adhered to. There are also opportunities to strengthen the use of diverse evidence by undertaking additional activities to gain further confidence. The principles include:

- Establishing an evidence starting point and stopping point.
- Continual monitoring to maintain prior belief.
- Relationship between evidence and type/DAL.
- Understanding the context and environment of use.
- Evidence categorisation and use.
- Evidence roles and effective combination.
- Change from emphasising the process to the product.
- Strategy to reach evidence threshold.
- Utilise opportunities to gather evidence metrics.

One of the principles for gaining further confidence for the software safety assurance is to exploit opportunities to gather additional metrics during the phases of development and in-service[38]. However, although metrics can provide valuable information they can have negative consequences if the metrics are not chosen and used well[39]. The concept of *technical debt* was used to illustrate these issues. Any measure of assurance must take into account the unintended consequences which could arise. Metrics adopted to make judgements must be used with an understanding of their limitations.

[38]For example, the implemented changes from TIM reviews.
[39]For example, incorrect interpretations which then informs a safety assurance argument.

There is a need for data, either large data sets or small, to be made sense of to allow it to be interpreted. The visualisation of data and the ability to *tell stories* with it allows the data to be turned into *information* to drive enhanced decision making. Based upon this principle a concept termed the *'Wheel of Qualification'* was created to allow the relationship between various forms of evidence to be associated with individual LRUs. The visualisation hides the complexity of the assurance activities to assist engagement with multiple key stakeholders. The use of the 'Wheel' has been adopted for a number of projects and presented at multiple workshops and meetings to engage with stakeholders of various roles, e.g. software developers to safety managers.

Chapter 7 has informed two research sub-questions:

- Sub-sections 7.1 and 7.2 have partly responded to the sub-question: *What software safety assurance evidence is relevant/admissible and what are the underpinning principles for the use of such evidence?*.
- Sub-sections 7.3 and 7.4 have partly responded to the sub-question: *What are the unintended consequences of adopting incorrect metrics when forming decisions and how can system/evidence relationships be communicated to stakeholders?*.

Chapter 8

Framework Design and Implementation Decisions

The need for further research in the area of software assurance evidence diversity for military airborne platforms has been established in previous chapters[1]. Also, from previous chapters comes the observation that any framework which is implemented should be fit for purpose. The purpose of the framework in this context is to allow the value of diverse evidence to be demonstrated and for outputs from the framework to assist SQEP decision makers.

To implement a framework underpinning elements need to be established. The characteristics of evidence need to be determined with an understanding of how these characteristics interact to combine. This is a key element to the DSF as understanding the properties of the evidence and the subsequent combination is a task which is acknowledged to be difficult (Weaver et al., 2005). Once the evidence characteristics are established a reasoning approach can be selected to assist with forming judgements based upon the evidence features. There have been a range of methodologies which have been implemented within previous studies in this area, e.g. BBN, DST, ER[2]. The result of this chapter will be a framework which is to be implemented on a range of cases studies, see Chapter 9 (*Case Studies, Exploratory Testing, and Evaluation of the DSF*).

This chapter will examine:

- *Framework Design Tenets.* The key tenets of the design to produce a framework which enables informed decisions to be made.
- *Framework Implementation Decisions.* Implementation details of the framework and the justifications for the design decisions. The section will include:

[1]From Chapter 2 (*Research Strategy*) through to Chapter 6 (*Current Permissible Evidence for Safety-Critical Software Assurance*).

[2]See sub-section 4.2.3.7.

- Evidence to form the initial framework.
- Attributes to capture judgements on the characteristics of the evidence.
- States of the evidence, e.g. obligatory data within a standard.
- Reasoning under uncertainty approaches.
- Structure of the reasoning approach.
- Visualisation(s) to be adopted.
- Wider characteristics of the evidence to assist decision makers, e.g. *change overheads*[3].
- Optimisation approaches to inform decision making for efficient and effective evidence selection.
- Initial data interrogation options to assist decision makers.

- *Summary: Framework Design and Implementation Decisions.* A summary of the implementation details and the justification for the approach.

8.1 Framework Design Tenets

The design of the framework is one which will include a number of key tenets to enable informed decisions to be made by stakeholders. The framework will: capture *what* evidence is of relevance; determine *how* judgements can be made; and allow *decisions* to be made using optimisation and data interrogation methods.

The approach will capture the following information and provide suitable mechanisms for the data analysis:

- *Evidence under review.* Determining what evidence is of relevance to a diverse software assurance argument. An initial set of evidence will be captured in the framework and, importantly, the framework will provide the mechanism for additional evidence to be defined and captured by stakeholders.

- *Evidence attributes.* Suitable metrics will allow judgements to be formed on the evidence. The choice of attributes will influence how the metrics are combined to inform the confidence in the evidence.

[3]When considering a change to an evidential item there are a number of practical considerations such as *time* to change, *cost* of change, and *quality* as a result of the change. There are more than just the theoretical benefits of the evidence to consider.

- *Evidence data states.* Not all evidence is equal in supporting a qualification argument. It is important to differentiate between data which is mandated as part of a standard and that which is supplementary.

- *Methods to reason under uncertainty.* There are a range of approaches to assess evidence and to allow judgements to be formed. The selected approach should serve a purpose and be proportionate to the problem.

- *Structure of the reasoning approach.* Defining the structure and implementing a solution. The method to implement the reasoning approach is to be proportionate to both the problem and the role it has within the research.

- *Visualisation approach.* The framework is to provide informative data to the decision maker. How this data is presented needs to be considered.

- *Evidence characteristics for measurement.* Any evidence may have theoretical value to generate *confidence*. However, there will be practical aspects to obtaining such evidence, e.g. time taken to generate the evidence, and these elements must be considered.

- *Methods for optimisation.* There are a number of data optimisation approaches which help to provide 'solutions' to the decision maker. Methods for the optimised improvements must be proportionate.

- *Data interrogation options.* The decision maker may wish to perform *what-if* analysis to understand various scenarios. The options presented must be as varied as possible but be based upon realistic outcomes to ensure there are viable solutions.

- *Optimised evidence / known limitations.* Optimised evidence attributes could be a framework output. These could inform stakeholders of potential next steps to reach a solution, e.g. to reach a defined DAL target. The output could also include a set of evidence with known limitations, e.g. where a DAL is not achieved. These limitations can be reviewed by stakeholders.

Figure 8.1 shows a potential *flow* of the framework design and the implementation steps.

8.2 Framework Implementation Decisions

The framework design allows a robust and proportionate DSF to be implemented. This section outlines the justifications for the DSF design decisions.

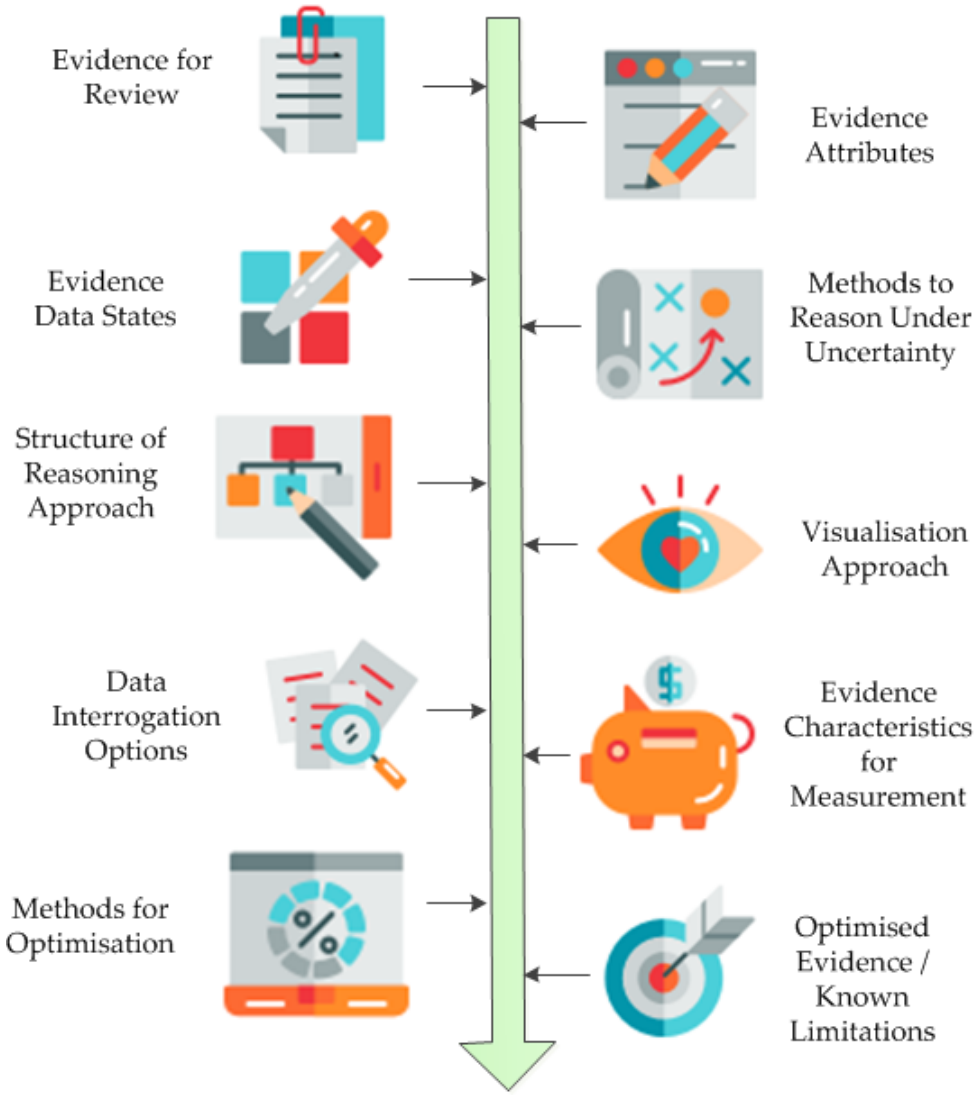


Figure 8.1: Key Tenets of the Framework Design

8.2.1 Evidence for Judgement

The sub-section *Potential Permissible Evidence for MOD Airborne Safety-Critical Software Assurance* (7.1) describes a number of evidence strands which may inform a diverse software/CEH assurance argument. Not all of the potential evidence may be relevant to all projects. However, the evidence in sub-section 7.1 is valid for the initial framework.

The framework will provide the ability to amend, add, and remove any evidential strand/item so that the evidence being assessed represents the project under review. It is important for stakeholders to represent their *worldviews*[4] within the framework.

The types of evidence to be initially included within the framework is in Appendix C. Appendix C contains the evidence types, e.g. staff competencies, and the sources which support the evidence being included within the framework, e.g. Guidance on High-Integrity Software-Based Systems for Railway Applications (RSSB, 2017). The evidence types have a one-to-many mapping to the sources; e.g. staff competencies are valid forms of evidence within a number of guidance documents such as RSSB (2017) and International Nuclear Regulators (2018).

8.2.2 Attributes to Inform the Judgement

A key element to forming judgements is the ability to capture information on an evidential item and to allow properties of that item to be expressed. In essence, this is measuring *characteristics* of the evidence. Understanding the characteristics of the evidence helps with establishing the fundamental features of the evidence itself and how the evidence interrelates with other evidence. The attributes that are chosen to reflect the features of the evidence also determine how the characteristics are combined. The choice of attributes is in some ways a more important driver then the choice of the reasoning approach. Once the characteristics of the evidence are established a reasoning approach can assist with implementing a plausible solution which captures how the evidence features behave.

Measurement is defined as a decision making process with a defined objective (Churchman, 1959). The objective is to gain a level of *confidence* in the software or PEs which will then inform decisions. The decision may be for further information to be gathered or to supplement the existing supporting information. It is important to note that the *confidence* is an output from the DSF and it is intended to be used within the context of a wider set of evidence by stakeholders, see Figure 3.5 within Chapter 3.

The attribute descriptions in the next sub-sections make reference to *parent* and *child* evidence. The child evidence feeds and supports the parent evidence. For a number of

[4]The term *worldview* is not a standard term but in this context it is a system of beliefs, by a stakeholder, that are interconnected (DeWitt, 2018).

the defined attributes the child/parent relationship is a fundamental element to be judged. Figure 8.2 shows the simple relationship with a single parent node and three child nodes, the child nodes also have a relationship as *siblings*.

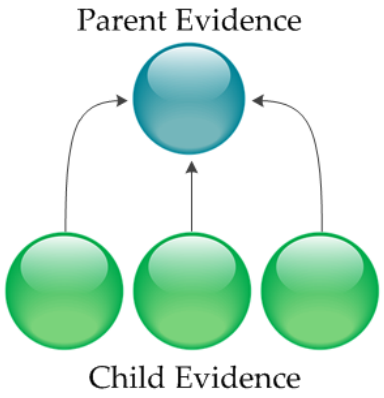


Figure 8.2: Parent/Child Evidence Relationship

The set of attributes captures the characteristics of the evidence and the relationships, e.g. child to parent node and child node siblings etc. The attributes are as descriptive as possible whilst being a manageable number. They also inform how the characteristics of any evidence are propagated to feed into other evidence[5].

A common attribute which spans a number of research papers is one which captures a measurement of the acceptance or level of belief in a claim or statement. This type of metric is frequently referred to as the level of *confidence*. As examples, Grigorova and Maibaum (2013) refers to needing to measure the *truth* of a claim with Denney, Pai and Habli (2011) referring to the level of *uncertainty* in a claim. Cyra and Gorski (2008) refers to *confidence* as being the *acceptance or rejection of a statement*. This indicates that measuring the degree of an overall factor is merited and leads to the framework capturing a *confidence* value.

Whereas *confidence* can be a measure which considers a number of factors there is also a requirement to capture the extent to which any evidence *fulfils its purpose*. This concept is supported by research such as Hobbs and Lloyd (2012), Nair et al. (2015), and Ayoub et al. (2012). They refer to the need to capture how *false* or *true* any evidence is and the levels of *truth* (Hobbs and Lloyd, 2012). There is also reference to measuring the level of *trust* or *belief* that the evidence can be assured to be as *specified* (Nair et al., 2015). Measuring the *likelihood of freedom from errors* is also another definition which is termed *trustworthiness* by (Ayoub et al., 2012). The ability to record the *extent* to which the evidence meets any *specification* or the level of *truth* of the evidence shall be captured by the framework in the form of a *quality* attribute.

[5]Further information on the methods to combine and propagate evidence characteristics is contained in sub-section 8.2.5.

Gathering diverse evidence involves collecting data and information from various sources. However, all evidence is not equal. This concept is captured within a number of research papers which investigate assurance confidence. A number of evidence items which support a claim will have varying levels of *influence* on the ability of the claim to be *true*. This approach is captured as the degree of *weight* (Guiochet, Hoang and Kaâniche, 2015, Cyra and Gorski, 2008), the level of *importance* (Hawkins and Kelly, 2009), and also the level of *power to convince* (Grigorova and Maibaum, 2013), for example. Within the framework this characteristic will be captured as a *contribution* attribute.

The degree to which any evidence meets its specification, for example, will be captured within the framework as the level of the evidence *quality*. However, the child evidence may not be able to allow the parent evidence to fully meet the claim or implied *quality* if the supporting evidence does not allow a complete judgement to be made. In essence, it is important to measure the level to which the parent evidence can be *inferred*, for example as stated by Denney, Pai and Habli (2011). This concept can also be described as the level to which a goal is addressed by the supporting nodes (Ayoub et al., 2013, Yuan et al., 2017). Hawkins and Kelly (2009) have also stated that there is a need to understand the level of confidence that can be gained by determining the truth of a safety claim. The research suggests that there is a requirement to capture the *sufficiency* of the *existing* evidence to meet the intent of the parent evidence. This can indicate the adequacy of evidence when taking into account any evidence not present; e.g. evidence which would ideally be present but is not. The framework will capture this concept as a *sufficiency* attribute.

Related to the characteristic of *sufficiency* is that of capturing how the supporting evidence *interrelates* to inform the confidence in the parent evidence. Supporting evidence can be complimentary or offer an alternative method to derive any confidence, for example. This concept is referred to by Wang, Guiochet and Motet (2017) as the level of *dependency* or *redundancy* of the supporting data. Indeed, diversity is underpinned by the levels of dependence (and independence) of any evidence (Bloomfield and Littlewood, 2006). The characteristic will certainly need to inform how any evidence is *combined* (Littlewood and Wright, 2007). Evidence items can also be described as to the extent to which it *overlaps* (Cyra and Gorski, 2008). These concepts will be captured as an *independence* attribute within the framework.

The attributes are based upon an understanding of *how* the quantification of confidence has been implemented within other research papers. The chosen attributes provide an intuitive method to capture evidence confidence. The attributes have also been subject to SME review and have been judged sound, justified, and workable. The chosen attributes

are agnostic to the forms of evidence[6] (a benefit of the DSF) and allows flexible combinational approaches to how the evidence is reasoned upon (again, a benefit of the DSF). The attributes also allow an adaptable approach to describing the evidence based upon the subjective assessments of SMEs. The chosen attributes are proposed within this research as a *theory* of how evidence can be characterised and are open to be discussed and reviewed by others – indeed, they are open to *falsification* (based upon the concepts and vernacular of Popper (2002)). The reviewed research does not explicitly refute the choice of attributes adopted within this thesis and, at this stage, there is no evidence to suggest the *incorrectness* of the attributes.

The five attributes are listed below, with the following sub-sections describing the attributes in further detail.

- Confidence.
- Quality.
- Contribution.
- Sufficiency.
- Independence.

8.2.2.1 Confidence

The definitions for *confidence* and *confident* are as follows:

- Confidence, *n.* 1. Trust in a person or thing. 2. Belief in one's own abilities; self-assurance. 3. Trust or a trustful relationship. 4. Something confided (Collins Dictionary, 1995*b*).
- Confident, *adj.* 1. Having or showing certainty; sure: *confident of success.* 2. Sure of oneself. 3. Presumptuous. [C16: from L *confidens*, from *confidere* to have complete trust in] (Collins Dictionary, 1995*c*).

To provide context, *confidence* in this instance is "trust in a thing" and "showing [a level of] certainty".

Some studies in the current research for confidence measurement do not provide a clear indication of *how* the outputs should be used by stakeholders to inform decisions. This is the case for previous studies such as Cyra and Gorski (2008), Duan et al. (2015), Guiochet,

[6]For example, the attributes can be applied to *quantitative* evidence (such as the level of test structural coverage) and *qualitative* evidence (such as measuring compliance to a standard).

Hoang and Kaâniche (2015), Yamamoto (2015), and Denney, Pai and Habli (2011). There is a requirement for the DSF output to be informative and usable for it to feed into a wider assurance argument.

For the safety assurance of PEs a method to measure confidence can be to determine a DAL for a system, or the contributing elements such as software or CEH. The term *DAL* is one which is commonly understood within the safety domain and is a feature of the safety assurance 'language'. Indeed, DO-178C, DO-254, and the FAA guidance on the use of PSH (CAST, 1998) all result in the final output being in the form of a defined assurance level. The use of DALs allows a benchmark to be established to act as a *target* when additional activities are conducted to gather further evidence or to refine the existing evidence.

A fundamental concept for the DSF is that there is, in reality, *degrees* of compliance to a DAL. An example is that a software review in accordance with DO-178C may result in some objectives not being met for a DAL A system. The impact of the system having non-compliance to one or more objectives is dependent on the *importance* of the objective and the wider evidence which can *mitigate* any non-compliances. Non-compliance with one or more objectives can still result in a system being declared as achieving a DAL A status. This could be the case if there is *no significant risk to airworthiness*, for example. At present, the non-binary view of DAL compliance is not sufficiently captured within existing methods. An output indicating the *degree* of compliance with a DAL would provide valuable information to a decision maker.

The framework will provide the DAL as an output for the *overall* evidence, i.e. the root node. The evidence sub-strands, i.e. the branches, will be assessed to generate a *confidence* value which will inform the parent. Sub-section 8.2.5 contains further information on these relationships.

8.2.2.2 Quality

The definition of *quality* is as follows:

- Quality, *n.* 1. A distinguishing characteristic or attribute. 2. The basic character or nature of something. 3. A feature or personality. 4. Degree or standard of excellence, esp, a high standard... 7. (*Logic*). The characteristic of a proposition that make it affirmative of negative... (Collins Dictionary, 1995*p*).

The term *quality* is interpreted as being the "basic character" (of the evidence) and the "degree or standard of excellence" which can be attributed to the evidence.

A characteristic such as *quality* allows a judgement on an evidential item to determine to what degree the *requirements are met* or the *rigour* of the evidence against a benchmark.

The judgement on the quality can be conducted via various methods, these are dependent on the type of evidence, for example:

- The *quality* of a SDP would be based upon a subjective opinion comparing the SDP against the objectives within a standard/guideline.
- The *quality* of in-service data could be a measure provided by subjective opinion which is based upon experience and knowledge of the domain. An example includes the level of in-service hours and the amount of representative hours, such as within the military domain rather than civil.
- The *quality* can be a measure of the success of a *quantitative* process, e.g. test results, based upon a subjective judgement.

The framework will allow a SMEs judgement to generate the *quality* attribute which will act as an input to the DSF. The judgement will be informed by the context of the evidence. Figure 8.3 shows the *quality* attribute concept at a simple level. Each *child* node has an associated *quality* attribute.

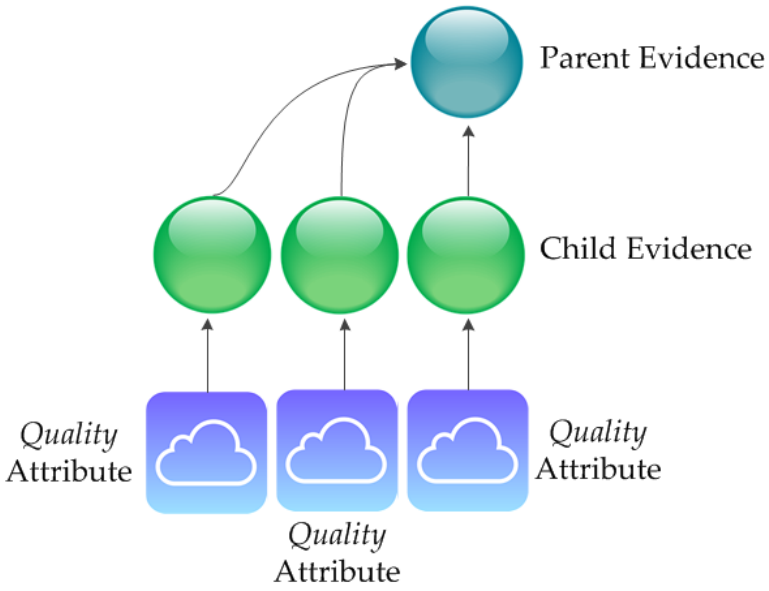


Figure 8.3: *Quality* Attribute Relationship for Child Nodes

8.2.2.3 Contribution

The terms *contribute*, *contribution*, and *relevant* are defined as follows:

- Contribute, *n., pl* -ties. 1. To give (support, money, etc.) for a common purpose or fund. 2. To supply (ideas, opinions, etc.). 3. (*intr.*) To be partly responsible (for). 4. To write (articles, etc.) for a publication (Collins Dictionary, 1995*d*).

- Contribution, *n., pl* -ties. 1. The act of contributing. 2. Something contributed, such as money. 3. An article, etc., contributed to a newspaper or other publication... (Collins Dictionary, 1995*e*).

- Relevant, *adj.* Having direct bearing on the matter in hand; pertinent (Collins Dictionary, 1995*q*).

Contribution is the level of influence, or the level of the direct bearing, that the child evidence has on the parent evidence. As an example, MC/DC, as child evidence, may feed into the broader parent category of coverage testing. Although an evidential item may have a very high level of *quality*, its *relevance* may not be as high. Conversely, if the *quality* of evidence is very low the influence on the parent evidence could be high, and this would act as *counter-evidence*[7].

The *contribution* is a measure which is derived independently of the *quality* as the *contribution* is attempting to understand the weighting that the node has on the parent. As an example, when forming a PSH argument there is a requirement to understand the significance of any software changes that have been made, i.e. those that have altered the core code of the software. The contribution of the 'significance of software changes' may support a parent category to determine the 'impact of any changes'. The *influence* of the child evidence on the parent combines the *quality* and the *contribution* of the child node. Sub-section 8.2.5 contains further information on these concepts.

Contribution is informed by two considerations: the parent which the evidence supports (i.e. the direct bearing on the parent); and the evidence which also informs the same parent (i.e. the sibling evidence). Figure 8.4 shows the *contribution* attribute concept at a simple level. Each *child* node has an associated *contribution* attribute but only one attribute is shown in the Figure for simplification.

8.2.2.4 Sufficiency

The terms *sufficient* and *sufficiency* are defined as follows:

- Sufficient, *adj.* 1. Enough to meet a need or purpose; adequate. 2. *Logic.* (of a condition) assuring the truth of a statement; requiring but not necessarily caused by some other state of affairs (Collins Dictionary, 1995*s*).

[7]Noting that *counter-evidence* refers to the provision of an item of evidence which has the potential to undermine a claim (Menon, Hawkins and McDermid, 2009*a*).

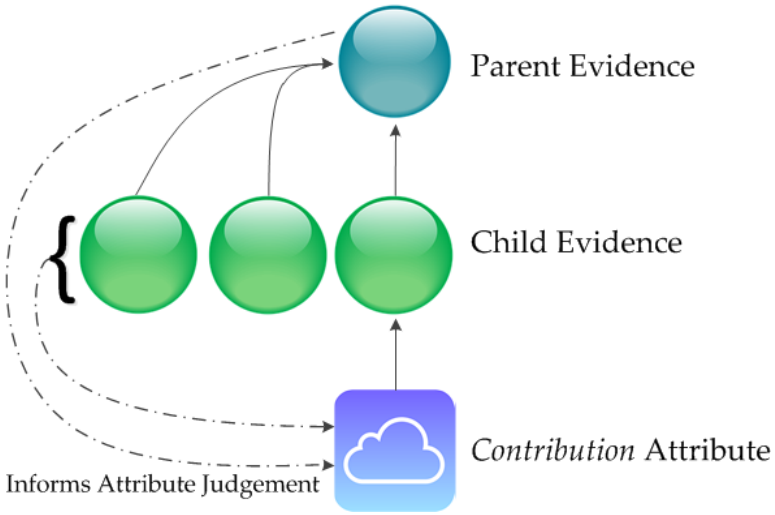


Figure 8.4: *Contribution* Attribute Relationship for a Single Child Node

- Sufficiency, *n.* 1. The quality or condition of being sufficient. 2. An adequate amount (Collins Dictionary, 1995*r*).

The framework should allow a judgement to be made on the holistic properties of the evidence. A key aspect to this is to judge how *complete* and *adequate* the evidence is which underpins a parent node. It is important to judge what evidence is missing, i.e. what *could* contribute, and not only the supporting *available* evidence. As an example, if evidence was available for the design, implementation, and testing stages of the life-cycle; but *not* the requirements stage, then the existing child evidence would not allow a complete perspective to be gained of the broad life-cycle parent evidence. The *sufficiency* concept is illustrated within Figure 8.5.

The *sufficiency* attribute captures the scenarios in which there may be *fundamental* evidence missing to feed into the parent evidence or where wider *supporting* evidence could be gathered. The *sufficiency* of the all child evidence is judged without any prior knowledge of the *quality* of the evidence. The judgement of the *sufficiency* will take into account the context of the software, e.g. platform configuration, to understand the level of *sufficiency* required against the defined scope of the safety assurance claim.

8.2.2.5 Independence

The terms *independent*, *independence*, *distinct*, *diverge*, and *mutual* are defined as follows:

- Independent, *adj.* 1. Free from control in action, judgement etc.; autonomous. 2. Not dependent on anything else for function, validity, etc.; separate. 3. Not reliant on the support, esp. financial support, of others... 7. *Maths.* (of a system of equations)

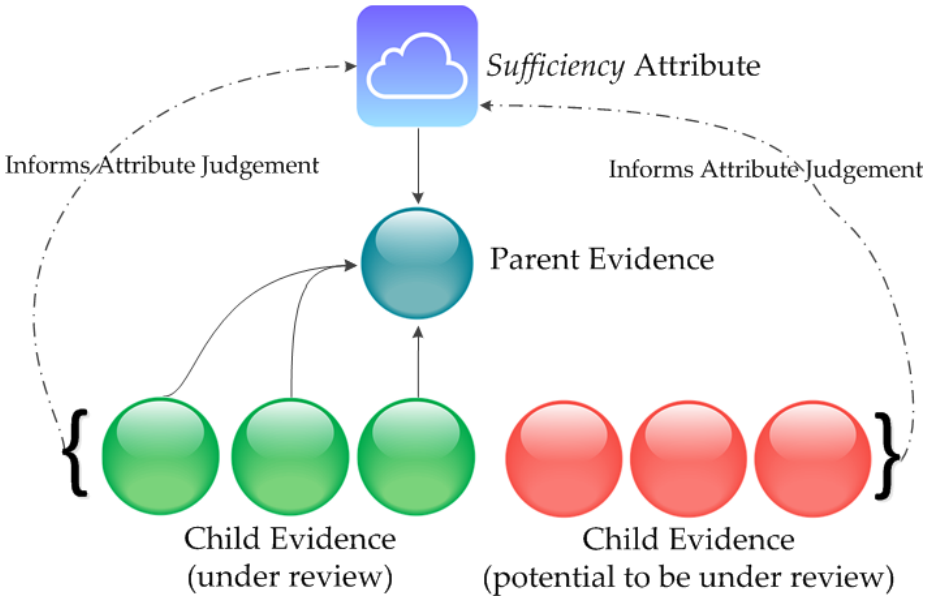


Figure 8.5: *Sufficiency* Attribute Relationship

not linearly dependent. 8. *Logic.* (of two or more propositions) unrelated... (Collins Dictionary, 1995*n*).

- Independence, *n.* The state or quality of being independent (Collins Dictionary, 1995*m*).
- Distinct, *adj.* 1. Easily sensed or understood; clear. 2. (when *postpostive*, foll. by *from*) Not the same (as); separate (from). 3. Not alike; different. 4. Sharp; clear... (Collins Dictionary, 1995*g*).
- Diverge, *vb.* 1. To separate or cause to separate and go in different directions from a point. 2. (*intr.*) To be at variance; differ. 3. (*intr.*) To deviate from a prescribed course... (Collins Dictionary, 1995*h*).
- Mutual, *vb.* 1. Experienced or expressed by each of two or more people about the other; reciprocal: *mutual distrust.* 2. *Inf.* Common to or shared by both: *a mutual friend*... (Collins Dictionary, 1995*o*).

Whilst the *sufficiency* attribute evaluates the evidence which is *not* present, the *independence* attribute is focussed on the relationships between the *available* evidence. The degree of *independence* captures the level to which *all* of the child evidence feeding a parent is providing different perspectives. This considers the sibling relationships of the child nodes.

The more *distinct* or *divergent* the evidence is, it could be argued, the larger the effect of the evidence. Conversely, evidence may also be *mutual* to each other. This indicates

that the evidence may be supported by the sibling evidence and is, in essence, collaborative. However, in this case the evidence is not providing additional concepts and therefore lacks *independence*. The *independence* value is created by a judgement on the collective set of the child node evidence.

Figure 8.6 shows the *independence* attribute concept at a simple level.

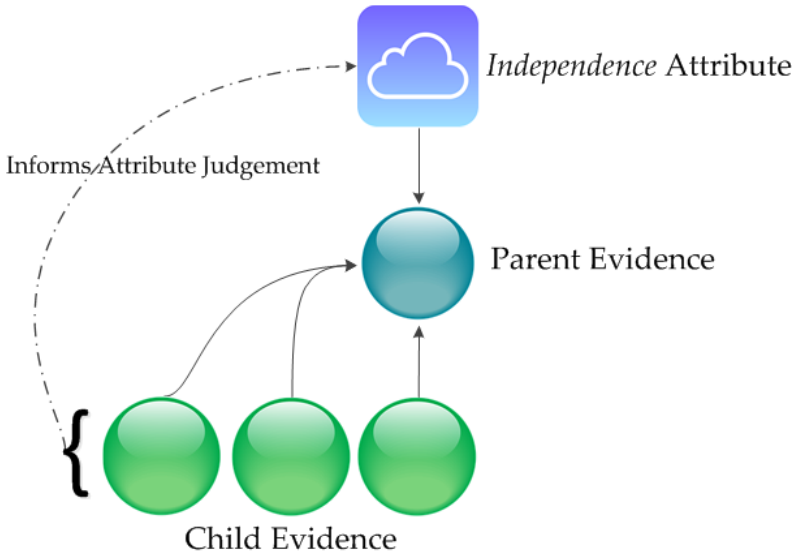


Figure 8.6: *Independence* Attribute Relationship

8.2.2.6 Achieving Diversity via Attribute Selection

The attributes are chosen to capture features of the evidence and its relationships to other forms of evidence. The attributes can also drive how a level of *diversity* can be achieved. The right balance and combination of *all* attributes can provide a satisfactory level of evidence diversity.

Independence is a key consideration for achieving diversity, as supported by Bloomfield and Littlewood (2006) et al. However, evidence which is significantly distinct (and therefore offers a level of *independence*) needs to have a relevant *weighting* for the distinctiveness to be of value. This is why the *contribution* of the evidence is important to capture. Likewise, the *quality* of the evidence also impacts the overall *confidence* and therefore the level of impact that the distinct evidence can have. There should also be a consideration of the collective *sufficiency* of the supporting child evidence. This ensures that the evidence which currently forms part of an assurance argument is not missing supporting items which could assist with improving the level of diversity.

In essence, there is a trade-off and balance between the attributes to ensure that a suitable level of diversity is reached, as illustrated simplistically in Figure 8.7. The aim is to gain

a supporting set of evidence which is comprised of *independent* evidence which provides a satisfactory *contribution* level and is of satisfactory *quality*. Achieving this balance where there are varying values for the attributes is reliant on judgement and subjective decisions.

The framework aims to provide a *decision support tool* which can be adopted by suitable SQEP SMEs to gather and assess evidence. In order to do this there is a need cognisance of the types of evidence being applied, the value, and the pedigree of the evidence. The characteristics of the evidence (at a leaf level) need to be suitably captured and the structure of the evidence needs to be defined. The structure of the tree and the supporting evidence can be SME defined and *does not* have to be limited to replicating the structure of a standard/guideline. Although doing so provides an implicit *warrant* in the Toulmin vernacular (Toulmin, 2003)[8]. Any SME defined evidence structure needs to be based upon evidential theory and how supporting evidence, for example, can underpin the overall confidence. The chosen attributes provide a mechanism for the evidence structure creation, for example the use of *sufficiency* to indicate the adequacy of evidence when taking into account any evidence not present. This allows for circumstances where the user subjectively decides that some evidence which would ideally be present is not; i.e. it captures the *completeness* of the parent based upon the child evidence.

Whilst constructing the parent/child branches and the evidence combination via the selection of attributes the SMEs must be aware of any overlaps in evidence (i.e. barriers to *independence*) and the limitations that this may have. Again, the evidence attributes will allow any observations regarding evidence duplication to be captured throughout the numerous branches of the evidence tree (e.g. by determining the *independence* attribute values). Where there are circumstances that a set of parents have common ancestor node(s) in their subtrees they would not be declared completely independent. However, the SMEs constructing the tree would need to be aware of such dependencies and make their judgement on any *independence* attributes accordingly. The onus is on SMEs to apply judgement and knowledge to structure the tree appropriately based upon evidence theory principles. There needs to be an awareness of how lower evidence has been sourced and judged.

If there are any particular queries or information sought regarding the lower level composition of the evidence structure (e.g. different techniques adopted for assessments or the diverse expertise of the SMEs undertaking review tasks etc) then these can be specifically added as evidence nodes with suitable attributes defined. The DSF allows SMEs to seek clarity regarding the underlying evidence and to *add/amend* nodes and attributes which reflect any areas of concern or clarification. An example is with a number of software V&V activities which should be conducted by *independent* SMEs who are SQEP. Evidence nodes

[8]Further information is contained in sub-section 8.2.5.7.

can be attached to a V&V parent node with child nodes created to state judgements on the independence of the SMEs and their pedigree/training. There should be an assessment of the level of *independence* between the SMEs and the value which can be gained from such evidence (e.g. limitations due to similar training/education of assessors). Attribute values can be defined for such child evidence nodes.

The DSF tool provides a mechanism to *support* SMEs to conduct such activities but the structure and the forms of evidence are SME defined. The DSF is deliberately flexible to allow for varying evidence structures which can be discussed and agreed by SQEP SMEs. The DSF captures *what* evidence is important to the SMEs and allows them to define *how* the evidence is structured. Those that only make use of an overall confidence value to inform decisions, for example high-level risk duty holders, will have a reliance on the underpinning evidence structure/types to have been developed and agreed by SMEs. This is currently the case within safety domains where there are deviations from extant recognised evidence. The DSF has the benefit of allowing these judgements to be *captured* and *interrogated* if required.

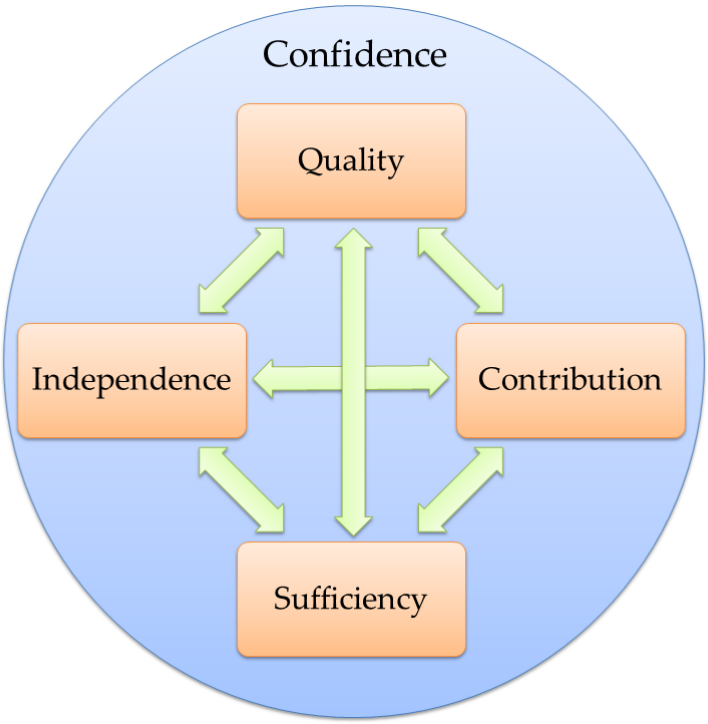


Figure 8.7: Link Between Attributes to Derive Diverse Evidence

8.2.3 Framework Evidence Data States

To act as an effective method to assist with decision making it is important for the framework to take into account the multiple *states* of the evidence data: (a) part of an *existing* software assurance argument; (b) included within an applicable *guideline/standard*; or (c) *supplementary* to the other two data states. All of the data states are legitimate to inform a diversity argument.

The data/evidence states can be used when determining the evidence to be gathered, the value of the evidence, and how persuasive the evidence could be to stakeholders for the assurance argument. It is envisaged that the evidence which forms a diverse argument will initially be one of three states:

- *Extant Data.* Data which is already part of a diverse argument is classed as being *extant*. This is to capture the assessment of pre-existing evidence within a brownfield environment. In these scenarios the evidence will already have been judged and therefore the attributes of the evidence, e.g. *quality*, will be known. Extant data can form a baseline of the known evidence or act as a foundation for any optimisation processes.

- *Obligatory Data.* Any assurance argument would normally be assessed against a known benchmark. In the case of military airborne software this could be DO-178C (RTCA, 2011*a*) for process-based evidence. At the very least there may be a requirement within a compulsory standard for the software development activities to adhere to certain safety properties. Evidence which is cited within guidelines/standards is classed as being *obligatory* and this is to allow this evidence to have an increased value when assessed within the framework. The presence of evidence which adheres to known guidelines/standards will achieve greater acceptance by the *majority* of stakeholders, e.g. regulators[9].

- *Ancillary Data.* A premise is that *all evidence has value.* The degree of value , i.e. the *weight*, will obviously vary but *any* evidence can support an overall diversity claim. There are cases where relevant evidence could inform the *confidence* of a system but the data may not be currently *extant* or part of existing guidelines/standards, i.e. not *obligatory.* An example is with TSOs as these are strong forms of evidence which are not supported within the standards as they have no *official* weight as evidence. It is important for this type of data to be captured as it allows the data, e.g. *quality* or *contribution* values, to be debated by stakeholders with the agreed result contained in the framework.

[9] As indicated within SME workshops.

8.2.4 Method for Reasoning With Uncertainty

The premise of the DSF is to: gather judgements from stakeholders on evidence attributes; allow *what-if* analysis; and to provide an informative output for decision making. The real-world activities to assure PEs can be part of *wicked/messy* problems (Mingers, 2011)[10]. There is a need to operate with a lack, or vagueness, of data and the activities can be conducted under *uncertainty*.

The term *uncertainty* is not one which has a full and accepted definition, however a sufficient definition is provided by Zimmermann (2000):

- *Uncertainty*. Implies that in a certain situation a person does not dispose[11] about information which quantitatively and qualitatively is appropriate to describe, prescribe or predict deterministically and numerically a system, its behaviour or other characteristic.

It is this concept which the DSF will assist stakeholders with to capture and reason with judgements on diverse evidence. For the method to reason under certainty to have value within the DSF the approach is expected to:

- Capture subjective judgements on evidence.
- Represent an evolution in how information is captured to gain stakeholder buy-in.
- Offer a flexible approach to dynamically adjust attributes and the method(s) in which they are captured.
- Allow beliefs to be captured which are valid for *what-if* scenarios but may not be valid for making final judgements, i.e. certain levels of belief are valid for problem *exploration* but not final conclusions.
- Tolerant of imprecise data.
- Allow relationships and dependences of nodes to be captured.

8.2.4.1 Causes of Uncertainty

There are a number of views on the causes of uncertainty. In essence, the differing views are mainly complimentary and refer to such aspects as having to deal with *insufficient data*

[10]For example, those problems which can have, for example, no definitive formulation, no stopping rule, or are considered symptoms of other problems etc (Rittel and Webber, 1973).

[11]In this context Zimmermann (2000) is referring to the term *dispose* to "incline (someone) towards a particular activity or mood" (OED, 2018*b*).

and *vagueness* (Colyvan, 2008). Other sources of uncertainty have been described as being due to *randomness*, *fuzziness*, and *incompleteness* (Blockley and Godfrey, 2000). When combined, Zimmermann (2000) and Colyvan (2008) provide a well-defined list of the sources of uncertainty:

- *Lack of Information.* There are numerous types: no information to base a possible state of nature on; the available information is not sufficient to allow the situation to be described *deterministically*; or a situation of needing *approximation* where sufficient information is not available (or not wanted) to make an exact description.

- *Abundance of Information (Complexity).* Humans have a *limited* ability to perceive and process large amounts of data simultaneously. However, commonly, there is a need to communicate about large numbers of features or properties of a system.

- *Conflicting Evidence.* There may be cases where information about a system may point to a certain type of system behaviour. Conversely, other information may allude to a system behaviour of another type. Further information may increase this conflict. This can occur due to incorrect data or the use of non-relevant features to describe the system.

- *Vagueness.* Uncertainty can arise out of vagueness in the language, in particular, from vague predicates. A vague predicate is one that permits *borderline cases*; e.g. the predicate "is a mature individual" is vague because it permits borderline cases (such as *adolescents*, which are borderline between adults and non-adults).

- *Ambiguity.* Linguistic information, for example, can have different meanings and in a given context it is not clear which way it is being used. This source of uncertainty is quite distinct from *vagueness*. Ambiguity does not give rise to *borderline cases* in the way a *vague* term can, e.g. the term *bank* is either a financial institution or the edge of a river.

- *Under-specificity.* This is where there is unwanted generality with the desired degree of specificity not provided. As an example, the statement that there will be "rainy days ahead" is under-specific as there are numerous questions to this, such as: Which days will be rainy? How many of them will be rainy? The term "rainy days" is also vague: Does a day with light mist count as a rainy day?

- *Measurement.* The term *measurement* also has very different interpretations in different contexts. As an example, *engineering measurement* is concerned with measuring devices for physical features such as weight, temperature, and length etc. An *imagined*

property cannot be measured perfectly, therefore there is only an indicated measure with uncertainty of the real value.

The method to reason under uncertainly within the problem of interest will consider: a lack of information; conflicting evidence; vagueness; ambiguity; and measurement.

8.2.4.2 Types of Information

Judgements will be formed on information from numerous sources. Understanding the forms of information may influence the solutions which are put in place and how the outputs will be interpreted. Zimmermann (2000) provides a well-defined list of information types:

- *Numerical Information.* Numerical information can come from a variety of sources and there is a need to determine the *scale* on which the information is provided.
- *Interval Information.* Information is available in this instance but is not as *precise* in the sense of a real-valued number. This information is *exact* or *dichotomous* in the sense that the boundaries of the intervals, no matter how they have been determined, are *crisp*.
- *Linguistic Information.* Information can be provided in a natural language and not in a formal language. Natural languages develop over time and there is also the need to distinguish between a word as a label and the meaning of a word.
- *Symbolic Information.* Information can be provided in the form of symbols. This can be numbers, letters, or pictures but they will not be as obvious as when words are being used as symbols.

The method to reason under uncertainly within the problem of interest will be subject to considering *numerical* and *linguistic* information.

8.2.4.3 Information Required by the Observer

The outputs of the DSF will have a specific purpose to serve a human observer. For human observers the outputs need to be *readable*. The outputs can be represented to the observer in differing ways depending on the evaluation methods. The information required to be provided to the human observer is described further within *Fuzzy Inference System (FIS) and Structure Implementation* (sub-section 8.2.5), *Visualisation Approach* (sub-section 8.2.6), and *Options to Assist Decision Making* (sub-section 8.2.9).

8.2.4.4 Uncertainty Reasoning Approaches

There are numerous methods to reason under uncertainty and these methods can be underpinned by a range of measures, such as *probability* or *possibility* theories. None of the methods/measures are infallible with each having their advantages and disadvantages (Brito, 2009). Indeed a probabilistic measure is popular when designing under uncertainty but it is not *omnipotent* (Chen, Nikolaidis and Cudney, 1999). Other approaches, e.g. Fuzzy Logic and imprecise probability, are subject to research to ascertain their suitability for related areas, such as reliability analysis (Baraldi et al., 2015).

As outlined in the *Scope for Further Investigation* (section 4.2.3) there are a number of methods, e.g. BBN, which have been adopted for previous studies in the field of software safety assurance. None have resulted in a clear and definitive approach being fully adopted. It is legitimate to adopt a method which is fit for purpose for the problem being addressed. The method will also allow the identified causes of uncertainty to be considered, e.g. vagueness, as well as the types of information, e.g. linguistic. The methods which may be appropriate for the problem of interest; include, but are not limited to:

- *Bayesian Theory.* A common approach to address uncertainty is that of Bayesian Theory which is based upon probability. The probability is interpreted as a *degree of belief* based upon the available evidence (*prior*) with the current knowledge represented by a probability distribution (*posterior*) on a proposition space (Pearl, 1988). New knowledge is learnt via conditionalisation, how beliefs are updated in light of new evidence (Meacham, 2015). BBN uses the fundamentals of probability theory and causal graphical representations via Directed Acyclic Graphs (DAGs). The BBN network representation is visual and easy to understand with probability theory being a well-defined method for dealing with knowledge of unknown certainty (Pearl, 1988). BBN can also allow numerous sources of information to be incorporated within the representation (Liu et al., 2003). However, issues with BBNs include: the reinforcement of the belief in one state would be associated with a decrease of belief in other states as the sum of all possible states must equal 1; assumptions regarding the independence of information/events may lead to counter intuitive and possibly incorrect results; a large number of prior probabilities are required which leads to needing to simplify assumptions; and BBN offers little opportunity to express incomplete information (Liu et al., 2003). Also, probabilities used in BBNs need to be precise. This is not fully compatible with data derived by expert information which tend to be indications (Mertens, 2004).

- *Dempster–Shafer Theory (DST).* DST is concerned with representing and reasoning with uncertain, imprecise, and incomplete information. A key component of DST is

to have the ability to represent *ignorance*. DST avoids the negation of belief from one form of evidence when another increases its belief, elements which are not sufficiently implemented with probability theory. Uncertainty is modelled by the degree of belief with ignorance represented by assigning belief to larger subsets (i.e. given more knowledge, the belief would be assigned to a smaller subset, or even a singleton). DST does have advantages: no *prior* is required for each elements in a set; ignorance can be captured due to lack of information and this value is altered as more information becomes available; and there is no *law of additivity* for the beliefs (Liu et al., 2003). Disadvantages of DST are that: it assumes that evidence is independent, which is not always the case; and it only works on exclusive and exhaustive sets of hypotheses, which is not always the case due to insufficient knowledge/resources (Liu et al., 2003).

- *Hierarchical Process Modelling (HPM).* Within HPM a belief is that a system can be represented by a network of 'blobs' and 'links' with hierarchy offering a way to manage complexity. Each level of the hierarchy expresses more detail, i.e. a decomposition (Yearworth, 2014*a*). HPM consists of conceptual models that contain processes which are structured into a hierarchical arrangement. These represent the minimum processes in a system required to achieve a *purpose*. HPM attempts to allow group decision making on *how* and the *why* information can support a *purpose*. HPM captures levels of belief in a proposition via a probability that it is: (a) false; (b) true; (c) unknown[12] (Yearworth, 2014*b*). HPM data can be captured within a tool called PeriMeta which captures the attributes required for the HPM calculations. Examples of the attributes include, but is not limited to, *necessity*[13] and *dependency*[14]. There are advantages to the HPM approach: there is an adopted tool for assisting with the problem structuring (i.e. PeriMeta), elements of a process which are *unknown* can be explicitly captured, and HPM via PeriMeta can capture the performance of a system quite rapidly. There are disadvantages to the approach: it is recognised that making judgements on the defined attributes is *very hard* with the attributes[15] and their relationships[16] being fixed[17]; there is a limit to the number of processes per level ($\sim 5 \pm 2$); and the aim of HPM is for *intervention* and not *prediction* (Yearworth, 2014*b*).

[12]This concept is visualised via the notation of the *Italian Flag* with the green representing how *true* the measure is; red representing *false*; and white representing *unknown*.

[13]Will the parent fail if the sub-process fails?

[14]How much overlap of evidence is there between the sub-processes?

[15]Such as *necessity*.

[16]Such as the direct mapping between the parent and child to capture *sufficiency*

[17]An intent of the research is determine relevant attributes and their relationships.

- *Fuzzy Logic (FL).* FL systems and FISs[18] allow the mapping of *fuzzy* inputs into a number of *fuzzy* outputs[19]. FL measures the *degree* to which a proposition is correct rather than *how likely* the proposition is to be correct, which is the case with probability theory (Scientific American, 2018). This mapping between inputs and outputs is conducted via sets of fuzzy rules, stated in an *IF...THEN* format, which relate the inputs/outputs. In cases where a FIS receives a crisp input then this would be *fuzzified*[20] by input membership functions (Liu et al., 2003). There are a number of advantages to the FL approach: there are low requirements on the precision of information; it is a good solution for some problems which arise due to language interpretation; an alternative method to map input spaces to outputs spaces; tolerant of imprecise data; and it is capable of dealing with incomplete data. However, there are also a number of disadvantages to the approach: it is not always clear how to construct membership functions and the inherent flexibility in the methods can be seen as an advantage but there is little guidance on the most suitable approach (Liu et al., 2003).

The main aim for the problem of interest is *not* centred upon the choice of method to reason under uncertainty but on the *value of a method* in assisting with understanding and using diverse evidence. This is a subtle yet important point as this work will not advocate a particular method to represent and reason with diverse evidence. This allows the concepts of the DSF to be applied to a range of reasoning approaches if the principles of the DSF are revised (e.g. acceptance of the law of additivity such as with BBN).

Further information on the advantages/disadvantages to each of the reasoning methods can be found within Pearl (1988), Sowa (1999), and Klir (2005). It is not the intention of this thesis to provide significant information on methods to reason under uncertainty. In terms of adopting a method to be implemented within the DSF the use of *FL* is a proportionate and suitable approach. There are a number of characteristics which confirm this (Liu et al., 2003):

- Suited where evidence is itself fuzzy in nature.
- Suitable for uncertain or approximate reasoning, especially for systems where mathematical models are difficult to derive.
- Allows decision making with estimated values under incomplete or uncertain information.

[18] Originated from the work of Zadeh (1965) with further research on fuzzy inference based upon Zadeh (1973).

[19] The *fuzzy* inputs/outputs are based on the premise that they are non-binary and they therefore resemble human reasoning.

[20] The process of converting data to a fuzzy set.

- Inherently accounts for noise in the data[21] because it extracts trends, not precise values.

In addition, FL and the use of FISs has been adopted for a range of risk assessment and decision support applications. Promising results can be found related to the cyber security domain (Sallum, 2015), security risks (Alnafjan et al., 2012) and security audits (Kozhakhmet et al., 2012). FISs have also been adopted for failure analysis purposes (Geramian et al., 2017) with FL deemed a suitable method within some areas of research to conduct analysis within complex decision making (Naseem et al., 2017).

8.2.5 Fuzzy Inference System (FIS) and Structure Implementation

The FIS implementation is one which follows a recognised approach with combinations and comparisons of data conducted in pairs. The analysis via pairs allows for a more simplified and structured assessment. This has benefits in terms of allowing stakeholders to traverse the data tree and the node values using logical sets of inference systems.

The reasoning approach will be formed of three sets of FISs. These will use the attributes which have been determined to provide the most descriptive and informative data[22]. This data will, in essence, be the judgements on the evidence nodes.

The creation of the FISs for the framework will use the *frbs* Comprehensive R Archive Network (CRAN) package (Rize et al., 2015) which is focussed on creating Fuzzy Rule-Based Systems (FRBSs)[23]. The following arguments have been adopted for the creation of the FISs with the choices guided by related FIS implementations and via the use, and subsequent observations, of the framework itself.

- *Defuzzifier*[24]: Weighted Average Method (WAM). WAM was adopted, in part, due to it being less computationally intensive (Ross, 2004) with the design of the framework possibly involving a FIS propagating iteratively over large evidence sets. WAM is also widely used in hierarchical evaluation problems (Guh, Po and Lee, 2008).

- *Inference*[25]. Intersections (t-norm[26]): Standard t-norm - $min(x1, x2)$ (Rize et al.,

[21]The term *noise* refers to the additional meaningless information within a larger data set.
[22]The attributes are: Confidence, Quality, Contribution, Sufficiency, and Independence.
[23]FRBSs are also known as FISs and Fuzzy Models (FMs).
[24]Defuzzification is a transformation that extracts the crisp values from the linguistic terms (MathWorks, 2018*b*).
[25]Inference refers to the process of fuzzy reasoning.
[26]A two-input function that describes a superset of fuzzy intersection (AND) operators, including minimum, algebraic product, and any of several parameterised t-norms (MathWorks, 2018*b*).

2015). Unions (s-norm[27]): Standard s-norm - $max(x1, x2)$ (Rize et al., 2015). The operators were selected as the standard t-norm (*min* operator) produces the largest membership value of all the t-norms and the standard s-norm (*max* operator) produces the smallest membership value of all the t-conorms. It is these features of the standard operators which are significant as they both prevent the compounding of errors in the input/output values (Klir and Yuan, 1995). Most of the alternative norms lack such significance (Ross, 2004).

- *Implication Function*[28]: Zadeh - $(a < 0.5 \;||\; 1 - a > b?1 - a : (a < b?a : b))$[29] (Rize et al., 2015). The Zadeh implication operation method was deemed suitable due to the acceptable close degree values confirmed within Botzoris, Papadopoulos and Papadopoulos (2015) and Zhu et al. (2007)[30]

- *Model*[31]: Mamdani. Advantages of Mamdani systems include being intuitive, having widespread acceptance, and being well-suited to human input (MathWorks, 2018*a*).

8.2.5.1 Child Node Confidence FIS

The *confidence* of the child nodes will be determined via the *quality* of the evidence and the *contribution* which that node makes towards its parent. This combination results in a confidence value which can be used to determine the *evaluation level*. The *evaluation level* is used as an interim process to calculate the combined outputs of multiple child node *confidence* values. Figure 8.8 shows this relationship.

8.2.5.2 Sibling Nodes Assessment FIS

The FIS is generated by combining: (a) the *independence* of the nodes feeding evidence to a parent and (b) the *sufficiency* of the current nodes feeding evidence to a parent in relation to other relevant evidence. The result of this combination is an assessment of the *sibling nodes*. Figure 8.9 shows this relationship.

[27]T-conorm (also known as s-norm) - A two-input function that describes a superset of fuzzy union (OR) operators, including maximum, algebraic sum, and any of several parameterised t-conorms (MathWorks, 2018*b*).

[28]The process of shaping the fuzzy set in the consequent based on the results of the antecedent in a Mamdani FIS (MathWorks, 2018*b*).

[29]When the rule $a \rightarrow b$ is considered.

[30]Both *Euclidean* and *Hamming* close degree values.

[31]The type of fuzzy inference method - broadly classified as *direct* or *indirect*.

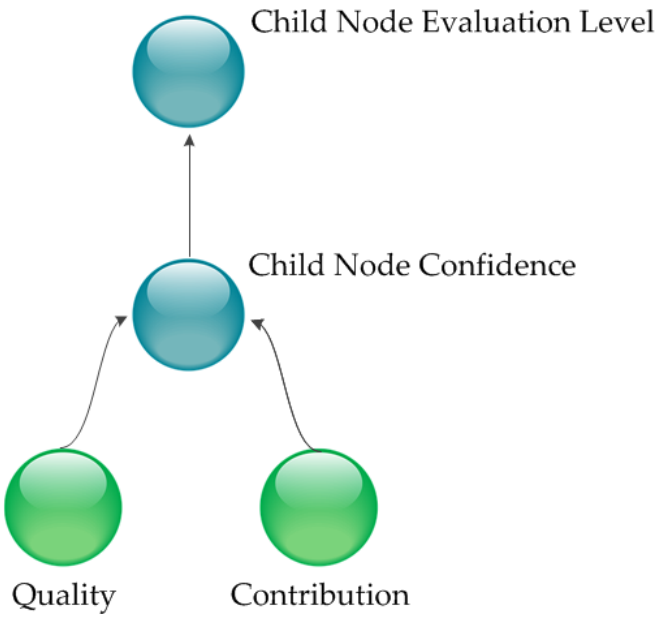


Figure 8.8: *Quality* and *Contribution* Relationship

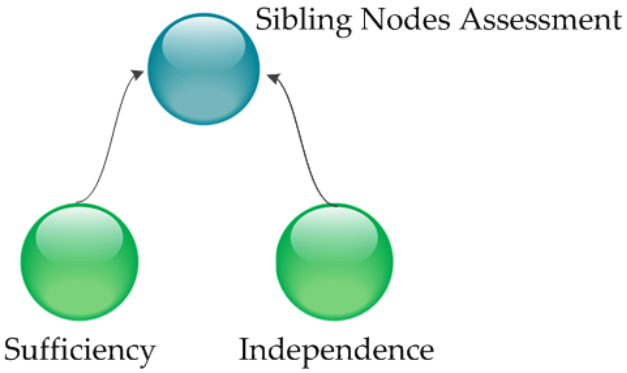


Figure 8.9: *Sufficiency* and *Independence* Relationship

8.2.5.3 Parent Node Quality FIS

The *child node evaluation level* is the combination of the child's *quality* and *contribution*. An assessment of the *sibling nodes* can be combined with the *child node evaluation level* to derive the *quality* value for the parent node, i.e. the parent being fed from other forms of evidence. For the root node the *quality* value is used to determine the overall DAL for the evidence. Figure 8.10 shows this relationship.

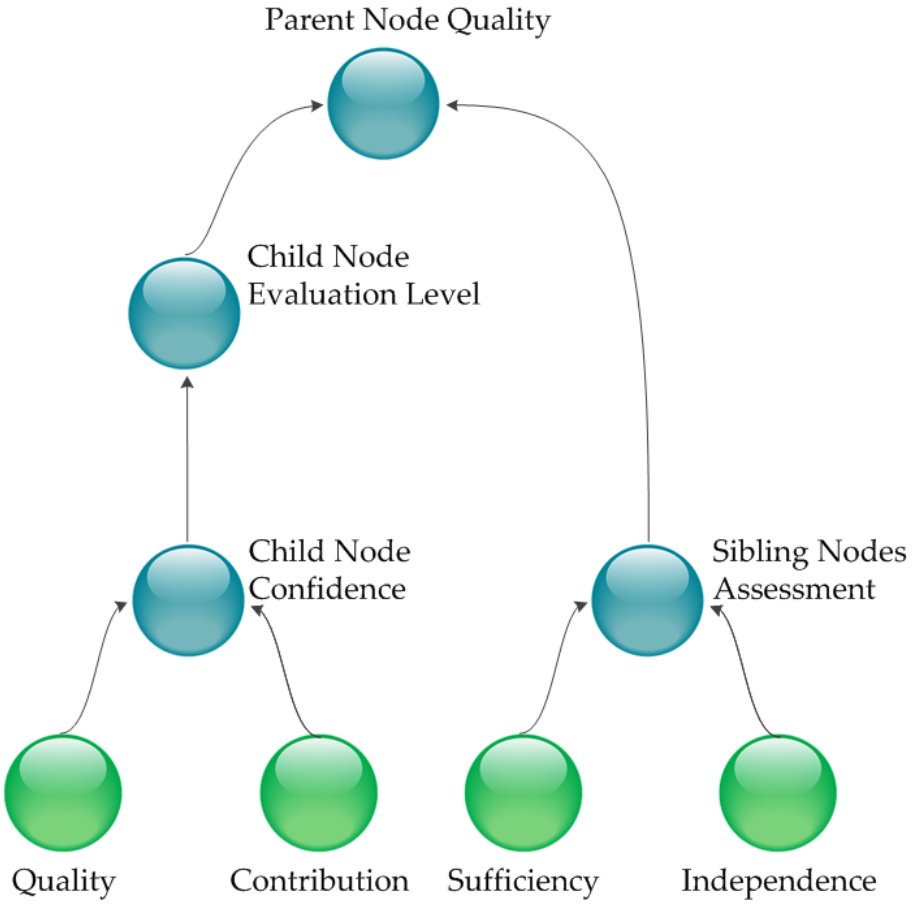


Figure 8.10: *Child Node Evaluation Level* and *Sibling Nodes Assessment* Relationship

8.2.5.4 Parent Confidence

The confidence of the parent evidence node is, in essence, a reuse of the FIS to calculate the *child node confidence*. The *parent quality* is combined with the parent's *contribution* that it makes to its *own* parent. Figure 8.11 shows this relationship.

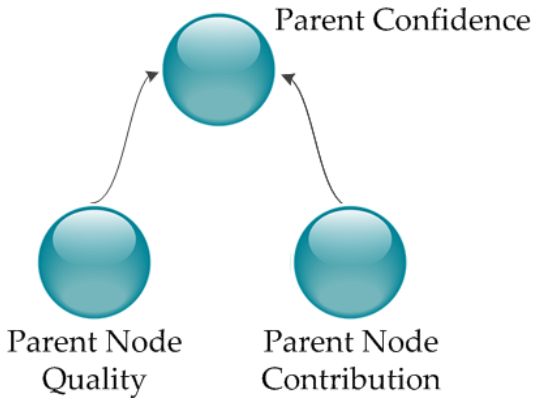


Figure 8.11: *Parent Node Quality* and *Contribution* Relationship

8.2.5.5 Overall FIS Structure

The sets of FISs are combined for each evidence *family*. This acts as a *pattern* which can then be repeated on all of the evidence to provide a root value which represents the DAL for the overall evidence.

Using the *pattern* across a range of evidence allows the nodes to be judged in a consistent manner using the same language. This allows stakeholders to have a common assessment method. Figure 8.12 shows the overall relationships for an evidence *family*.

8.2.5.6 Fuzzy Inference System's (FISs) Membership Functions (MFs)

MFs within FISs represent the degrees of truth for a given value. This degree of the membership is a value between 0 and 1 and is associated with each point in the input space, termed the *universe of discourse*. The purpose of the MF is to articulate that a system, or a characteristic, may belong to a MF *to a degree*.

There are a number of types of MFs. A degree of membership must be between 0 and 1 but the actual type of the membership function can be based on the need for simplicity, convenience, speed, and efficiency (MathWorks, 2018*b*). MF types are based upon a number of basic functions: piece-wise linear functions, Gaussian distribution function, sigmoid curve, and quadratic/cubic polynomial curves. However, the simplest MFs are formed using straight lines, e.g. triangular and trapezoidal, and it is these MF types which are to be adopted within the FISs. It is the *intersections* of the MFs which feed the FIS values.

An example of MFs to capture the *quality* of any given evidence is represented by Figure 8.13. The MFs to represent *quality* use the following linguistic terms (and numeric values): *very low* (0-30), *low* (30-50), *medium* (30-70), *high* (50-90), and *very high* (70-100). The degree of membership is represented on the y-axis.

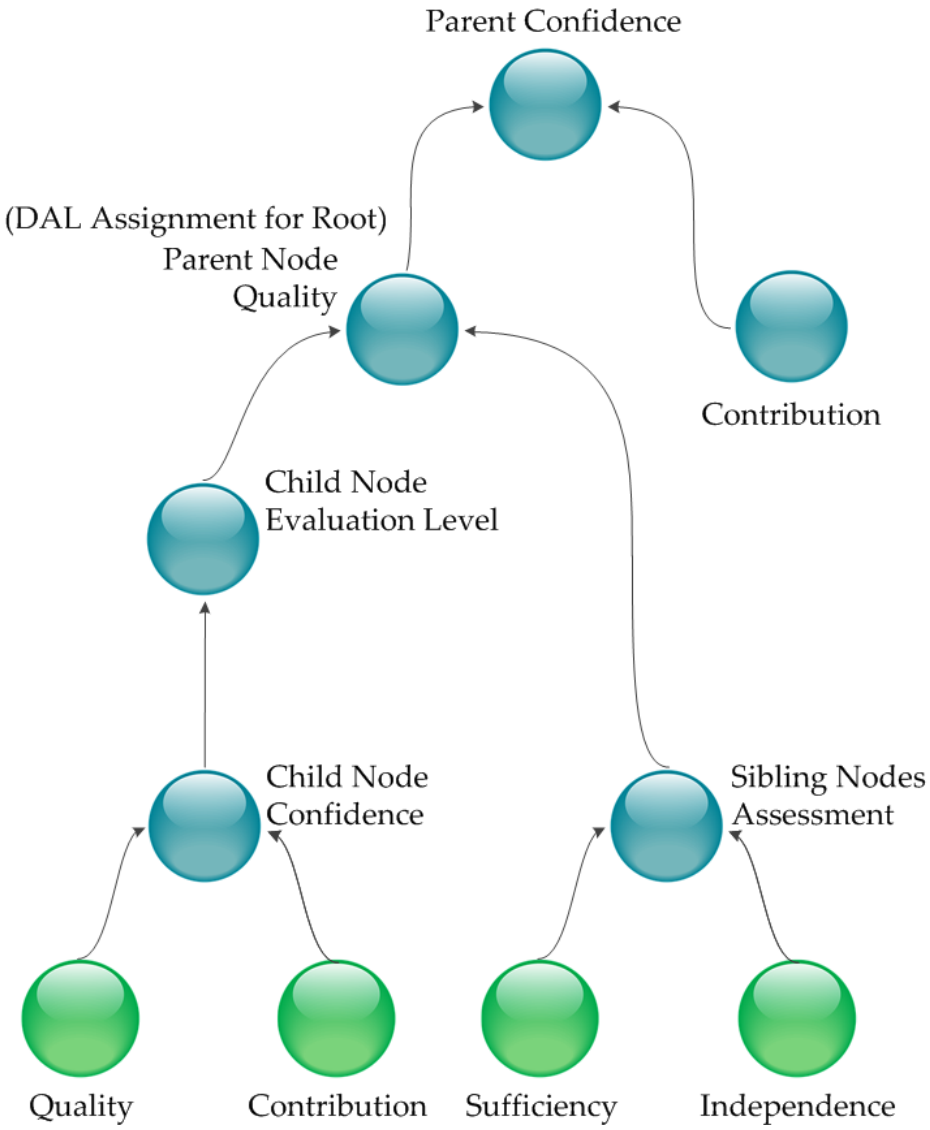


Figure 8.12: Overall FIS Relationships for an Evidence *Family*

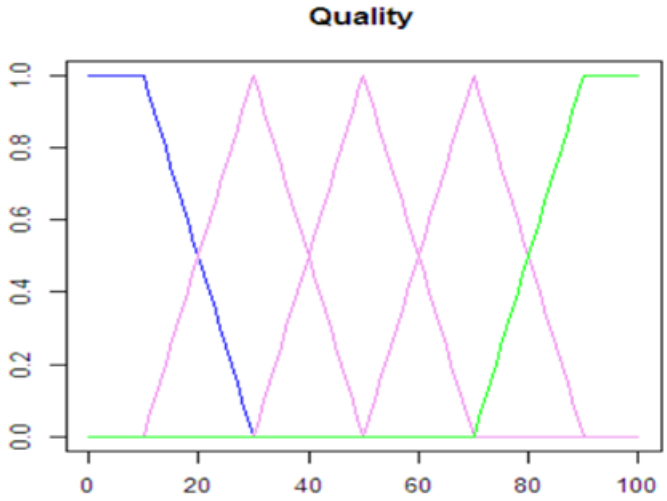


Figure 8.13: MFs for Evidence *Quality*

The degree to which the overall evidence relates to the DALs is determined via the degree of MF to a particular DAL value. The MFs for DALs A-E (and *none*) are shown in Figure 8.14. Again, the degree of membership is represented on the y-axis.

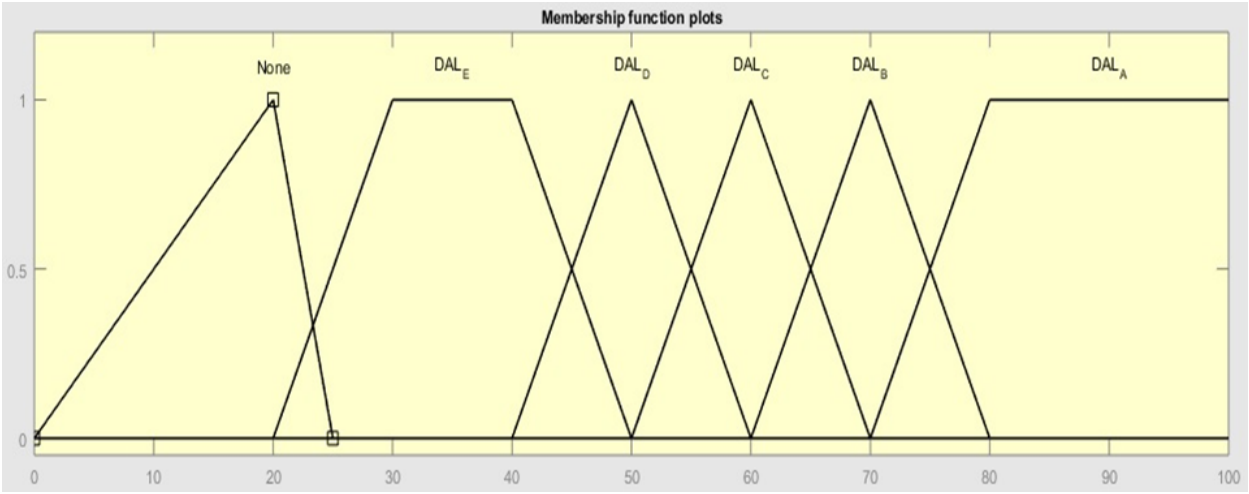


Figure 8.14: MFs for Overall Evidence DALs

The use of the MFs for the DALs allows complete compliance to be achieved for a particular DAL. These scenarios would be based upon the overall confidence values for the evidence tree. Table 8.1 shows the degree of membership for each DAL and the calculated overall confidence value of the evidence via the FIS.

Table 8.1 shows specific *minimum* overall confidence values required to achieve a particular DAL. The examples within Table 8.2 show the degrees of membership for the DALs based upon other overall confidence values. These examples are more typical of the results

	Degree of Membership				
Overall Confidence (OC)	**DAL E**	**DAL D**	**DAL C**	**DAL B**	**DAL A**
80	0	0	0	0	1
70	0	0	0	1	0
60	0	0	1	0	0
50	0	1	0	0	0
30	1	0	0	0	0

Table 8.1: MFs for Precise DALs

which are expected to be obtained from the case studies as they indicate that the overall evidence will have *degrees* of *compliance* to one or more DALs. The degree of membership will then allow stakeholders (the decision makers) to determine the suitable next steps to *increase* the degree of membership for a particular DAL if appropriate.

	Degree of Membership				
Overall Confidence (OC)	**DAL E**	**DAL D**	**DAL C**	**DAL B**	**DAL A**
76	0	0	0	0.4	0.6
63	0	0	0.7	0.3	0
58	0	0.2	0.8	0	0

Table 8.2: MFs for DALs

8.2.5.7 Structure Definition and Roles/Forms of Evidence

There are a number of patterns which can be applied to the creation of evidence structures within the main threads. These approaches can reflect the key evidential areas of interest for the SMEs. A common pattern for the generation of the parent/child linkages is to base it on a top-down composition which defines the child nodes as acting as the total (or near total) *sufficiency* of the parent evidence. In this pattern the evidence structure is providing a *framework to be populated* and is separate to the *actual* evidence judgements which will be captured later. As an example, if the software development life-cycle is deemed to be a suitable parent evidence node then the child nodes will be selected by their ability to determine the *confidence* of the life-cycle. *Confidence* of any parent is calculated, in part, by the *sufficiency* of its child nodes. For a software development life-cycle parent node it can be *argued* that an assessment of the child nodes such as customer query analysis, requirements, design, implementation, V&V, and through-life maintenance development would all collectively be *sufficient* to inform the *confidence* of the parent node[32].

[32]The premise of the attributes' relationships is outlined within sub-section 8.2.5.5.

Another possible pattern is one which captures the activities needed to *create* the evidence and to also measure the confidence in the evidence output itself. This is related to the *evidence common characteristics* concept by Ayoub et al. (2012). Child nodes associated with any parent may reflect one of two concepts: (a) the activities required to *create* the parent evidence and the (b) evidence needed to provide *direct confidence* in the parent. This is a subtle distinction. For example, software requirements development should be conducted by suitably skilled staff adopting suitable requirement standards. These activities are needed to *create* the requirements. Evidence needed to provide *direct confidence* in the requirements could be based upon suitable requirement specifications. The evidence *creation* activities can be associated directly with the evidence it relates to or it can be included in a separate assessment of the wider tools and skill-sets, for example. These may cover *all* software development life-cycle steps. The DSF is flexible to allow a range of patterns to be adopted based upon such SME judgements.

These patterns are theories for the creation of evidence structures and are open to debate and discussion. However, patterns such as these proved useful for the DSF case studies and exploratory testing.

The creation of any new branches within the structure is dependent on the type of argument which is being proposed and if there are existing standards/guidelines which can support the structure of the evidence. As an example, a suitable approach is to reflect the objectives contained within a recognised or extant standard, such as the objectives within Annex A of DO-178C. Such an approach provides a *warrant* using the Toulmin vernacular (Toulmin, 2003). A standard has acceptance by the relevant regulator that following the requirements/objectives will achieve the stated aims of the standard. Using the requirements/objectives of the standards in effect forms the argument of the safety claim with the artefacts forming the evidence. Wider research suggests that standards such as DO-178C can explicitly meet the narrowly defined safety goals (Holloway and Graydon, 2018)[33] and therefore safety arguments being formed from the requirements/objectives of standards is a suitable approach. There are also suggestions that safety standards should be stated in the form of assurance cases to make the link to safety arguments more explicit (Rushby, 2015).

Where a standard/guideline does not exist (or is not deemed to be the most effective structure by the SMEs) then an approach could be applied which reflects the *philosophy* of the SMEs generating the structure and the important intrinsic properties which are to be captured. For example, the parent to child node decomposition could be based on deriving what would provide total (or near total) *sufficiency* of the parent node or potentially the activities required to *create* the parent node evidence (e.g. suitable tools and skill-sets). Ex-

[33]Related to software *development*.

amples of patterns have been described in this sub-section; however, the process of evidence construction contains a level of intrinsic intellectual activity, it is not a mechanistic process for which a complete set of rules can be provided.

Evidence structure can also be constructed from a *bottom-up* or *top-down* perspective. The chosen approach is dependent on the *available* evidence and the premise being generated. A *bottom-up* approach is one which may be required if there is a need to determine *what* the evidence can inform with no fully developed pre-defined evidence structure. The parent/child branches are created based upon the *available* evidence with the attributes used to determine the *sufficiency* of the evidence, for example. This approach builds the evidence structure, in theory, from the *bottom-up* although there is always a need for the SME to be aware of what evidence would *fully* support the available evidence to provide a full picture of the evidence confidence. This approach was not generally adopted for the case studies as the preferred approach is to use a pre-defined structure with stated *sufficiency* and *contribution* values for example. This pre-defined structure reduces the risk and perception of *gaming* (e.g. that SMEs would place more positive judgements only on what is *available*). However, the DSF can accommodate such an approach if deemed suitable by SMEs. The *top-down* approach is the more standard method which is based upon determining patterns from parent to child.

The DSF is purposefully flexible to take into account a range of evidence gathering scenarios. This includes, the creation of a safety assurance framework from a greenfield perspective, i.e. that which does not have an existing safety argument in place with evidence being predominantly based upon process evidence. Brownfield[34] developments are also accounted for with the ability to include evidence which supports the in-service confidence - the DSF is evidence agnostic. The evidence attributes can capture any such scenario.

As outlined within sub-section 3.1.3 the outputs generated from any framework implemented from this research[35] can inform an overall *system* safety claim by acting as a sub-claim. The method to generate the sub-claim is deliberately able to be defined by the stakeholders and decision-makers. The structure of the evidence can be formed to present: (a) sufficient confidence gained via satisfying (in part or in full) a defined standard/regulation[36]; (b) sufficient confidence to be gained via a claim, argument, evidence approach[37] which does not have a reliance on a standard/regulation; or (c) a combination of standard/regulation satisfaction and a claim, argument, evidence approach.

[34]'Brownfield' projects, in this instance, are defined as those which were previously created (Baley and Belcham, 2010).

[35]Specifically the outputs such as evidence *confidence* and the degrees of memberships to the DALs; see Tables 8.1 and 8.2.

[36]As outlined within this sub-section.

[37]As outline within sub-section 3.1.3.

8.2.6 Visualisation Approach

A visualisation for decision making and evidence comprehension has been proposed earlier in this thesis to allow tiers of diverse evidence to be displayed and managed via a single 'dashboard'. Further information on the *Wheel of Qualification* can be found within subsection 7.4.

To create the *Wheel of Qualification* visualisation the principles and guidance from Nussbaumer-Knaflic (2015) and Kirk (2016) were adopted. These principles and guidelines informed the method to display the evidence attributes values. They also informed how the evidence structure is to be presented to assist with decision making. The main purpose of the visualisation is that it assists decision makers to make informed judgements based upon potentially complicated evidence sets.

There is a need to ensure the accuracy of the data visualisations by cross-referencing the results. Data visualisation errors, e.g. via unintentional mistakes, can lead to misinformation or incorrect decision making (Evergreen, 2017).

8.2.6.1 Visualisation Structure

To assist with the decision making process the representation of the data needs to be clear and unambiguous. There are a range of chart types, e.g. plots and tables etc, with each having a number of chart methods, e.g. Bubble Chart and Kagi Chart etc. The visualisation is to assist with decision making and not to perform formal analysis on the supporting data. There are a number of visualisation packages on CRAN[38] to support R[39].

For the visualisation of the relationships between evidence nodes there are a number of options available (Data Visualisation Catalogue, 2017):

- *Parallel Coordinates Plot.* Used for comparing a number of variables together and viewing the relationships between them. Each variable is given its own axis and all axes are placed in parallel to each other. However, the order in which the axes are arranged can impact the way in which the information is understood. Also, the plots can become over-cluttered, although this can be mitigated via the use of interactive plots.

- *Network Diagram.* Used to show how entities are interconnected. The nodes and links can be used to visualise additional information such as a node size being in proportion to an assigned value. Diagrams can be used to interpret structure. However, they have a limited data capacity and can be difficult to read with a large number of nodes.

[38]See the following for further information: https://cran.r-project.org/.

[39]R is a free software environment for statistical computing and graphics. See the following for further information: https://www.r-project.org/.

- *Arc Diagram.* Used as an alternative to Network Diagrams (via a 2D visualisation). Arc diagrams can be used to find co-occurrence within data. However, Arc Diagrams do not show structure between nodes and too many links can make the diagram difficult to understand.

- *Chord Diagram.* Used to visualise inter-relationships between entities. Ideal for comparing similarities within a dataset. Colour can be used to group data into different categories. This can assist in making comparisons and distinguishing groups. However, over-cluttering can become an issue.

- *Non-Ribbon Chord Diagram.* Used to emphasise the connections within the data. In essence, it is a stripped-down version of the Chord Diagram as only nodes and connections are shown.

- *Linkage Diagram.* Used as a visual representation of hierarchies via nodes and links/ connections. Branches represent relationships and connections between members. Can be used to show large data sets.

- *Venn Diagram.* Used to display all logical relationships between a collection of sets. Each set is a collection of entities that have commonality. Overlapping sets result in an intersection area. Large volumes of data would result in a very complicated diagram.

- *Treemap.* Used as an alternative to show a Linkage Diagram and can show quantities for each category via area size. The area is in proportion to the quantity. Treemaps are a compact method of displaying Linkage Diagrams and provide a method to provide an overview of data.

- *Circle Packing.* Used as an alternative to a Treemap. Circles used instead of rectangles. Containment within each circle represents a level in the hierarchy. Circle-Packing Diagrams are not as space-efficient as Treemaps.

- *Sunburst Diagram.* Used to show linkages through a series of rings. Each ring corresponds to a level in a hierarchy. The segment sizes can be proportional to a value.

Based upon a review of the above visualisation methods the approach which is the most appropriate for the DSF is that of the Linkage Diagram. The diagram type is suitable to allow stakeholders to capture and structure evidence in a logical and straightforward manner. There is a trade-off between the richness of any linkage/visualisation approach and that which allows data to be managed and considered by stakeholders via uncomplicated methods. Linkage diagrams will allow methodical analysis of the data by stakeholders.

The Linkage Diagrams have similarity to GSN structures, which will be familiar to the stakeholders who will interact with the DSF. Also, the Linkage Diagrams allow a substantial level of information to be viewed allowing a holistic perspective to be gained when the supporting assurance evidence is being assessed. Initial feedback is also supportive of the use of Linkage Diagrams[40]. Linkage Diagrams are also known as Linkage Trees; Figure 8.15 shows the 'anatomy' of a Linkage Tree.

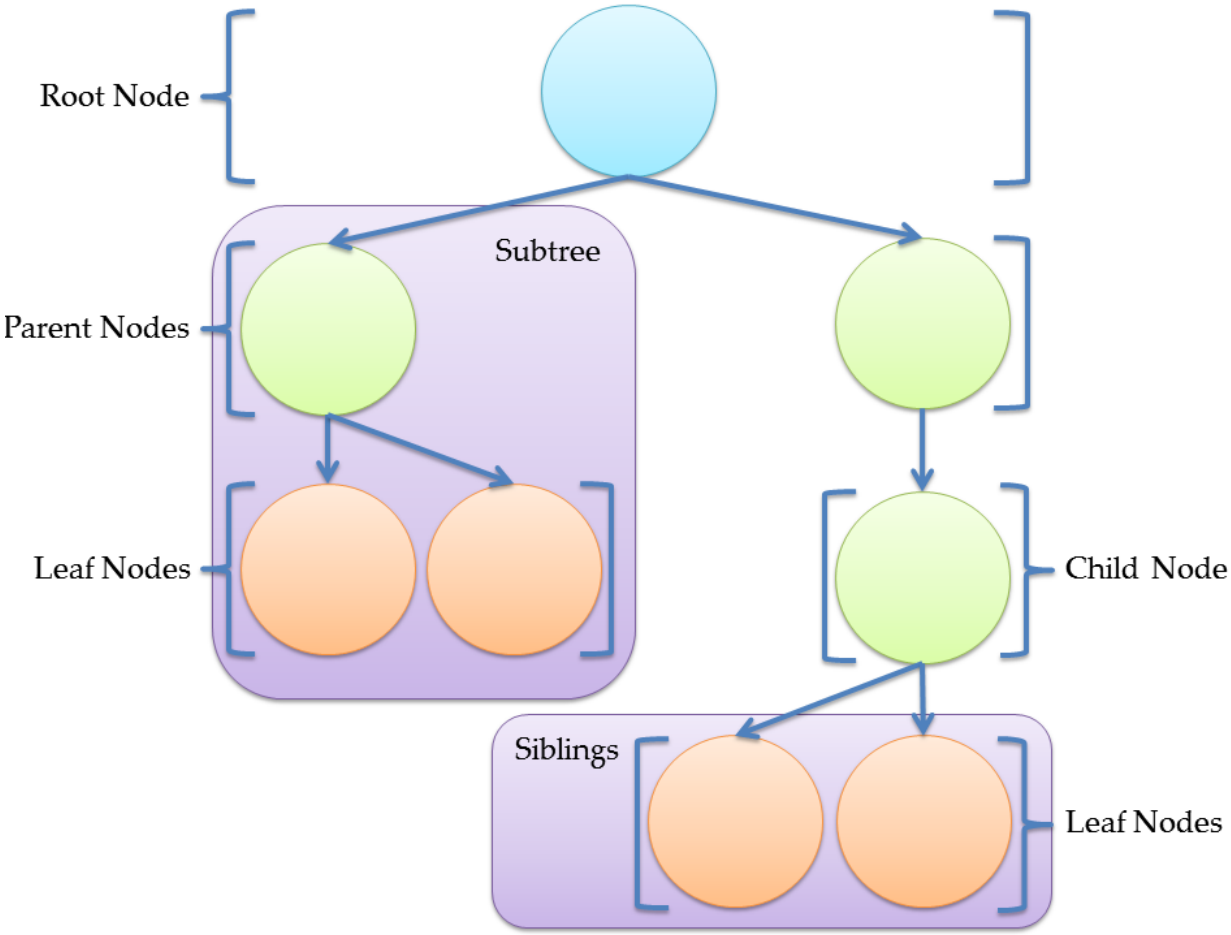


Figure 8.15: 'Anatomy' of a Linkage Tree

8.2.6.2 Visual Indicators

Consideration is needed into how variations in data are to be represented. Each node may have multiple attributes and values so there is a need to allow the decision maker to select the attribute of interest for a particular scenario being reviewed. The structure should be able to display the relevant values of the nodes, e.g. *confidence* level. Also, when *what-if* analysis is conducted there will be a need to understand the *differences* between scenarios

[40]From SME group discussions.

and the values of the same node attributes. The colour gradients for the evidence attribute values would be in relation to the MFs values. The colour gradients for the *what-if* analysis would indicate the *degree* of change of a node attribute value *between* scenarios.

Figure 8.16 shows the potential colour gradients to present node attribute value/differences, *red* indicates a low value through to *blue* indicating a high value. As with the FISs MFs this allows *degrees* of compliance to an attribute, such as *quality*, to be represented.

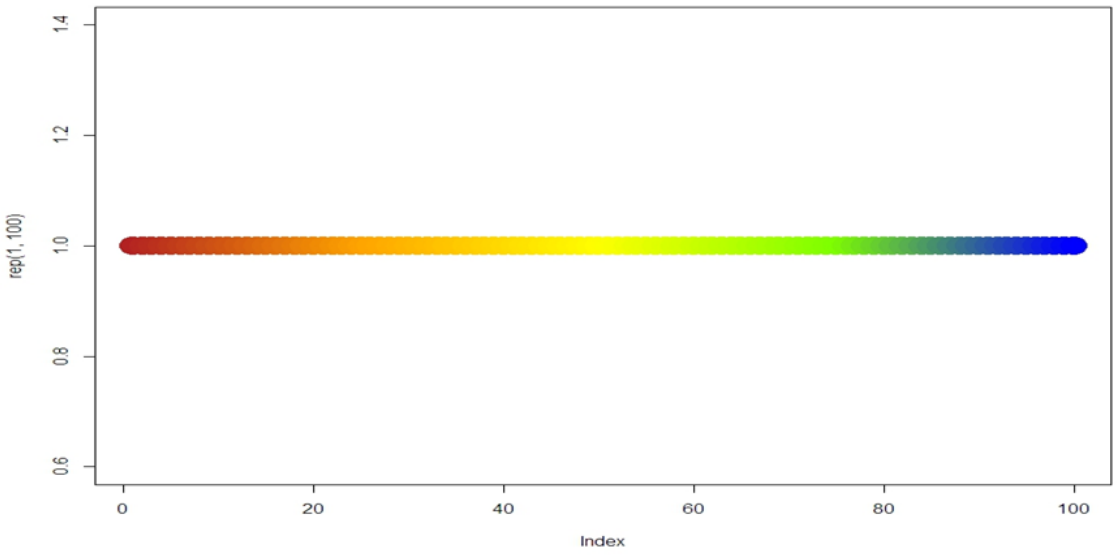


Figure 8.16: Colour Grades for Nodes

The colour grades for the nodes will allow a visual representation of the overall tree to show varying levels of attribute values and/or potential optimisation differences, e.g. *delta* views between scenarios. This method allows particular evidence strands/nodes to be reviewed quickly and efficiently. Figure 8.17 shows: (a) an example of a tree with appropriately coloured (graded) nodes and (b) a representative example of how the colour grades can be associated with the degrees of membership for one or more DALs[41].

The advantage of the data structure visualisation is that it shows the level of confidence in the evidence at the leaf nodes and the subsequent parents. Evidence and parent nodes with *weak* levels of confidence (and will be of more of concern) are closer to the red colour gradient. Evidence and parent nodes with *strong* levels of confidence are closer to the blue colour gradient. The data tree shows how evidence attributes propagate through the tree to ascertain particular causes of concern or areas of strength. The propagation also indicates

[41]It should be noted that it is not the intention for the node text to be readable within Figure 8.17 as it is the node colour gradings being illustrated. However, decision makers using the framework will have the ability to search and review the individual nodes of the tree.

the contributions that the nodes make to their parent. The FISs are repeatedly implemented throughout the tree at each parent/child(ren) structure to derive the values.

8.2.6.3 Traversal Order

The traversal path of the structure is also important as this will drive the order of the FISs and other calculations. The leaf nodes will be determined first with the child node calculations feeding the parent nodes. The root node will be determined by its child nodes. This is known as a *post-order* traversal (left, right, root). For Figure 8.18 the traversal order of the nodes would be as follows: $A \rightarrow B \rightarrow F \rightarrow H \rightarrow C \rightarrow D \rightarrow E \rightarrow G \rightarrow I \rightarrow J$. The important aspect is that the evidence tree calculations need to be from the *bottom-up* with nodes at the *same* level being able to be calculated in any order. The parent-to-children relationships are fundamental with the DSF being agnostic to the ordering of a node's siblings.

8.2.7 Capturing Related Evidence Characteristics

There is a need to look wider than the attributes which are initially placed on the evidence, e.g. *quality* and *contribution*, to ascertain if changes can increase confidence in the diverse argument. Understanding the *immediate* attributes to inform confidence are, in reality, only one consideration for any practical implementation of the framework. This acceptance and ability to consider the wider evidence factors is an additional element missing from existing methods in this domain.

When considering a change to an evidential item (be it extant, obligatory, or ancillary) there are a number of practical considerations which will provide a view on the evidence *realities* rather than the *theoretical* benefits of the evidence. Considerations mainly include those which feature as key project management constraints (Atkinson, 1999) for when there is a requirement to potentially *purchase knowledge* (McDermid, 1998). The project which is the subject to the software assurance activities will have its own defined timelines and budget constraints and therefore the ability to enhance any evidence should take these factors into account.

- *Time*. Changes to extant data within an assurance argument have time implications, e.g. the development of more formal design documentation to be more in-line with defined DAL requirements. There are also timing considerations for the addition of evidence, e.g. conducting further testing. These implications should also include the time to create relevant business cases and to contract with vendors for the changes

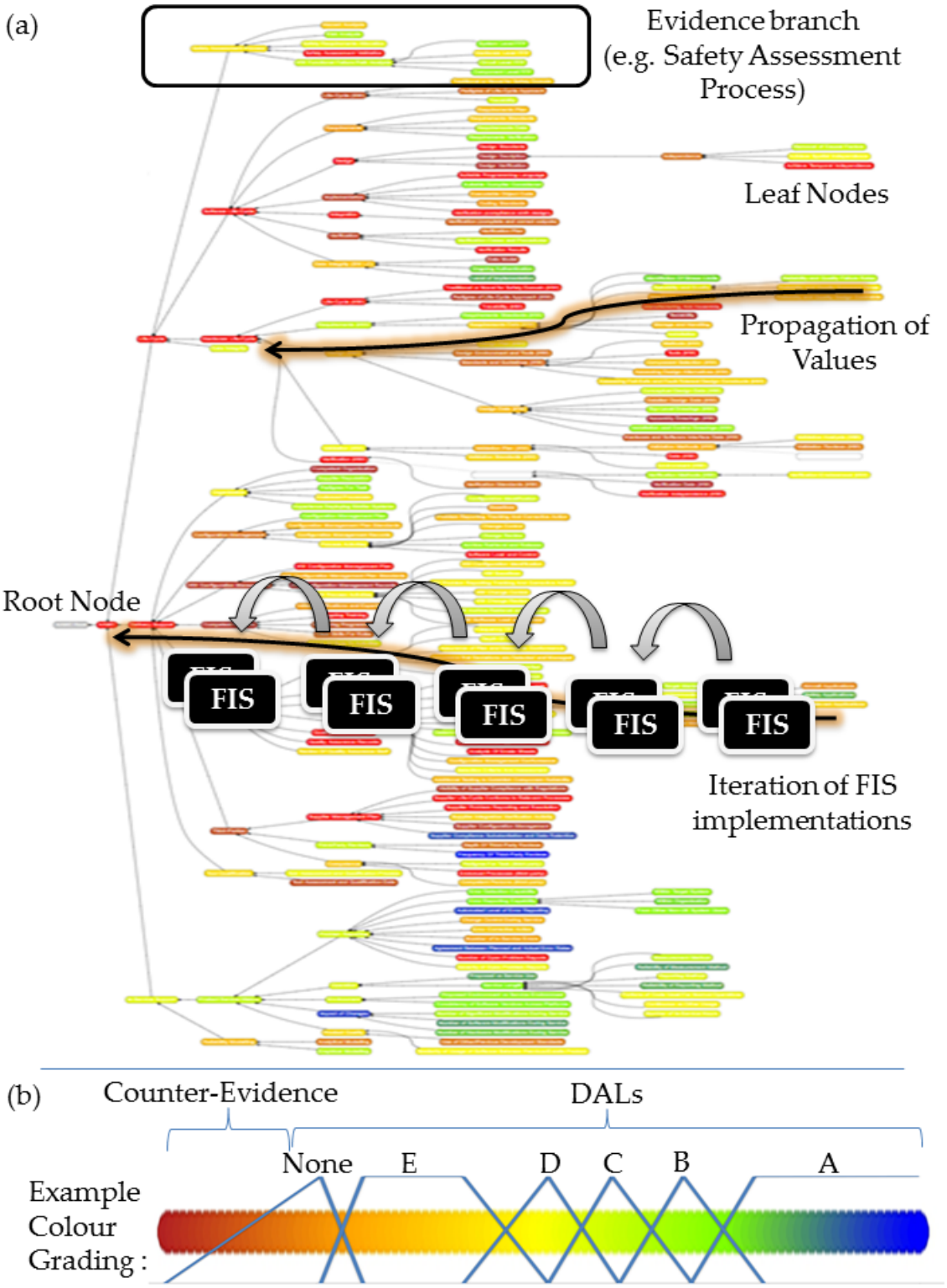


Figure 8.17: Example of Colour Grades for Nodes and Properties of the Linkage Tree

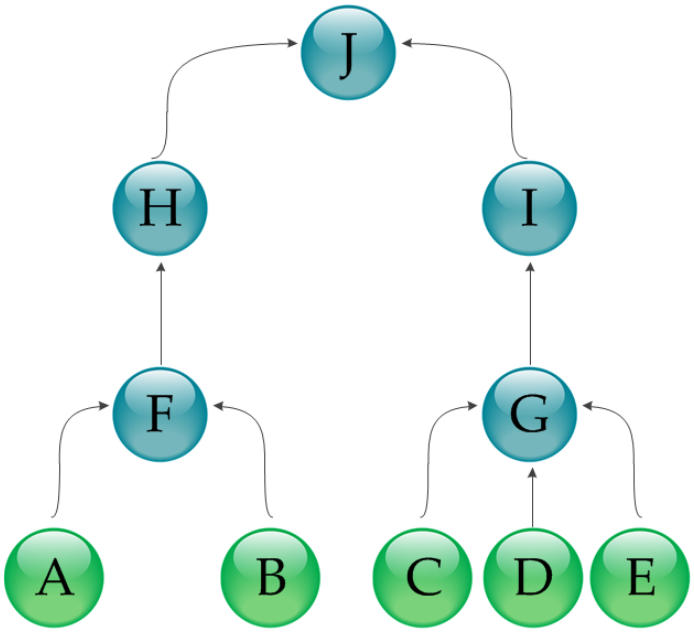


Figure 8.18: Tree to Illustrate Traversal Order

to be implemented. There are varying time implications for all evidential amendments/additions and therefore any decision or optimisation should take these into account.

- *Cost.* There is the cost to consider, in monetary terms, when making evidence amendments/additions. Traditionally, a task which takes an increased period of time will have correlating increased costs. However, this is not always the case as short-term activities can be relatively costly, e.g. performing MC/DC activities.

- *Quality.* For any evidential item there will be a limit to the quality which can be achieved and the risks associated with the evidence which may become apparent if further exploration is conducted. There needs to be recognition that with any activity counter-evidence may be discovered. The likelihood of counter-evidence being discovered will vary, e.g. development of planning artefacts to be in keeping with a DAL is a lower risk than additional testing activities being encountered.

These characteristics are to be captured within the framework as an *overhead* associated with each evidential item at the leaf nodes, see Figure 8.19. The framework will not explore the concept of evidence *overheads* further than the ability to define the overall value. This approach allows the framework to be fed the relevant values from external tools, e.g. Microsoft Project, whilst not limiting the factors to be considered. The topic of evidential *overheads* is one which would be worthy of further study, see the *Recommendations to Enhance Current Software Safety Assurance Processes* chapter.

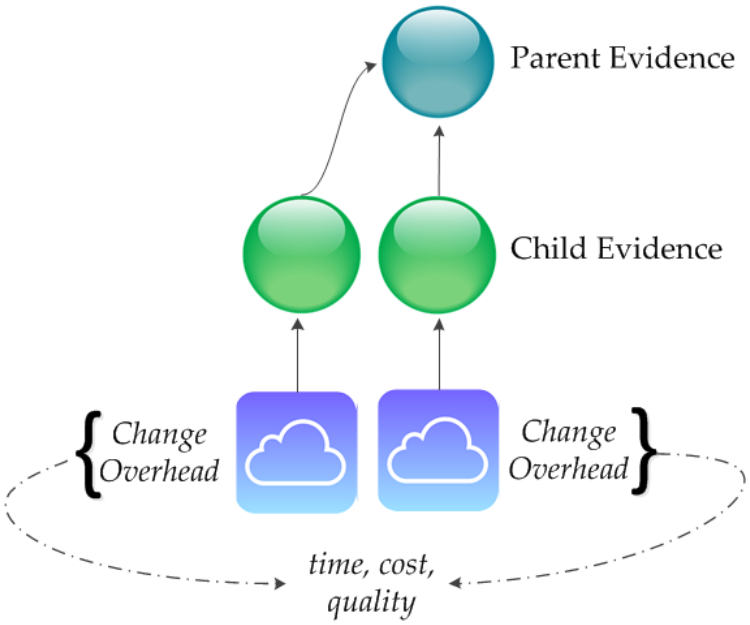


Figure 8.19: *Change Overhead* Associated with Child Nodes

8.2.8 Approach for Optimisation

A method to reason under uncertainty, e.g. to determine the overall degree of membership to the DALs, is only one element to the DSF. Another key feature of the DSF is to help decision makers to *determine a target* DAL for a system and to assist with decisions made to *reach the target*. This approach is to guide decision makers to use the differing states of evidence, e.g. ancillary, and to adjust the evidence attributes when deemed to reflect the 'real-world'.

To be of value to the DSF and the premise of the thesis, the approach chosen to perform the optimisation is expected to:

- Derive *sufficient* answers to assist with decision making.
- Allow the inclusion of a number of variables/factors to determine the 'strength' of a potential solution.
- Demonstrate the ability to determine more *valuable* results via convergence on options based upon globally available data.
- Allow refinements of the optimisation based upon inputs into the system.
- Allow useful and informative data to be extracted.

8.2.8.1 System Components of an Optimisation Problem

In essence, the process of optimisation is to select the best element, or set of elements, from a set of available alternatives. The concept of *optimisation* is part of a wider theory of evolutionary computing. The evolutionary approach is applied to automating the problem solving for a system. A system, in this context, consists of three components: *inputs*, *outputs*, and an *internal model* connecting the two (Eiben and Smith, 2003). The type of problem is determined by the system components which are known: *optimisation*, *modelling (or system identification)*, and *simulation* (Eiben and Smith, 2003).

Within an optimisation problem the internal model and the desired output are known. The task is to determine a suitable input. Figure 8.20 illustrates the system components states for an optimisation problem.

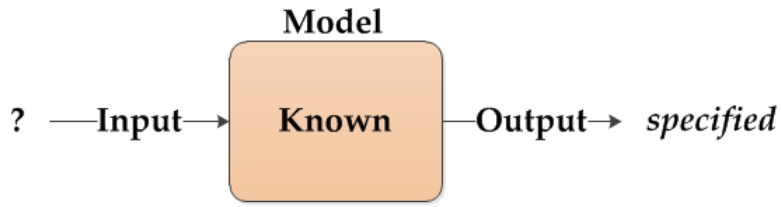


Figure 8.20: System Component States Within an Optimisation Problem (Eiben and Smith, 2003)

8.2.8.2 Optimisation Within Its Context

Once a suitable strategy has been devised by the decision makers on the types of evidence and the potential values of the evidence attributes to achieve a DAL then there is a need for the DSF to *optimise*. The optimisation can be applied to *some* or *all* of the evidence attributes, these act as the *inputs* to the optimisation problem. Optimisation will allow the decision makers to implement a *minimum* of changes *and* the most *effective* changes to achieve the target DAL.

8.2.8.3 Methods to Conduct Optimisation

As with the method to reason under uncertainty for the DSF the choice of the optimisation technique also needs to be fit for purpose and proportionate. The main tenet of the thesis is *not* to make clear recommendations on the most suitable search/optimisation techniques but to utilise an optimisation method which is *appropriate* for the problem.

There are a range of search/optimisation techniques which broadly fall within three categories:

- *Calculus Based.* Direct and indirect.
- *Random.* Guided and non-guided.
- *Enumeratives.* Guided and non-guided.

There are a great number of techniques and algorithms which fall under the three broad categories stated above; e.g. there are numerous "nature-inspired" algorithms such as *ant and bee*, *firefly*, *bat*, and *flower pollination* (Yang, 2018). Due to the DSF requiring to implement a *sufficient* optimisation technique it is not intended to conduct an exhaustive analysis of the techniques. A number of potential techniques are outlined below.

- *Genetic Algorithm (GA).* GA[42] are inspired by the principles of genetics and evolution. They attempt to mimic the reproduction behaviour observed in biological populations. The principal of *survival of the fittest* is adopted to select and generate individuals (design solutions) which are adapted to their environment (design objectives/constraints) (Haasan, Cohanim and De Week, 2005). Over a number of generations (iterations) the desirable traits (design characteristics) will evolve and remain in the genome composition of the population (set of design solutions generated each iteration) over traits with weaker undesirable characteristics (Haasan, Cohanim and De Week, 2005). Advantages of GAs include: greater success of finding global optimal[43]; do not require derivatives[44]; can be applied with both discrete and continuous parameters[45]; can be applied to complex and not well defined problems; bad solutions do not negatively affect the end solutions; and is very well established within the domain (Abdmouleh et al., 2017). Disadvantages include: can be time consuming for large and complex problems (repeated fitness function evaluation[46]); can suggest bad solutions; can be *trapped* into local optima[47]; and they can be inaccurate (Abdmouleh et al., 2017).
- *Simulated Annealing (SA).* SA[48] emulates the physical process of annealing; that of submitting a solid to high temperature, with subsequent cooling, to obtain high-quality

[42]Introduced in the mid-1970s by John Holland (Mitchell, 1996).
[43]A solution which is as good (or better) than *all* possible solutions.
[44]Derivative-based algorithms do not take into account multiple optima as they go to a local optimum near to where they started (Burns Statistics, 2018).
[45]*Discrete* data is that which can be counted and *continuous* data is that which can be measured.
[46]The concept of a *fitness function* is to determine how close a given design solution is to achieving the set aims.
[47]The relative best solution within a given neighbouring solution set.
[48]Originally proposed in the early 1950s as a method to model the natural process of solidification and formation of crystals (Lee and El-Sharkawi, 2008). Kirkpatrick, Gelatt and Vacchi (1983) and Cerny (1985) noted the physical process of annealing could be associated with some combinatorial optimisation problems (Lee and El-Sharkawi, 2008).

crystals. Defect-free crystals (solids with minimum energy) are more likely to be formed under a slow cooling process. The two main features of SA are: the transition mechanism between states and the cooling schedule (Lee and El-Sharkawi, 2008). Within combinatorial optimisation of complex problems, SA aims to find an optimal configuration (or state with minimum 'energy'). Advantages of SA: can be simple to implement; can provide good solutions for many combinational problems; and it can be robust (Abdmouleh et al., 2017). SA disadvantages include: may terminate in local minimum[49]; can have large computing time; cannot provide information on the level by which the local minimum deviates from the global minimum[50]; local minimum can depend on the initial configuration; and cannot given an upper bound for the computation time (Abdmouleh et al., 2017).

- *Particle Swarm Optimization (PSO).* PSO is initialised with a population of random solutions (called *particles*). With each particle having a *velocity* as they travel through the search space. The velocities are dynamically adjusted based upon their behaviours. A combination is made of the history of the particles own current, and best, locations with those of one or more members of the swarm. This process repeats to move closer to an optimum of the fitness function (Sarkar, Roy and Purkayastha, 2013). Advantages of PSOs include: being simple to implement; few parameters to adjust; parallel computation; robust; have higher probability and efficiency in finding the global optima; fast convergence; do not overlap and mutate; and can have short computational times (Abdmouleh et al., 2017). Disadvantages include: difficult to define initial design parameters; cannot work out the problems of scattering; can converge prematurely; and be trapped into a local minimum (especially within complex problems) (Abdmouleh et al., 2017).

- *Harmony Search.* Harmony Search is inspired from harmony improvisation with various pitches (inputs) being combined to reach a perfect harmony (output). The technique makes use of a Harmony Memory Considering Rate (HMCR) and Pitch Adjusting Rate (PAR) which are used to generate and further mutate a solution (Wang, Gao and Zenger, 2015). Advantages of the approach includes: no initial value settings are required (Wang, Gao and Zenger, 2015); can use discrete and continuous variables; cannot diverge; and may escape local optima (Abdmouleh et al., 2017). Disadvantages include: ability to search for local optima is weak (Wang, Gao and Zenger, 2015);

[49] A local minimum, also called a relative minimum, is a minimum within a neighbourhood (Weisstein, 2018*b*).

[50] A global minimum, also known as an absolute minimum, is the smallest overall value of a set, function, etc., over its entire range (Weisstein, 2018*a*).

can reach a high number of iterations; may encounter unproductive iterations without improving the solution; and can have a high dimensional multimodal problem (Abdmouleh et al., 2017).

- *Greedy Algorithm.* This approach is a problem solving heuristic which makes the local optimal choice *at each stage* with the aim of finding the global optimum. The algorithm chooses what appears to be the best option at any one step. There are advantages to this approach: finding a solution is easy and straightforward with a run time which is significantly reduced (Choudhary, 2018). However, there are numerous disadvantages: no possible alternatives are selected which means that if a wrong segmentation is reached the algorithm gets "stuck" in it (Ibanez, Santos and Berreira, 2006). Also, there is a need to work much harder to understand correctness issues and it is hard to prove why any given solution is correct (Choudhary, 2018).

As stated, the aim of the optimisation process within the framework is for it to be *fit for purpose* and *proportionate*. The thesis is not based upon generating a perfected or endorsed optimisation approach rather it is to determine and illustrate the *value* which optimisation can play in the adoption of a diverse evidence software safety assurance process. Within the software engineering and safety domain the GA optimisation approach has a level of pedigree. It is being implemented within areas for software/system reliability modelling (Hsu and Huang, 2010, Tian et al., 2009); software quality assurance (Suresh, 2015) and modelling (Drown, Khoshgoftaar and Seliya, 2009); test case/data generation (Dong and Peng, 2011, Lijuan, Yue and Hongfeng, 2012), testing efficiency (Khan and Amjad, 2016); software cost estimation (Li, Xie and Goh, 2007, Gharehchopogh, Rezaii and Arasteh, 2015); and security assurance (Dong et al., 2010). The use of GAs has provided value in each of these research areas.

It is clear that the use of GAs is not suitable for all instances of optimisation, e.g. there are benefits of Breeding Algorithms (BAs) over GAs (Xiao-ping, Shi-zhao and Xin-wei, 2008). However, GAs are deemed to be very effective and the research and application has been popular in the past (Yue et al., 2009) and continues to be so with a range of recent research adopting GAs. It is classed as being a useful piece in the puzzle (Cronin and Butka, 2018) with it comparing favourably to other optimisation approaches (Hristakeva and Shrestha, 2005, Kaewyotha and Songpan, 2018). GAs are also being adopted for modern novel research areas, e.g. Unmanned Aerial Vehicle (UAV) deployment (Cho and Kim, 2018).

8.2.9 Options to Assist Decision Making

A key element to the DSF is the requirement to allow decision makers to conduct *what-if* analysis and to explore scenarios. A rich set of options to perform such analysis will allow for a more varied and inclusive solution space. There is a need to ensure that the decision options are suitable and proportionate. Therefore, the options presented to the decision makers will be based upon logical steps to achieve potential outcomes.

It is envisaged that the following options will be provided to assist the decision makers. The options are not listed in a particular order; they represent a flow of decisions that will be user-determined. The options to assist with decision making can be formed into broad categories:

1. Counter-Evidence.

   (a) Determine if the current evidence contains *counter-evidence* once the node attributes (e.g. *confidence*) are calculated.

   (b) Allow the *counter-evidence* threshold to be user-defined (i.e. the *quality* level which constitutes negative evidence).

   (c) Allow the *counter-evidence* values to be displayed on the visualisation.

   (d) Allow nodes identified as being *counter-evidence* to be selected by the user with attributes refined with subsequent optimisation.

2. Attribute selection.

   (a) Allow attributes of valid nodes to be user-defined. Attributes of certain nodes are calculated by the FISs and cannot be user-defined. An example is the child node *confidence* which is calculated *from* the user defined *quality* and *contribution*.

   (b) Allow attributes to be selected for the various states of the evidence; e.g. setting the quality attributes for the nodes of interest for extant data.

   (c) Allow attributes to be chosen for one or more nodes/strands at any one time.

3. DAL selection/output.

   (a) Allow the *target* DAL to be user-defined for the evidence. The *target* DAL can subsequently be aimed for via the optimisation processes.

   (b) Show the DAL which is achieved with the current evidence.

   (c) Show the DAL which could be achieved via changes to node attributes.

(d) DAL display will show level of compliance to each DAL, i.e. the degree of MF for each DAL.

4. Optimisation.

   (a) Allow attributes to be selected for any valid node/strand and to optimise based upon user-defined attributes.

   (b) Allow *overhead* values for the nodes (e.g. time to implement any change) to be taken into account when the optimisation of the tree/nodes is performed. *Overhead* acts as a *penalty* for the node.

   (c) Method will derive a suitable value for the node attributes (if defined for consideration) which is between the existing attribute value and that which is user-defined. Optimisation will limit the level of change, i.e. minimal changes to be made to the existing value.

5. Visualisation.

   (a) Allow differences between the existing attributes of the evidence and the *what-if* calculations to be displayed. Will also show the calculated reasoning values based upon the attributes on the nodes.

   (b) Show the impact that a node attribute change has and the propagation this has within the tree structure.

   (c) Show all evidence contained within the tree and the current or status, e.g. *extant* etc.

   (d) Colour grade the tree nodes based upon the node confidence values.

   (e) Root node for the tree will display the DAL output values and the level of membership of each DAL.

   (f) Show the difference between current and potential node attribute values, indicated via percentage values.

   (g) Show *overhead* values for all nodes in the tree.

8.3 End-to-End DSF Process

The design decisions for the DSF results in an end-to-end implementation which allows SMEs to follow a structured flow in order to arrive at an *informed* qualification decision. Figure 8.21 shows the user flow.

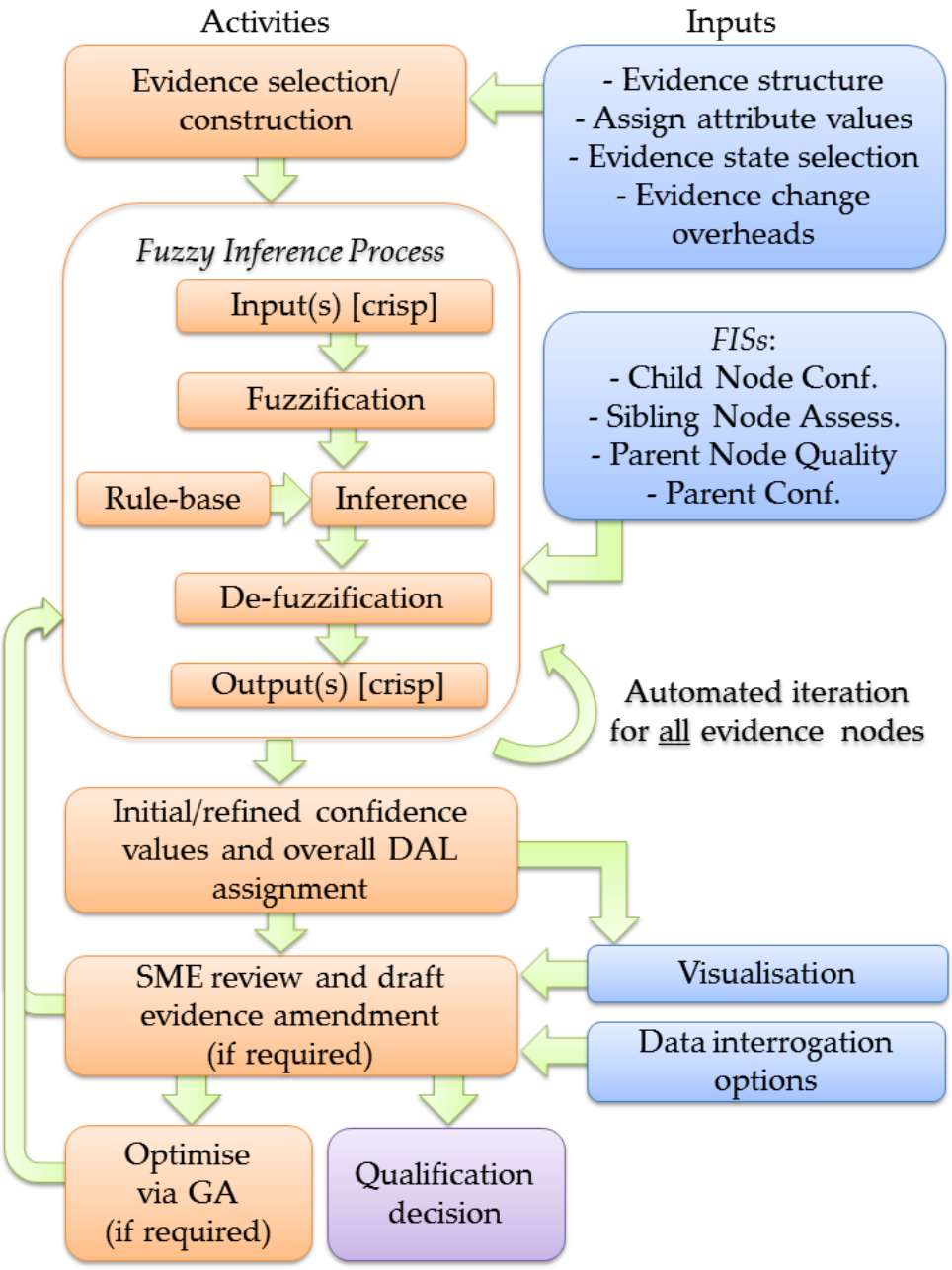


Figure 8.21: End-to-End DSF Process

8.4 Summary: Framework Design and Implementation Decisions

A design flow was devised which took into account a number of underpinning principles and features for the DSF. A set of characteristics of assurance evidence were devised which described the features of the evidence and how the evidence attributes were combined. The underpinning principles also related to the method to conduct the reasoning and the optimisation process. The features of the DSF allows stakeholders to explore scenarios and alternative solutions based upon the use of diverse evidence.

The states of the evidence are also defined to allow the differing values which may be placed on such data to be accounted for, e.g. regulators may place greater weight on evidence which supports a standard/guideline. Informative visualisations have been developed with a range of configurations to allow stakeholders to comprehend and make decisions in an efficient manner. A varied set of initial options have been devised to allow stakeholders to explore evidence sets and to amend/add/remove the nodes. This is in addition to the variety of evidence attribute value alterations.

The underpinning methods to calculate the diverse evidence values, i.e. the reasoning and optimisation approaches, are determined to be fit for purpose for the DSF. They assist with demonstrating the value of diverse evidence and allow judgements on evidence to be defined. The structure of the FIS and the subsequent MFs are also proportional and suitable for the problem.

The creation of the DSF allows for a number of case studies to be implemented to demonstrate the value of the DSF and the benefits which diverse evidence can have to support a PEs safety assurance argument.

Chapter 8 has partly responded to the research sub-question: *What is a suitable structure for software safety assurance evidence and can mathematically derived approaches inform how judgements are made on the evidence and for proposing alternative/optimised solutions?*.

Chapter 9

Case Studies, Exploratory Testing, and Evaluation of the DSF

The previous chapters outlined the justification for why there is a need for the DSF and provided the justification for the design decisions. The previous chapters provided the foundations to develop the framework itself.

The statements and decisions made in previous chapters[1] regarding the value and use of diverse evidence need to be placed under practical review. Findings can be derived via suitable case studies and by the use of exploratory testing. Importantly case studies can explore a research topic within its context (Saunders, Lewis and Thornhill, 2012). Exploratory testing also allows *deductive reasoning* to be adopted based upon previous findings which informs future tests (Mitrea, 2011). Case studies and exploratory testing are methods which are stated to have clear benefits to research and theory testing (Saunders, Lewis and Thornhill, 2012, Whittaker, 2009, Gerrard and Thompson, 2002). This chapter outlines the case studies (and additional activities), the results, and the observations from these exercises. An evalution of the DSF is also provided.

This chapter will examine:

- *Purpose and Aims of the Case Studies and Exploratory Testing.* The purpose of what the case studies will achieve and how they will assist with eliciting information for relevant observations.
- *Caveats and Scenario Selection.* Descriptions of the case studies which will be adopted to demonstrate the value of the DSF and the use of diverse evidence.

[1]From Chapter 2 (*Research Strategy*) through to Chapter 6 (*Current Permissible Evidence for Safety-Critical Software Assurance*).

- *Variable Types Changed as a Result of Case Studies and Exploratory Testing.* Information on the types of variables to be manipulated via the case studies and exploratory testing.

- *Potential Evidence Assessment Flow.* Information on a proposed evidence assessment flow intended to allow for the initial review and assessment to gather diverse evidence.

- *Case Study Results.* Information on the outcomes achieved when diverse evidence is adopted via the DSF. Also, how the DSF was utilised to inform the gathering of additional evidence.

- *Observations from Case Studies and Exploratory Testing.* Information on the observations from the case studies and exploratory testing. Observations are focussed on the relationships between the attributes of the evidence and how changes were propagated.

- *Evaluation of the DSF.* An evaluation of the DSF with observations from the expected findings and the actual case studies and exploratory testing results. As assessment is made of the DSF in relation to the research in the area of quantitive confidence judgements.

- *Summary: Case Studies and Exploratory Testing.* Salient points from the chapter's findings.

9.1 Purpose and Aims of the Case Studies and Exploratory Testing

Despite the misconceptions regarding the purpose and value of case studies[2] it is accepted that case studies are valid for exploring wider research phases and for testing propositions (Yin, 2003). In addition to conducting case studies, exploratory testing is also to be implemented to understand the relationships between the node attributes. Exploratory testing is a recognised technique within the software testing domain. The process generates tests of interest whilst being cognisant of the action taken and the subsequent impact (Kaner, Falk and Nguyen, 1999)[3].

The case studies and the exploratory tests have been chosen to investigate the utility of the DSF and the benefit of adopting diverse evidence for a range of *prototype/research*

[2]From a social science perspective research approaches should be adopted hierarchically and this thinking translates to other research domains. As an example, there is a misconception that case studies should be adopted for the early exploratory phase of research (Yin, 2003).

[3]Further information is contained within sub-section 9.5.5.

equipment. In essence, the choice of case studies is to explore the systems within the *wider solution space* contained within Figure 1.1 in Chapter 1. The case studies and exploratory tests are generated with the aim to:

- Respond partially or in full to a number of the findings within *Background and the Problem of Interest* (chapter 3) which highlighted why further work was required in the area of diverse software qualification evidence. The findings included, but were not limited to, the lack of ability to optimise results and the myriad of attributes which could be utilised within a quantitative confidence argument.
- Utilise the DSF options to propose and generate alternative diverse evidence arguments. The DSF provides the *tool* for the decision maker. Options to explore diverse evidence options include, but are not limited to, the selection of one or many evidence attributes[4] and for the provision to provide values with subsequent optimisation[5].
- Demonstrate the value of the overall reasoning under uncertainty approach and that the chosen method is fit for purpose.
- Demonstrate the value of optimisation within any decision making process and that the chosen method is fit for purpose.

It should be noted that the purpose of the case studies and exploratory testing (and the DSF itself) is *not* to optimise a perceived ideal scenario, e.g. to gain compliance to standards. It *is* to optimise the evidence which *exists* or can be obtained via the most *efficient* and/or *effective* changes whilst taking into account the overheads associated with any node/attribute alterations. A key principle is that the results are achieved via optimising, and not gaming[6].

9.2 Caveats and Scenario Selection

To meet the stated purpose and aims for the case studies a number of scenarios have been selected to allow suitable DSF features to be explored. They demonstrate the value and benefit to a measured diverse evidence approach via the use of suitable attributes.

The following should be noted regarding the pre-existing systems which have been subject to assessment via the DSF. *What follows is a reasonably comprehensive list of caveats - they*

[4]For example, *sufficiency*.

[5]For example, allow decision makers to propose values for attributes, such as *quality*. This will also allow decision makers to ascertain the amount of *quality* improvement to achieve a sufficient level of *confidence* in the evidence.

[6]See sub-section 7.3.1 for further information on the concept of *gaming*.

are important to ensure that there are no misconceptions about the information which is used for the research case studies.

- The pre-existing systems have been chosen as case studies to allow the principles of the DSF to be explored and for interesting observations to be gained. The systems are suitable for investigation as part of the defined *research* strategy.
- The inclusion of a system within the case studies should *not* be taken as an indication that the system is being considered as part of a current or future platform.
- The evidence gathered for the systems and the decisions which are informed by the DSF provides, in some instances, a *hypothetical* perspective on the evidence which could be expected for a qualification argument. This has been based upon SME feedback[7].
- The evidence reviewed and provided for the case studies has been gained in the context of conducting research to ascertain the utility of the DSF and the use of diverse evidence generally.
- The expected DALs for the systems are based on assessments which reviewed the *hypothetical* safety integrity levels of the systems. The DALs reflect the *potential* functionality which the system could provide within a *hypothetical* platform architecture.
- None of the pre-existing systems included within the case studies are within any UK in-service platforms and have been selected for research purposes. Any PSH associated with a system has been gained via fielded prototype testing and/or initial trials conducted by one or more nations.

Figure 9.1 illustrates the process adopted to identify suitable systems and the related evidence which can inform the case studies and explortory testing.

Despite the pre-existing systems not being considered for any current or future platforms there are further caveats to the data used in this chapter. Due to commercial sensitivities the following should be noted regarding the case studies and the examples:

- The names of any pre-existing systems have not been provided.
- The types of *hypothetical* platforms which the systems may be valid for have not been provided.

[7]SMEs were selected on their experience and knowledge of representative systems. Information was gained from those that had conducted a range of 5 software assurance assessments as this would allow their knowledge to be gained from a sample of systems and associated evidence.

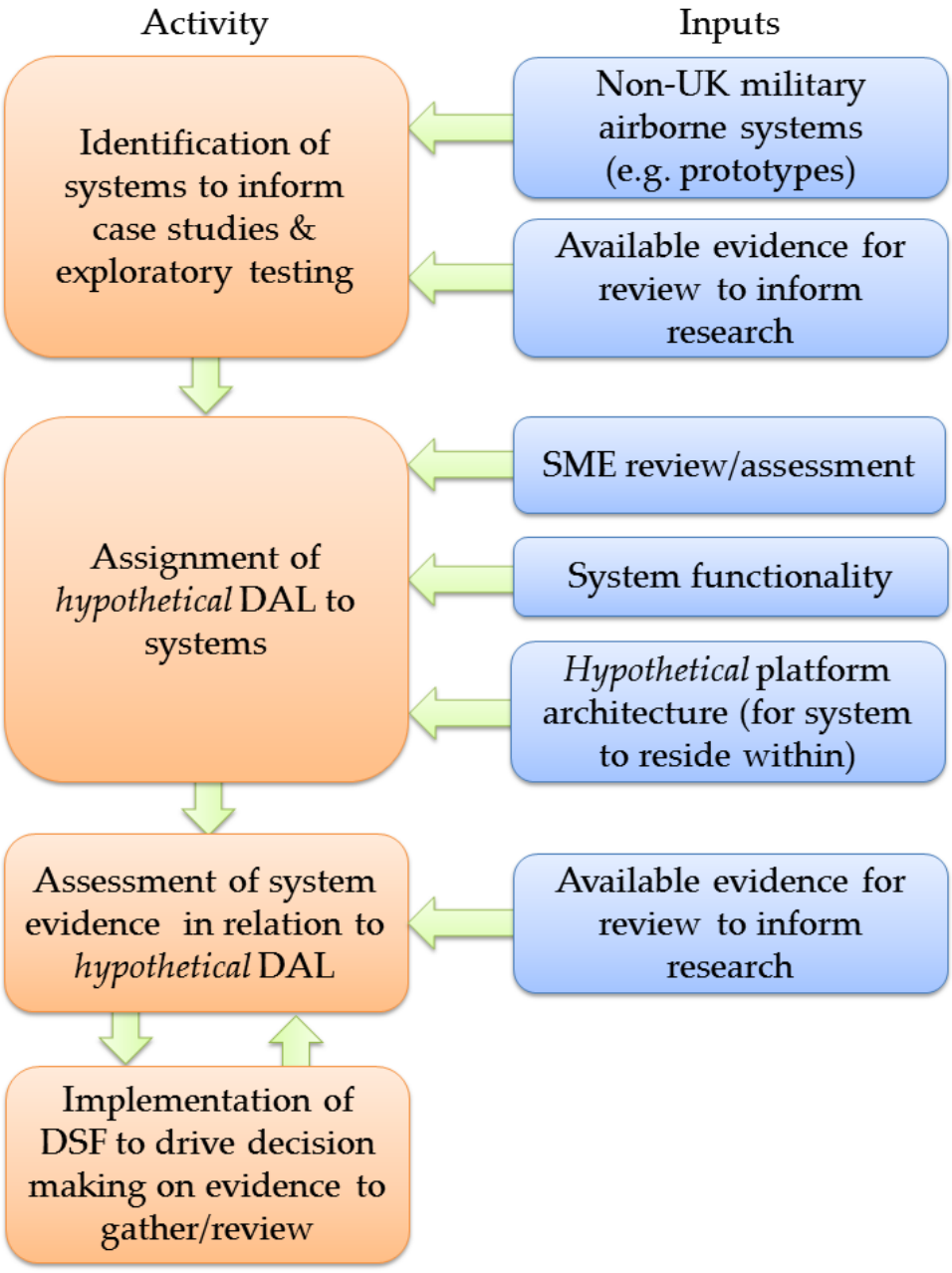


Figure 9.1: Process for the Identification of Pre-Existing Systems and the Related Evidence for Case Studies and Exploratory Testing

- No information is provided on *why* a particular initial evidence threshold was reached, e.g. software developed to differing standards.
- No information is provided on *why* only particular evidence would be available for the review, e.g. access control.
- The names and details of nations which own the systems are not provided. Where relevant they will be referred to as nation *X*, *Y*, etc.

The following case studies and exploratory tests are to be implemented:

1. *System A*. The system has been identified as being *hypothetically* DAL A LRU. The software was developed to DAL C. There is no CEH evidence available for review and there is a limited level of PSH which could be utilised within a diverse evidence argument.
2. *System B*. The system has been identified as being *hypothetically* DAL A LRU. The software and CEH were developed to suitable levels of rigour. There is additional evidence which could be utilised within a diverse evidence argument.
3. *System C*. The system has been identified as being *hypothetically* DAL B LRU. The software was developed to DAL D. CEH design logic information and CEH testing process evidence is available for review. There is a limited level of PSH which could be utilised within a diverse evidence argument.
4. *System D*. The system has been identified as being *hypothetically* DAL C LRU. The software was developed to a suitable level of rigour. There is no access to the software or CEH process information. There is a level of PSH from the use of the system within other contexts which could be utilised within a diverse evidence argument.
5. *Exploratory Testing*. In order to fully exploit the features of the DSF and to maximise the observations from the study it is intended for a number of further modelling/experiments to be conducted. These will focus on understanding the relationships between nodes within the structure and how changes to the attributes are propagated. A number of hypotheses are to be devised to act as initial ideas for experimentation/observation. The modelling is to inform the case study 'solutions' which aim to reach the perceived *hypothetical* target DAL of the system. Hypotheses include, but are not limited to, understanding such aspects as if small incremental improvements to evidence attribute values can have greater benefit than large improvements to fewer nodes, e.g. based upon *change overheads* associated with the evidence.

9.3 Variable Types Changed as a Result of Case Studies and Exploratory Testing

From a research perspective the case studies and exploratory tests are performed via a manipulation of variables, for example via the number of nodes (i.e. the evidence) or the attribute values (e.g. *quality*). In essence, the findings are as a result of a comparison or a correlational view of two types of variables (Salkind, 2010):

- *Independent Variable*[8]. Variables that are directly changed by the researcher or SME. In essence, the *cause* of any change(s).
- *Dependent Variable*. Variable that is changed due to the manipulation of the *independent* variables by the researcher or SME. In essence, the *effect* of any change(s) and what is being measured.

The structure of the evidence allows the variables to be identified at a *local* level so that changes can be made to an *evidence family* and the linked immediate nodes (subtrees and ancestor). Changes to these nodes can then allow observations to be made on the wider diverse evidence. There are a number of attributes which can be manipulated within an *evidence family* and the immediate nodes in order to measure changes.

There are a range of methods to instigate changes within a DSFs evidence tree. Within this chapter the variables for change are stated in relation to a particular evidence (leaf) node under review - this is termed the *Node of Interest (NoI)*. This terminology allows consistent comparisons. It is this relationship which was adopted in order to derive the hypotheses for further evaluation. The evidence structure and attributes (both *independent* and *dependent*) are in relation to the *NoI*.

Table 9.1 contains a sample of the nodes which are within a standard evidence tree. The Table states the nodes and the associated attributes which can be altered to influence the overall *confidence* (and DAL membership) of the ancestor node. It is these nodes/attributes which were reviewed and assessed as part of the case studies and modelling/experiments. This is in addition to the data category types (e.g. *extant* and *ancillary*) which form part of any 'solution'.

Figure 9.2 illustrates this concept further. The concept also acts as a *pattern* for the wider variable comparison/correlation observations within the diverse evidence structure.

[8]Note that the term *independent* in this context is different to that defined for an evidence attribute such as *quality*.

Node	**Attributes**
Node of Interest	*Quality* *Contribution* *Change Overhead* *Confidence*
Parent of Node of Interest	*Sufficiency* *Independence* *Contribution* *Confidence*
Siblings of Node of Interest	*Quality* *Contribution* *Change Overhead* *Confidence*
Parent of Subtree	*Sufficiency* *Independence* *Contribution* *Confidence*
Child Nodes of Subtree	*Quality* *Contribution* *Change Overhead* *Confidence*
Ancestor of Subtree Parent and Node of Interest Parent	*Sufficiency* *Independence* *Contribution* *Confidence*

Table 9.1: Variables for Change - Identified Nodes and Associated Attributes

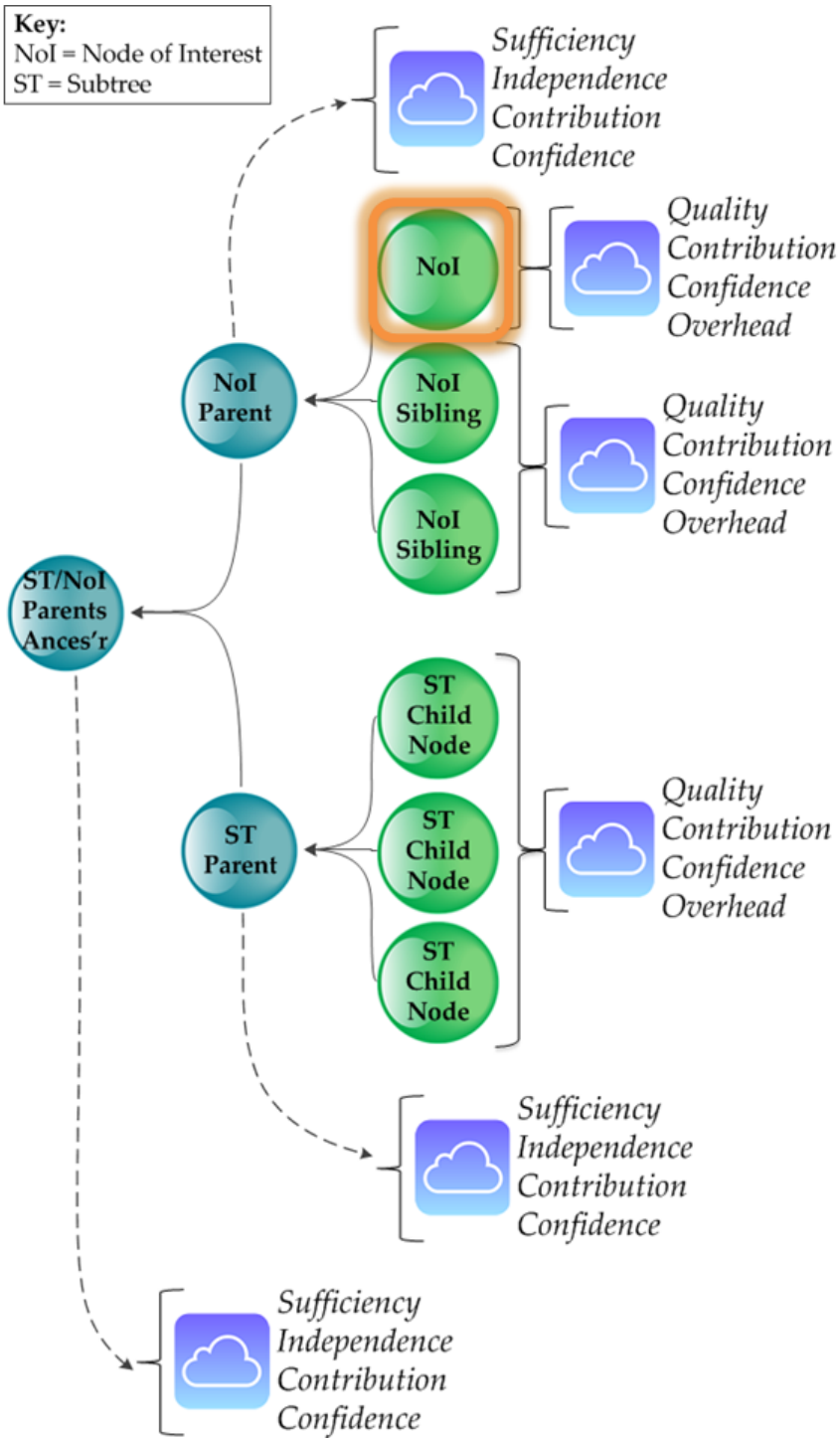


Figure 9.2: Independent and Dependent Variables in Relation to a NoI

9.4 Potential Evidence Assessment Flow

In order to devise a consistent approach to gathering and assessing evidence, a structured flow was created. This 'flow' was to purely act as an initial method to determine and assess the steps within the process to gather diverse evidence. Therefore, the case studies and exploratory tests purposefully did not fully adhere to the initially determined 'flow'.

Figure 9.3 shows the extent of the process flow to devise alternative evidence approaches. It includes directions to fully utilise the visualisation and optimisation options as part of the DSF approach. In essence, the initial proposed process flow states the following:

1. Guidance on the data population, i.e. the nodes and their attributes for extant, obligatory, and ancillary data. Population of Comma-Separated Values (CSV) files for import into the DSF.

2. Execution of the DSF to obtain initial confidence and DAL membership values, via the FISs.

3. Perform a Counter-Evidence (CE) check and direction for removal via *quality* attribute change(s), use of visualisation techniques for impact comparison, and assessment of CE for subsequent reduction. If the target DAL is not achieved then further steps are required. If the target DAL is reached then perform optimisation (see item 6).

4. The nodes and attributes are then *reviewed* for their impact and subsequently *amended*. For *each* of the steps within Table 9.2 the following activities are conducted:

   (a) The execution of the DSF.

   (b) Pre- and post-change comparison via visualisation methods.

   (c) Review of the revised overall confidence level.

   (d) If the target DAL is not achieved then further steps are required, i.e. the next step stated in Table 9.2.

   (e) If the target is reached then perform optimisation (see item 6).

   The steps within Table 9.2 are undertaken for the node/attribute reviews. The activities in the list above are conducted for *each* step within Table 9.2 to assess the impact of any changes.

5. If the target DAL is not achieved then direction is provided to assess additional evidence and for risk assessments to be conducted to address or tolerate evidence shortfalls.

No.	Node	Attribute(s)	Data
1	Leaf	*Quality*	Extant
2	Parent	*Sufficiency*	Extant
3	Parent	*Sufficiency* and *Independence*	Extant
4	Leaf	*Quality*	Obligatory
5	Leaf	*Quality*	Ancillary
6	Parent	*Sufficiency*	Obligatory and Ancillary
7	Parent	*Sufficiency* and *Independence*	Obligatory and Ancillary

Table 9.2: Steps for Node/Attribute Reviews

6. If the target DAL can be achieved for any of the above stages then direction is provided to set if evidence *change overheads* are to be considered, execution of DSF, potential pre- and post-change comparison via visualisation, perform optimisation (via GA) and review optimised data to determine suitable evidence to gather/contract. Reassess the node contributions if required.

It is recognised that for each *branch* within the structure, e.g. life-cycle, there will be a significant number of leaf and parent nodes to assign attribute values to. As an example, within the case studies there are over 100 nodes just for the life-cycle branch[9]. This is illustrated within Figure 9.4 which shows (a) the supporting *life-cycle* evidence which has been assessed by SMEs and has suitable values assigned to the node attributes (e.g. leaf-node *quality*); (b) the FISs which are implemented and repeated for the *life-cycle* branch to derive the branch FIS output values; (c) the *life-cycle* branch total which is an output to inform a case study and is based upon the supporting activities within the previous steps; and (d) the context of the branch output and supporting evidence of the FISs in relation to the data tree. Steps (a), (b), and (c) are repeated for each of the evidence branches, e.g. Delivery Support (DSP), PSH, etc. This illustrates the scale of producing the evidence values.

9.5 Case Study Results

The established principles, devised case studies, and the accepted exploratory testing allowed a range of interesting results to the obtained. The following sub-sections state the evidence initially established for the *hypothetical* systems and the subsequent additional evidence gathered. The evidence stated and the *confidence* gained to reach a threshold are developed incrementally. The overall evidence *confidence* is used to derive the DAL membership via

[9]In addition to, for example: >140 nodes for the security-related airworthiness branch; >90 nodes for the delivery support branch; >40 nodes for the testing branch; and >40 nodes for the PSH branch.

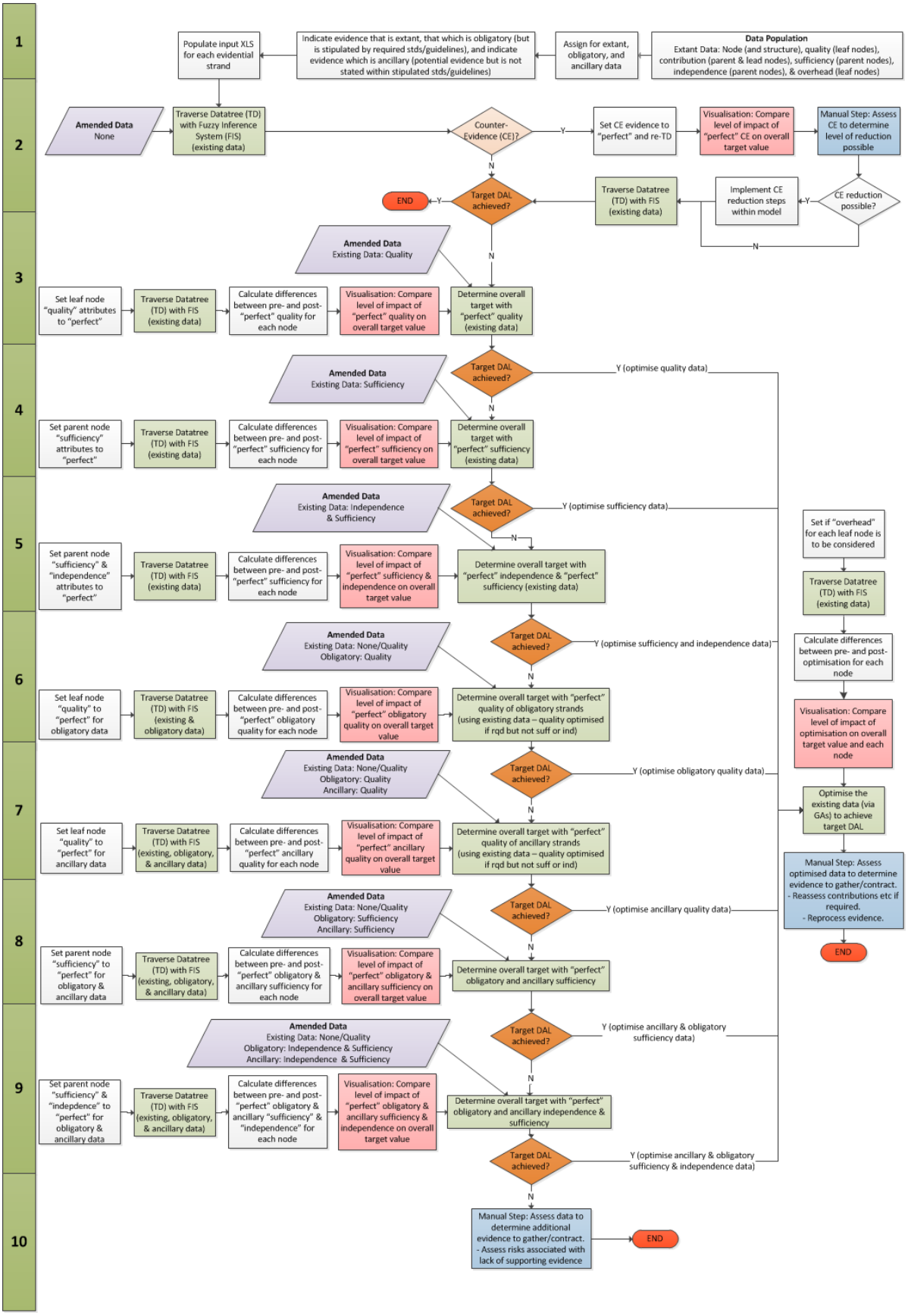
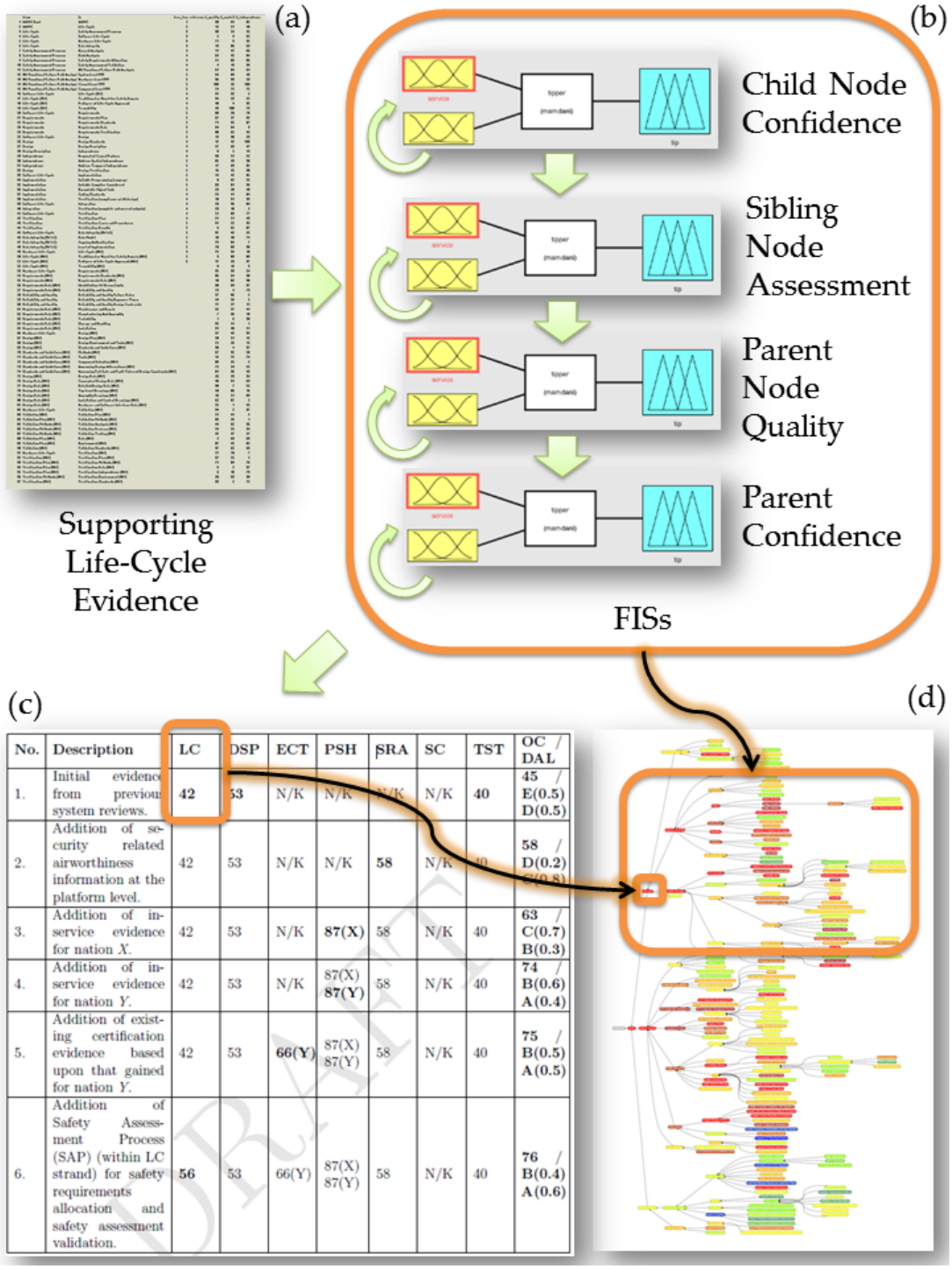


Figure 9.3: Potential Evidence Assessment Flow



No.	Description	LC	DSP	ECT	PSH	SRA	SC	TST	OC / DAL
1.	Initial evidence from previous system reviews.	42	53	N/K	N/K	N/K	N/K	40	45 / E(0.5) D(0.5)
2.	Addition of security related airworthiness information at the platform level.	42	53	N/K	N/K	**58**	N/K	40	58 / D(0.2) C(0.8)
3.	Addition of in-service evidence for nation *X*.	42	53	N/K	**87(X)**	58	N/K	40	63 / C(0.7) B(0.3)
4.	Addition of in-service evidence for nation *Y*.	42	53	N/K	87(X) **87(Y)**	58	N/K	40	74 / B(0.6) A(0.4)
5.	Addition of existing certification evidence based upon that gained for nation *Y*.	42	53	**66(Y)**	87(X) 87(Y)	58	N/K	40	75 / B(0.5) A(0.5)
6.	Addition of Safety Assessment Process (SAP) (within LC strand) for safety requirements allocation and safety assessment validation.	**56**	53	66(Y)	87(X) 87(Y)	58	N/K	40	76 / B(0.4) A(0.6)

Figure 9.4: Activities to Generate a Single Evidence Branch for a Case Study

the root node which may be used to compare against any *hypothetical* target DAL for the system. In the tables which follow the values indicate the level of confidence for the specific evidence strands (e.g. Life-Cycle (LC)) and the overall confidence level (i.e. OC) which takes into account *all* of the evidence strand values for that instance.

It should be noted that although the information has been gathered for research purposes for *hypothetical* systems and DALs there are still commercial sensitivities associated with the data. Therefore, the precise data related to each of the individual evidence nodes cannot be provided. However, information is provided on the confidence gained for each of the *branches*, e.g. life-cycle information and PSH if relevant. The inability to disclose this information still allows the concepts and DSF implementation to be illustrated for context to the findings/observations.

It is *not* intended for each of the case studies to provide an action-by-action account of the activities undertaken to increment the evidence confidence. The sub-sections which follow provide the incremental results and a narrative of salient points from the exercises.

Within the case study results the *branch* names are abbreviated to the following. It should be noted that the branches include evidence sub-types which were discussed in previous chapters (e.g. data integrity).

- Life-Cycle: LC.
- Delivery Support: DSP.
- Existing Certification: ECT.
- Product Service History: PSH.
- Security Related Airworthiness: SRA.
- System Complexity: SC.
- Testing: TST.

The results also include a number of further abbreviations for readability purposes:

- Overall Confidence: OC.
- Not Known: N/K.
- Design Assurance Level: DAL.

9.5.1 Case Study 1: System A

System *A* was assigned as a *hypothetical* DAL A. Table 9.3 contains the initial evidence confidence (in row No. 1). The subsequent additional evidence which was gathered is stated (in row No. 2 onwards) and the associated confidence levels for the *branch* and the *overall* confidence. The initial branch confidence values and the subsequent changes to the evidence branches are shown as **red and bold**.

No.	Description	LC	DSP	ECT	PSH	SRA	SC	TST	OC / DAL
1.	Initial evidence from previous system reviews.	**42**	**53**	N/K	N/K	N/K	N/K	**40**	**45 / E(0.5) D(0.5)**
2.	Addition of security related airworthiness information at the platform level.	42	53	N/K	N/K	**58**	N/K	40	**58 / D(0.2) C(0.8)**
3.	Addition of in-service evidence for nation *X*.	42	53	N/K	**87(X)**	58	N/K	40	**63 / C(0.7) B(0.3)**
4.	Addition of in-service evidence for nation *Y*.	42	53	N/K	87(X) **87(Y)**	58	N/K	40	**74 / B(0.6) A(0.4)**
5.	Addition of existing certification evidence based upon that gained for nation *Y*.	42	53	**67(Y)**	87(X) 87(Y)	58	N/K	40	**75 / B(0.5) A(0.5)**
6.	Addition of Safety Assessment Process (SAP) (within LC strand) for safety requirements allocation and safety assessment validation.	**56**	53	66(Y)	87(X) 87(Y)	58	N/K	40	**76 / B(0.4) A(0.6)**

Table 9.3: System A - Incremental Evidence Results

Salient points from the generation of the case study are as follows:

1. The initial available evidence for the system related to the software Life-Cycle (LC), DSP, and Testing (TST). This was pre-existing evidence which had been reviewed

previously as part of wider research activities. The evidence obtained for the initial assessment was from previous reports written by SMEs. A review of the pre-existing evidence was conducted to determine attribute values for each of the evidential types. The process involved stating leaf node *contribution*, *quality*, and *change overhead*. The parent nodes were reviewed to state the *sufficiency*, *independence*, and *contribution* values. The review was recorded for import into DSF.

2. Due to wider research efforts for the system of interest a number of security related airworthiness activities were being reviewed. The activities were at an *aircraft* (platform) level and not specific to the system of interest. Evidence from these activities were leveraged for inclusion within the incremental diverse evidence argument. The proposed nodes and attributes were generated and assessed for import into the DSF. An activity was conducted to ascertain the benefit of conducting *system* level security related airworthiness reviews in addition to that performed at the *aircraft* level. This exploratory assessment was conducted via the use of the visualisation approaches and optimisation as part of the DSF. Any addition of evidence for the Security Related Airworthiness (SRA) at the *system* level would require further activities.

3. Due to the *change overheads* associated with any SRA alterations a more efficient and effective method to gain overall confidence (in comparison) was to make use of the PSH evidence which would exist for the system. This would involve the *gathering* of evidence rather than the *generation* of evidence[10]. The PSH information was captured for nation X which the system is being trialled by. This information was imported into the DSF.

4. The system of interest is being trialled by more than one nation (i.e. not just nation X). Another nation (nation Y) was approached to ascertain the level of information which they hold which could inform an additional PSH. This approach was considered as there was an awareness of the experience hours gained by nation Y and the *change overheads* that would be associated with capturing this information. The addition of PSH from nation Y was considered due to the stated contribution that PSH had within the framework and the perceived support of PSH by the stakeholders. The framework output and knowledge of the perceived benefits of gaining nation Y PSH allowed stakeholders to have awareness of the level of confidence that would be gained by such information gathering. The information to be captured as part of the nation Y PSH was visualised and optimised to guide the data to be requested.

[10]The distinction being in this context that evidence that already exists needs to be *gathered*, whereas evidence that could be obtained (but does not exist) needs to be *generated* - a subtle but important point when *change overheads* are considered.

5. At this stage the system evidence consists of LC, SRA, and PSH from two nations (X and Y). This provided a reasonable level of confidence but the overall DAL membership was still towards DAL B (acknowledging that the DSF outputs are to be used as a *guide* and not to be a substitute for expert judgement). Further *what-if* analysis was conducted based upon perceived *change overheads* for node attribute alterations. An incremental increase in *confidence* was discovered to be possible via the addition of Existing Certification (ECT) evidence. Due to the qualification status of the system within nation Y it was determined that ECT data could be gathered. The value and availability of the ECT information prompted the use of the evidence type as part of the wider research and to mitigate a number of systems within the research.

6. Dialogue with stakeholders resulted in the existing evidence being proposed and in principle deemed sufficient. However, a stakeholder expectation was for safety assessment information to be generated to provide further *confidence* in the LC information which was gathered. The benefit of gaining the evidence was determined via the DSF with the perceived incremental *confidence* level established. The perceived *confidence* level informed the proportionate effort to gather the SAP information.

9.5.2 Case Study 2: System B

System B was assigned as a *hypothetical* DAL A. Table 9.4 contains the initial evidence confidence and the subsequent additional evidence gathered.

No.	Description	LC	DSP	ECT	PSH	SRA	SC	TST	OC / DAL
1.	Initial evidence from dedicated system review.	**77**	**70**	N/K	N/K	N/K	N/K	**71**	**75 / B(0.5) A(0.5)**
2.	Addition of security related airworthiness information at the platform level.	77	70	N/K	N/K	**58**	N/K	71	**79 / B(0.1) A(0.9)**

Table 9.4: System B - Incremental Evidence Results

Salient points from the generation of the case study are as follows:

1. This is an interesting case study from the perspective that the system already had a software and CEH development status which was deemed to be suitable in relation to the defined *hypothetical* DAL. A detailed assessment had been conducted on the

system software and CEH activities as part of ongoing research efforts. The detailed assessments and subsequent reports were written by SMEs. The reports were reviewed and assessed in order to generate imports to the DSF. The process involved stating leaf node *contribution*, *quality*, and *change overhead* values. Parent nodes were reviewed to state *sufficiency*, *independence*, and *contribution* values. The pre-existing reports outlined that minor shortfalls existed in the system evidence and this was reflected in the imported data into the DSF.

2. Due to wider activities associated with the research there were a number of SRA reviews being completed at an *aircraft* level (not *system* level specific). To take into account these activities and to gain consistency with other systems the SRA information was included within the DSF.

3. A form of *what-if* analysis was conducted using visualisation and optimisation approaches offered by the DSF. This was to establish the benefit of additional evidence with the perceived knowledge of the *change overheads* associated with the *obligatory* and *ancillary* data. An initial CBA established that there would be *limited* benefit to conducting these further activities.

4. Due to the perceived LC and TST compliance to the *hypothetical* target DAL a relatively high initial confidence level and DAL membership was established. However, the addition of wider diverse evidence does have the ability to incrementally improve the overall confidence level. The impact of which is dependent on the attributes that the additional evidence has, e.g. *contribution*. Wider research activities which relate to the *system* level can have benefits which the DSF can capture to incrementally increase the overall confidence level. At a *system* level the benefits may be minimal if perceived *hypothetical* target DALs are already met. However the adoption of such approaches would allow a consistent qualification approach to be achieved.

9.5.3 Case Study 3: System C

System *C* was assigned as a *hypothetical* DAL B. Table 9.5 contains the initial evidence confidence and the subsequent additional evidence gathered.

Salient points from the generation of the case study are as follows:

1. For this system there was a relatively reduced level of available information for review. For this research the SMEs were required to conduct interviews with commercial SMEs and to have a number of follow-up queries to establish suitable evidence. The reports generated by the SMEs were reviewed and assessed to generate imports to the DSF. The

No.	Description	LC	DSP	ECT	PSH	SRA	SC	TST	OC / DAL
1.	Initial evidence from dedicated system review.	**50**	**47**	N/K	N/K	N/K	N/K	**40**	**45 / E(0.5) D(0.5)**
2.	Addition of PSH from known nation due to system use within prior prototype platform.	50	47	N/K	**55**	N/K	N/K	40	**51 / D(0.9) C(0.1)**

Table 9.5: System C - Incremental Evidence Results

process involved stating leaf node *contribution*, *quality*, and *change overhead* values. The parent nodes were reviewed to state *sufficiency*, *independence*, and *contribution* values. A number of shortfalls were identified in the reports which were captured within the DSF. This concluded that a *hypothetical* DAL E/D could be established for the initial evidence.

2. The system had been trialled for a known period of time within nation *Z* so the reduced *change overheads* associated with such data allowed PSH information to be gathered. However, the level of information available for the PSH evidence was limited and this is reflected in the DSF.

3. Due to the level of evidence available to establish confidence there would be limitations placed on the system. The DSF was used to attempt to establish additional diverse evidence to not require these limitations. A number of *what-if* scenarios were conducted to visualise and optimise any findings which could be used to inform recommendations. However, it was established that evidence would need to be *generated* (rather than *gathered*) and therefore, there would be significant *change overheads* to consider. The inability to gain any further information confirmed there would be limitations with the evidence (and the system). This exercise confirmed that are limits to what the evidence can provide in terms of building confidence in a given system. A system such as this, if *hypothetically* brought into service, may operate with CLE.

9.5.4 Case Study 4: System D

System *D* was assigned as a *hypothetical* DAL C. Table 9.6 contains the initial evidence confidence and the subsequent additional evidence gathered.

Salient points from the generation of the case study are as follows:

No.	Description	LC	DSP	ECT	PSH	SRA	SC	TST	OC / DAL
1.	Initial evidence from dedicated system review.	**38**	**42**	N/K	N/K	N/K	N/K	N/K	**39 / E(1)**
2.	Addition of existing certification evidence from a relevant domain.	38	42	**63**	N/K	N/K	N/K	N/K	**49 / E(0.1) D(0.9)**
3.	Addition of PSH evidence from use within a relevant domain.	38	42	63	**82**	N/K	N/K	N/K	**57 / D(0.3) C(0.7)**

Table 9.6: System D - Incremental Evidence Results

1. The system had a set of initial evidence related to the LC and DSP. This assessment was based upon evidence which was not fully compliant with the extant standards. An assessment of the available LC and DSP evidence allowed imports to be generated for the DSF. This process involved stating leaf node *contribution*, *quality*, and *change overhead* values. Parent nodes were reviewed to state *sufficiency*, *independence*, and *contribution* values. The initial evidence did offer a level of confidence which was captured within the DSF.

2. The system of interest was being trialled within a related domain by nation *W*. Due to the level of hours gained via related experience there were relatively low *change overheads* associated with the ECT evidence. This evidence was captured within the DSF. The analysis showed that further evidence was required to reach the *hypothetical* target DAL.

3. Further evidence was gathered in the form of PSH due to the associated *change overheads* linked to the fact that existing certification had been gained (captured within the ECT). A robust PSH evidence assessment allowed a reasonable DAL C membership to be gained. CBA showed that additional activities to increase the confidence and the DAL membership were *not* warranted.

9.5.5 Purpose of Exploratory Testing

In addition to the structured case studies it was very beneficial to conduct a number of smaller and less structured models/experiments to understand further the behaviour of the DSF. The case studies provided a rich set of results which adopted key features of the

DSF with the use of the visualisation techniques to assist with analysing the evidence. The optimisation features were also utilised to assist with *what-if* analysis to determine evidence to gather/generate. Exploratory testing allowed deductive reasoning to be applied to understand the evidence attributes and the node relationships in greater detail.

The aim of the exploratory testing was to establish how alterations to certain variables (stated within *Variable Types Changed as a Result of Case Studies and Exploratory Testing* - subsection 9.3) reacted within formulated scenarios. There were a large number of variables to explore and to ascertain how any alterations impacted the immediate evidence family, subtree/ancestor, and also the wider evidence within the structure.

Exploratory tests investigated, but were not limited to, the following:

- Is it less of an *overhead* to correct the counter-evidence of a node or to improve the *quality*, *independence*, or *sufficiency* of the sibling nodes?
- Is there a direct correlation between an increase in *overhead* 'budget' and the overall *quality* which can be achieved?
- Are there cases where it is better to add evidence rather than improve the *quality*, *independence*, or *sufficiency* of the existing evidence, i.e. less *overhead*?
- Is it best to add nodes to improve *sufficiency* (but not improve *quality*) or to add nodes which are not *independent* (with no *quality* improvements)?

This proved a very useful activity with observations captured which were, importantly, repeatable.

9.6 Observations from Case Studies and Exploratory Testing

This section contains a number of observations made during the implementation of the case studies and the exploratory testing exercises. They result in actions that need to be considered when SMEs are devising diverse evidence strategies. The observations are made on the relationships between the *attributes* of nodes and how alterations/addition of attribute values can impact the immediate and wider evidence set values.

1. If there is an improvement in the ability to manage node overheads (i.e. there is an increase in the *change overhead* value which can be included within any calculations) this does not automatically equate to an increase in the child/parent *confidence*.

(a) Improving the *quality* of nodes or the addition of nodes to increase parent node *sufficiency/independence* can lead to 'expensive' *change overheads*. Lower *change overheads*, but with larger benefits, can be made via the correct node selection, e.g. *contribution* based.

(b) Improvements to any nodes need to be *targeted* to ensure that it is aimed at the appropriate evidence, e.g. the node has high *contribution* and/or addition of *sufficiency/independence*.

(c) An improvement in the ability to process greater *change overhead* values, e.g. additional funds, is valid for counter-evidence or areas which need specific attention for localised improvements rather than overall confidence building.

2. Stakeholders may require a *balanced* set of evidence which contains a range of evidential types, i.e. a set of evidence which has *independence*.

(a) If the parent nodes are of low *confidence* then improve their *sufficiency* and *independence* values. This also improves the overall evidence confidence, i.e. the DAL.

(b) If the *contribution* of the child and parent is high then it is preferential to *improve* the *quality* of the existing evidence rather than add evidence, i.e. that with less *change overheads*.

(c) If the existing evidence has a low *contribution* then it is preferential to *add* evidence rather than improve the *quality* of the existing evidence; *if* additional evidence improves the *sufficiency/independence* of the parent node.

3. Improving the *sufficiency/independence* of a parent node can have greater impact than the addition of higher *quality* child nodes.

(a) If the existing evidence is maximised by improving the parent/child nodes that have high *contributions*, e.g. improved child node *quality*, then further improvements can be achieved via the addition of evidence.

(b) The addition of child node evidence *does not* need to increase the average *confidence* of the existing child nodes; i.e. the additional evidence *quality* does not have to be as high as the existing evidence. However, the child evidence *does need to* improve the *sufficiency/independence* of the parent.

(c) An increase in the average *quality* of the child nodes can increase the parent *confidence*; however, the level of the *quality* increase needs to be *significant* to have any substantial impact.

(d) Table 9.7 shows a very simplified set of attribute relationships with a single parent node with four leaf nodes. The examples result in *very* modest increases to the *confidence* value; however the example serves the purpose to illustrate the observation. Deviations from *observation (a)* within Table 9.7 in the subsequent observations, *(b)* and *(c)*, are shown as **red and bold**.

i. *Observation (a)* within Table 9.7 shows the *confidence* value (54) for a simple structure with *medium/moderate quality/contribution* values.

ii. *Observation (b)* shows an increase in *confidence* (by nearly 2%) if the leaf nodes improve the *quality* from *medium* to *high*.

iii. However, it is shown within *Observation (c)* that *medium quality* with *high contribution*[11] evidence which *improves* the *sufficiency/independence* of the parent (e.g. from *medium* to *high*) has a greater impact on the *confidence* improvement (by over 9%). Any *change overheads* would also need to be taken into account.

4. The *contribution* of any evidence (child or parent) is paramount.

(a) Child nodes may have high values for *quality* and parents may have high values for *sufficiency/independence*, however for the parent node it is the *contribution* which can have a significant impact on the *confidence* value.

(b) Increases in node *confidence* are observed for those nodes with high values for *sufficiency/independence* attributes. However, in these instances the sibling nodes of the parent will also have to increase to gain benefit; i.e. there is a need to increase child nodes *and* parent node attributes (but not the *contributions*) to observe increases.

(c) For a parent with high *sufficiency/independence* it is still the *contribution* which impacts the *confidence* the most.

5. It can be more effective to make small improvements to a larger number of evidence *quality* and/or parent *sufficiency/independence* values than to make significant changes to fewer items of evidence.

(a) A key element to consider for node improvement is that of *contribution* and *change overhead* values.

[11] Noting that there is a correlation between supporting evidence being deemed *sufficient* and the *contribution* levels of the supporting evidence.

		Attribute[1]				
		QLT	**CTR**	**IND**	**SFY**	**CNF**
Observation (a)						
Parent	Organisation	-	-	*Medium* (50)	*Medium* (50)	54
Leaf Nodes	Competent Organisation	*Medium* (50)	*Moderate* (50)	-	-	-
	Supplier Reputation	*Medium* (50)	*Moderate* (50)	-	-	-
	Pedigree for Task	*Medium* (50)	*Moderate* (50)	-	-	-
	Experience Deploying Similar Systems	*Medium* (50)	*Moderate* (50)	-	-	-
Observation (b)						
Parent	Organisation	-	-	*Medium* (50)	*Medium* (50)	**55 (+1.9%)**
Leaf Nodes	Competent Organisation	***High* (70)**	*Moderate* (50)	-	-	-
	Supplier Reputation	***High* (70)**	*Moderate* (50)	-	-	-
	Pedigree for Task	***High* (70)**	*Moderate* (50)	-	-	-
	Experience Deploying Similar Systems	***High* (70)**	*Moderate* (50)	-	-	-
Observation (c)						
Parent	Organisation	-	-	***High* (70)**	***High* (70)**	**59 (+9.3%)**
Leaf Nodes	Competent Organisation	*Medium* (50)	*Moderate* (50)	-	-	-
	Supplier Reputation	*Medium* (50)	*Moderate* (50)	-	-	-
	Pedigree for Task	*Medium* (50)	*Moderate* (50)	-	-	-
	Experience Deploying Similar Systems	*Medium* (50)	*Moderate* (50)	-	-	-
	Endorsed Processes	***Medium* (50)**	***High* (70)**	-	-	-

Note(s): 1. Quality=QLT; Contribution=CNT; Independence=IND; Sufficiency=SFY; Confidence=CNF.

Table 9.7: Observations - (b) Improving the Quality of Existing Child Nodes vs (c) Addition of Evidence to Improve Independence/Sufficiency of Parent Node

(b) It is more effective to target evidence of higher *contribution* with low *overheads* than evidence which appears to be requiring substantial improvement.

(c) The observations within Table 9.8 shows a very simplified set of attribute relationships with a single parent node with four leaf nodes. The examples result in *very* modest increases to the *confidence* value; however the example serves the purpose to illustrate the observation. Deviations from *observation (a)* within Table 9.8 in the subsequent observations, *(b)* and *(c)*, are shown as **red and bold**.

i. *Observation (a)* within Table 9.8 shows the *confidence* value (42) for a simple structure with four nodes with *low quality* and *high contribution* with one node having *low quality* and *low contribution* values.

ii. *Observation (b)* shows an increase in *confidence* (by over 2%) if the evidence with *low contribution* has the *quality* increased from *low* to *high*.

iii. However, it is shown within *Observation (c)* that *low/medium* evidence which *improves* the evidence with *high contribution* (e.g. from *low* to *low/medium*) has a greater impact on the *confidence* improvement (by over 14%). Any *change overheads* would also need be taken into account.

6. Improvements made to evidence with low/medium *quality* values need to be undertaken if there is benefit to the wider system confidence.

(a) Due to the *change overheads* associated with all evidence there is a requirement for changes to be made only if they *improve confidence*.

(b) There is little benefit to making improvements if it only allows greater perceived conformance to a standard. The objective itself and the benefit of compliance should be considered.

(c) The use of *overhead* values and the measurement of impact on node *confidence* allows for the observation that *all* additions/amendments to the body of evidence needs to play an active part in the *confidence* building.

7. High *quality* and high *contribution* values for evidence should have sibling nodes which provide context to the node's evidence.

(a) Parent nodes of child nodes with high *quality* and high *contribution* values should have commensurate *sufficiency* (if possible) to allow the benefit of the evidence to be exploited.

(b) A commensurate *sufficiency* will also allow the context of the high performing child node to be understood so that it is not in isolation.

		Attribute[1]				
		QLT	**CTR**	**IND**	**SFY**	**CNF**
Observation (a)						
Parent	Organisation	-	-	*High* (70)	*High* (70)	42
Leaf Nodes	Competent Organisation	*Low* (30)	*High* (70)	-	-	-
	Supplier Reputation	*Low* (30)	*High* (70)	-	-	-
	Pedigree for Task	*Low* (30)	*High* (70)	-	-	-
	Experience Deploying Similar Systems	*Low* (30)	*High* (70)	-	-	-
	Endorsed Processes	*Low* (30)	*Low* (30)	-	-	-
Observation (b)						
Parent	Organisation	-	-	*High* (70)	*High* (70)	**43 (+2.4%)**
Leaf Nodes	Competent Organisation	*Low* (30)	*High* (70)	-	-	-
	Supplier Reputation	*Low* (30)	*High* (70)	-	-	-
	Pedigree for Task	*Low* (30)	*High* (70)	-	-	-
	Experience Deploying Similar Systems	*Low* (30)	*High* (70)	-	-	-
	Endorsed Processes	***High* (70)**	*Low* (30)	-	-	-
Observation (c)						
Parent	Organisation	-	-	*High* (70)	*High* (70)	**48 (+14.3%)**
Leaf Nodes	Competent Organisation	***Low/ Medium* (40)**	*High* (70)	-	-	-
	Supplier Reputation	***Low/ Medium* (40)**	*High* (70)	-	-	-
	Pedigree for Task	***Low/ Medium* (40)**	*High* (70)	-	-	-
	Experience Deploying Similar Systems	***Low/ Medium* (40)**	*High* (70)	-	-	-
	Endorsed Processes	*Low* (30)	*Low* (30)	-	-	-

Note(s): 1. Quality=QLT; Contribution=CNT; Independence=IND; Sufficiency=SFY; Confidence=CNF.

Table 9.8: Observations - (b) Improving the Quality of a *Single* Existing Child Node vs (c) Improving the Quality of *Multiple* Existing Child Nodes

(c) The benefit of a higher *sufficiency* parent node is not so much a requirement for medium *confidence* child nodes as the impact on the parent *confidence* will be limited.

8. It is legitimate to *add* child node evidence to a parent where the child node does *not* increase the *average* child node *confidence* as long as the parent *sufficiency*/*independence* is increased.

   (a) A key element to *confidence* improvement is ensuring that the evidence is helping to *build* confidence.

   (b) Average child node *confidence* is valid if it is informing the *confidence* in the parent node.

9. Possible to improve *independence* but not improve *sufficiency*.

   (a) By definition an improvement in the *sufficiency* of a parent node is due to additional evidence being added which provides further information to support any *confidence* value. Therefore, *sufficiency* should increase the *independence*.

   (b) However, improving *independence* does not necessarily improve *sufficiency* as the additional evidence may be informing evidence which already exists within the framework.

   (c) Care needs to be taken when additional evidence is being chosen so that the *change overheads* are minimised and that the most benefit can be realised. Improving the diversity with additional evidence can gain more benefits than adding further evidence to what exists (even if it is *independent*).

10. It is legitimate to influence the *world-view* of stakeholder(s) to amend the perceived evidence *contribution*.

    (a) As established, evidence *contribution* is key.

    (b) If all avenues have been explored to improve evidence *confidence* via the addition of evidence, and/or *quality*, *independence*, *sufficiency* improvements then the only other attribute for modification is that of *contribution*.

    (c) It is legitimate to influence the *world-view* of stakeholder(s) to amend node *contribution* values if there is significant scientific/research evidence to warrant such action.

(d) It may be more cost effective to commission research to alter a *world-view* (legitimately) than to amend other evidence attributes or to add evidence.

(e) Great care should be taken with such an approach to ensure that there are no accusations of *gaming*.

11. Actions to consider in the event of Counter-Evidence existing.

(a) The impact on any counter-evidence is dependent on the *contribution* of the node of interest, and that of its parent node.

(b) In the first instance, efforts should be made to improve the *quality* of the node of interest. This will be based upon the ability to gain improved *quality* and the associated *overhead* of any changes to the node.

(c) Secondly, *independence* and *sufficiency* improvements should be made to the parent node of the node of interest. This is achieved via the addition of sibling nodes to the node of interest. The sibling nodes *quality* and *contribution* will have to be commensurate with the level of counter-evidence and the average *confidence* of the pre-existing sibling nodes.

(d) A third approach is the addition or improvement of evidence within the *evidence family* so that the counter-evidence can be mitigated. This can be within the same subtree or preceding subtrees.

(e) It should be noted that an increase in the *quality* of the siblings of the node of interest *can* improve the parent node confidence but there would need to be substantial *quality* improvements which may not be feasible due to associated *change overheads*.

12. Any amendments to node attributes or the addition of actual nodes, i.e. evidence, should model reality and therefore the *perceived* targets for the system/software may not be reached.

(a) There are a range of actions that can take advantage of diverse evidence to *spread* confidence. However, any modelling of the evidence and the attributes should be based on the known and perceived evidence.

(b) It is very possible that changes to the evidence may not result in the target for the system/software being reached as the evidence may not support the required DAL.

(c) It is important for this to be recognised so that continual *adjustments* of the model are not made to arrive at the desired result. There is a need to avoid *gaming*.

(d) Knowing that a system and/or its supporting software cannot achieve a target DAL is informative in itself. Action can be taken with this knowledge to conduct risk acceptance or to apply limited clearances/approval of the system, for example.

9.7 Evaluation of the DSF

9.7.1 DSF: Assessment of the Implementation

The observations which have been made on the DSF case study outputs and the exploratory testing have been extremely valuable in gaining an understanding of how evidence can be gathered and assessed.

There have been some interesting and surprising results; e.g. it was envisaged that the addition of high *quality* evidence would provide *significant* improvements to the overall evidence confidence. However, the main factor was the *contribution* of the evidence which played a greater role in gaining evidence confidence. This observation can influence decisions in the scenarios where efforts could be made to improve existing evidence attributes (e.g. improve the *quality*) *or* to gather further suitable evidence (e.g. that which *contributes* significantly). This is also illustrated with the observation that there is greater benefit to making small incremental changes to *highly* contributing evidence rather than large changes to *lesser* contributing evidence. These observations are important as they contradict the traditional methods to gain evidence for process *compliance*. There has been a perceived necessity within guidelines to gather evidence for *all* objectives as they are all traditionally treated *equally*. The DSF assists with illustrating that they are not all equal.

The concept of *backing* and *reinforcing* evidence should also be a consideration[12]. With low levels of evidence confidence the addition of suitable evidence initially provides more significant gains in the overall confidence values. This rate of improvement slows as the level of confidence increases with there being a requirement on the stakeholders to make more discerning choices for the evidence amendments. However, the rate of growth and subsequent improvements was of interest with *high* confidence evidence making small improvements; e.g. ECT evidence confidence of *66* improving the Overall Confidence (OC) from *74* to *75* which can be explained by the agreed contribution of the ECT.

Another notable observation is that additional child node evidence does not necessarily need to increase the average confidence of the existing child nodes; i.e. the additional evidence quality does not have to be as high as the existing evidence. However, the child evidence does need to improve the sufficiency/independence of the parent. As with other observations, this contradicts the premise that *all* objectives are to be met with little regard

[12]See sub-section 8.2.5.7.

to the wider influence on the evidence siblings or parents as not all evidence is equal. In reality, the evidence needs to be evaluated on what it provides within an evidence *family*, e.g. the increase to the *sufficiency/independence* of the evidence and the *contribution* it makes.

The DSF is centred upon SME judgements which are based upon education and experience. As evidence *contribution* is now understood more fully, providing evidence to influence SMEs can be cost effective. Therefore, commissioning of research to influence a stakeholder's world-view becomes a legitimate and useful approach.

The evidence selected for the case studies and exploratory testing was not exhaustive. Nor was the permissible evidence within Chapter 7. However, the case studies and exploratory testing did illustrate that the differing types of evidence can be judged consistently. This included process-based evidence, e.g. SDPs, and in-service considerations, e.g. quality of error reporting process, etc. This spanned a range of evidential types which provided quantitative and qualitative results. The DSF allowed them to be judged in a consistent manner. Additional appropriate evidence can be accepted by the DSF, e.g. simulation, to include further evidence. The DSF can expand on the evidence which was included within the case studies outlined in this chapter.

The chosen attributes, e.g. *confidence* etc., had the aim of allowing the *influence* of evidence to be captured. Importantly, the attributes allowed an assessment on if the evidence confidence needed to be reduced due to potential *missing* evidence. The DSF also provided the facility to potentially remedy any reduced confidence via the addition or amendment of suitable evidence attributes.

The attributes and their formats were devised to allow DSF decision makers to generate a large quantity of assessments on the evidence in an intuitive manner. The attributes balanced the practicalities of extensive evidence assessments and the level of detail required to derive proposed optimised solutions. SME feedback was received regarding the granularity of the attributes chosen. The potential to further refine attributes such as *independence* to explicitly state convergence or mutuality factors was raised. As was the potential for the attributes to consider the form of evidence, e.g. *backing* evidence, which would allow metrics and decision support to strengthen the evidence which is gathered. These are valid observations which may possibly tip the balance of the DSF to being more complicated to implement, and therefore to derive lessons from. However, implementing such observations may increase the perceived *rigour* of the approach.

Evidence states can be either *obligatory*[13] or *ancillary*[14]. *Extant* evidence is either *obligatory* or *ancillary*. Separating the *contribution* of any evidence and its state proved useful to

[13]Recognised by regulators as part of guidelines/standards.

[14]Supporting evidence which can inform *confidence* but is not part of existing guidelines/standards.

ensure that the SME judgement on evidence *contribution* was not influenced by the defined guidelines/standards. It was important for the evidence to be judged on its merits and not by the content of particular standards, which SQEP SMEs may disagree with. However, the ability to weight evidence to favour *obligatory* data had benefits. It allowed the DSF to optimise evidence which favoured an overall solution which had the *intent* and *persuasion* to meet regulatory requirements. This allows the DSF to not reflect a 'pick and mix' approach to evidence but one which is considered and optimised appropriately. The assignment of suitable *contributions* in the DSF helps to alleviate risks of evidence 'cherry-picking'. However, it should be noted, that a diverse spread of evidence is not a negative approach as long as it is assessed consistently.

The ability to apply penalties to the GA answers[15], and hence influence the optimised and appropriate evidential solutions, proved to be of greater value than expected. The application of *change overheads* to the evidence acted as a differentiator for the potential evidence choices. The *overhead*concept allowed *practical* evidence shortfall mitigations to be devised, e.g. risks due to timeliness of information. The decision makers can also consider the perceived value of the evidence from a *theoretical* position. The *change overhead* influenced the choices made within the case studies and the exploratory testing. At present the *change overheads* do not have the facility to apply priorities to the risks associated to time, cost, and quality. If the true benefit of the *change overhead* was envisaged during the design of the DSF then a more comprehensive facility may have been implemented. The facility would have been subject to greater analysis within the case studies and exploratory testing.

Key feedback from SME input was for the DSF to allow visual indicators to assist with the decision making. The SME feedback recognised that evidence sources would become relatively complicated quite quickly with the premise being that *all* evidence is of relevance. The selection process for the visualisation approaches within the DSF was considered and based upon SME input. The level of visual indicators and assistance proved valuable to the decision makers; however, there are subtle changes to the visualisations which could provide greater comprehension and drive better decision making. As an example, the ability to provide visualisations such as Bubble Charts to assist with stakeholder communication. There is however, a balance to be gained with the visualisations. There is a risk that data represented in certain formats provides a false impression of precision and therefore the data drives the decision making to a greater extent. Visualisations to convey a message about the data is valid as this can *assist* decision making. Data presented via visualisations which *controls* decision making is to be avoided. This is a key tenet to the DSF which provides visualisation approaches of value.

[15]Implemented via penalty functions.

The use of the DSF to implement the case studies and exploratory testing allowed the end-to-end process and the outputs to be validated. The DSF is, as stated, to act as a tool to assist with gathering and judging diverse forms of evidence. The factors which are considered for these activities are subjective and based upon SME considerations. However, the results and outputs from the tool are reasonable based upon the analysis and SMEs judgements which have been fed into the DSF. In addition, based upon SME feedback the DSF evidence relationship outputs and the overall outputs are sufficiently accurate and valid to provide decision-support information. The aphorism "all models are wrong, but some are useful" (Box, 1979) is useful in the context of the DSF as the framework is to provide decision-support to judge evidence attributes to inform a level of confidence. The framework advances the current approach as it structures subjective assessments; however, there will always be expert judgement required to interpret the DSF outputs.

The approach defined in this thesis, with the use of such approaches as the DSF, can assist stakeholders in achieving a *defensible* position. This is accomplished by the DSF capturing explicit expert consensus-agreed values for the intrinsic attributes of the evidence[16]. This is for the underpinning leaf node evidence and also the SMEs judgements on the branches of the overall evidence structure. This allows SMEs to show their *belief* in certain evidence types. The subjectivity of SMEs is captured and exposed within the DSF allowing SME judgements to be open for scrutiny. In essence, SMEs are 'showing their workings' which allows others to understand *how* the confidence levels were reached and acts as a mechanism for the subjective judgements to be defended.

9.7.2 DSF: Comparison to Related Work

The limitations of the current methods discussed in Chapter 3 focussed the design of the DSF and the wider research activities with the identified shortfalls addressed by the DSF. The DSF uses the evidence attributes in a very specific and non-trivial way with the attributes informing the method to combine the evidence. This sub-section compares the treatment of the attributes and evidence by the DSF to the existing research in this area.

A number of the case studies associated with the existing confidence quantification methods are based upon simplified and idealised examples, e.g. Littlewood and Wright (2007), with a deliberate simplifications of real situations. In addition, some of the concepts were subject to incomplete treatment (Delic, Mazzanti and Strigini, 1995) or a simplification of the captured attributes (Bouissou, Martin and Ourghanlian, 1999). Bloomfield and Littlewood (2006) use special examples of diverse argument legs for their review with Yamamoto

[16]The attributes are: *confidence, quality, contribution, sufficiency*, and *independence*. See sub-section 8.2.2.

(2015) using very small sample sets. The case studies implemented in support of the DSF were based upon scenarios adopting evidence sets which were relevant to given systems. The implemented case studies were not idealised and based upon real scenarios. Therefore, the observations from the DSF were based on learning from *reality*, i.e. via case studies, and by envisaged scenarios, i.e. via exploratory testing. The exploratory testing allowed further investigations of the node/attribute relationships within the DSF. The examination of the attribute relationships was a key reason for the development of the DSF. Understanding the relationships has allowed observations to be made to *guide* future diverse evidence gathering approaches.

Some existing methods for the quantification of qualification evidence propose that the more formal notation to capture confidence, based upon mathematical principles, results in a degree of SME judgement being removed. The removal of SME judgement is based upon allowing quantified results to act as the decision making *driver*. This thesis suggests that quantitative (and even qualitative) representations of confidence should act to *support* SMEs judgements. The support can take the form of a structured approach, as with the DSF, but there is a reliance on expert judgement. A full formalised approach for developing judgements on confidence is not feasible given the current shortfalls in existing methods. The activities within this thesis have provided guidelines and tools to assist with the forming of judgements. This philosophical underpinning, in relation to the role/level of expert judgement, was explored. The thesis supports the need to *embrace* SME judgements and to facilitate its capture rather than to replace it.

It can be argued that any data generated as part of a tool should assist with the SMEs comprehension (Kirk, 2016). A number of the existing methods provide structures to form arguments, e.g. use of GSN by Ayoub et al. (2013), Duan et al. (2015), and Zeng, Lu and Zhong (2013). However, with these studies there is no correlation between (a) the values generated due to the quantification of the assurance confidence and (b) the visual representations provided to the SMEs. This results in there being no link between the data and the representation. The DSF provides a clear link between the visual representation, e.g. via graded tree structures, and the underpinning data. This is important to allow expert judgement comprehension and to assist the expert with forming further analysis decisions, e.g. via *what-if* scenarios.

Existing methods were identified which do not specify how the results should be used to determine if a system is sufficiently safe, nor how attributes should be measured, e.g. Cyra and Gorski (2008), Duan et al. (2015), Guiochet, Hoang and Kaâniche (2015), Yamamoto (2015), and Denney, Pai and Habli (2011). The DSF provides an output which can allow judgements to be based upon it directly, i.e. a DAL, or for the output from the DSF to feed into a wider safety argument. The output is to assist decision making and is not to remove

the expert judgement which is associated with assessing the validity of a DAL.

A number of the existing approaches assign confidence values to outcomes but the methods to capture the supporting information requires interpretation with differing levels of abstraction to assign suitable values. The proposed methods in some existing approaches (e.g. Guo (2003), Nair, Walkinshaw and Kelly (2014), and Duan et al. (2015)) would require SMEs to restructure how evidence is traditionally captured. The DSF adopts an approach which is purposefully closely related to current practice. The aim is to put forward an approach, in the first instance, which is usable to allow a gradual change in process and stakeholder mindset.

Some of the existing methods have complicated structures in terms of the attributes which are captured, the level of attribute association, and the argument formation, e.g. Wang, Guiochet and Motet (2017). In practice there would be difficulties in repeating these structures for a larger set of evidence. The DSF contains *patterns* to allow the attribute relationships to be repeated within the structure based upon a parent-child(ren) relationship. This allows evidence strands to be added/amended intuitively and it is agnostic to the form of evidence to allow a greater diversity of sources. The method in which confidence values are propagated within the DSF is also scalable. The patterns introduced in the DSF are repeatable and provide consistency.

A key element to a decision making process is to assess a number of alternative solutions (Turban, Sharda and Delen, 2010). Allowing judgements to be captured is only one element of a wider decision making process. Alternative forms of evidence should be devised and assessed. This is a key concept for any approach to measure confidence and capture diverse evidence. The approaches which currently attempt to allow confidence to be quantitatively measured using diverse evidence do not provide such rich tools. The DSF provides a method to *compare* and *measure differences* in alternatives. The DSF also allows decisions to be based upon *efficient changes* via optimisation. The approach allows an end-to-end decision making process to be adopted, as outlined in Figure 8.21.

The decisions made, or any optimisation being performed, should be based upon known or perceived risks. To conduct any evidence gathering activities there are a number of factors to consider which are *not* solely based upon the availability of data. The use of additional diverse evidence would require considerations such as the time to generate/perform the evidence and the financial costs, for example. The DSF associates *overhead* values with evidence so that the *cost of change* can be factored into any decisions. *Cost* in this context refers to time, quality, and financial factors. Again, such a rich consideration is not part of any existing methods.

There are numerous approaches to reason under uncertainty via the quantification of assurance confidence, e.g. ER by Nair, Walkinshaw and Kelly (2014), BBN by Hobbs and

Lloyd (2012), DST by Ayoub et al. (2013), etc. There are no overriding arguments to fully support one approach. The DSF has implemented a number of FISs due to the intuitive method to create membership functions and the ability to generate a structure which will scale and allow the propagation of values. The DSF does not propose fuzzy logic or the use of FISs as a *preferential* approach but these are used to illustrate that diverse evidence can be captured and reasoned upon. Fuzzy logic and FISs within the DSF has allowed valid observations to be gained from the attribute relationships and to capture lessons regarding the visualisation of data and the *change overheads*, for example[17].

Many of the approaches for the quantification of evidence are focussed on providing an overall numerical value for the *confidence* in a safety claim being met. Wang, Guiochet and Motet (2017) also use DST to generate the *trustworthiness* of a claim via *belief*, *uncertainty*, and *disbelief*. Contributing *weights* of the supporting nodes is the degree that the nodes *independently* contribute (the *appropriateness* of the goals). Wang, Guiochet and Motet (2017) propose two argument types: *dependent* and *redundant*, to propagate the *trustworthiness*. However, the arguments for the propagation have to be manually specified for each instance. Also, the arguments types are binary in terms of their selection as the type is either *dependent* or *redundant*. Therefore, so are the consequences of the value propagation. The DSF uses the evidence attribute values to signify the node relationships rather than the need to determine them explicitly for each parent-children instance. Thus, providing a more practical implementation.

Ayoub et al. (2013) looks at the degree of belief on the *sufficiency* and *insufficiency* of the evidence to support the conclusion. However, the adoption of DST by Ayoub et al. (2013) means that evidence nodes are assumed to be *independent*, an acknowledged weakness of the approach. Ayoub et al. (2012) is not based on the *quantification* of confidence as it has a focus on common areas of concern to consider for hazards, e.g. the tool used to provide an output. The Ayoub et al. (2012) process is to arrive at at a level of *trustworthiness* in the evidence. There is no consideration of the evidence relationships and priorities. The Ayoub et al. (2012) research is more an extension to justifying confidence in a top-level safety claim within a GSN approach. The DSF provides a more rigorous assessment of the fundamental node relationships compared to Ayoub et al. (2012), for example.

Nair et al. (2015) base the main criteria for assessing confidence upon *trustworthiness* (i.e. capturing an assurance that the evidence is as specified) and *appropriateness* (i.e. capturing the satisfaction of the claim). The Nair et al. (2015) ER approach involves no comparison to the siblings of the evidence, i.e. no measure of *sufficiency* or *independence* - unlike the DSF. Nair et al. (2015) claim to provide a "systematic guided process that considers an *exhaustive*

[17]See sub-section 9.7 for further information.

list of confidence factors" which are linked to *hazards* rather than *evidence confidence*. What Nair et al. (2015) claims is an *exhaustive* approach for *each* hazard could be unsustainable if attempted to be implemented in practice. This is due to the volume of the analysis required for the Nair et al. (2015) approach.

Hobbs and Lloyd (2012) introduce a BBN approach which adopts the concept of a *leaky* noisy-OR to account for the level of confidence that the provided evidence represents *all* of the evidence required to support the conclusion. This leads to a conclusion being *true* even if *all* evidence is *false*. The use of a *leaky* noisy-OR by Hobbs and Lloyd (2012) takes an optimistic measurement of evidence which, in terms of safety, should take a more cautious approach. Any *positive* confidence in evidence *must* be based upon a determined set of evidence with defined judgements. The *leakage* concept adopted by Hobbs and Lloyd (2012) acts as an unsubstantiated method to gain evidence confidence with no defined limits to the *leakage* value stated in the paper. Any *positive* claim to confidence within the DSF is based upon the supporting evidence attributes. In addition, the BBN approach results in one or more forms of evidence being based upon the *law of additivity*. The DSF avoids this limitation via the use of fuzzy logic and FISs[18].

Cyra and Gorski (2008) have a method to judge the *trust* in a claim based upon the stated *arguments* which are supported by *warrants*, *assumptions*, and *facts*. The evidence to support these attributes are judgements on the *quality* and *validity* of the arguments. However, there is no concept of judging the evidence in relation to the level of mutuality it represents to other evidence (the DSF does consider such an attribute). In addition, the rules of aggregation need to be *user-defined* for *each* claim. This would be a very intensive process in practice and, as stated previously in this section, this is simplified by the DSF implementation via consideration of such properties during the *construction* of the evidence tree. Cyra and Gorski (2008) also make no direct link between the visual representation and the procedures to capture and calculate the confidence values - however, this is provided by the DSF. Cyra and Gorski (2008) do not specify how to use the results to determine if a system is sufficiently safe, nor do they state which attributes should be measured. They do not provide a clear indication of *how* the outputs should be used by stakeholders to inform decisions. The DSF differs in that a degree of membership to one or more DALs is provided. Also, the thesis research has provided a flow of the activities needed to gather/generate evidence, e.g. via the flow illustrated in Figure 9.3 and the case study and exploratory testing observations outlined in sub-section 9.6.

Denney, Pai and Habli (2011) aim to calculate uncertainty in safety claims via the use of BBNs. The confidence in a given leaf node is represented by assigning values to multiple

[18]See sub-section 8.2.4.4 for further information.

scales (*very low* to *very high*) with the values for all scales combined being 100%. However, the values are subjectively provided for a single scale. There is no ability to assess multiple confidence factors. This is also the case for Duan et al. (2015) which provide a concept to capture expert judgements via subjective logic but does not detail any method to *combine* these confidence values. A fundamental element to the DSF is the combination of evidence with lessons/observations captured from the research activities[19].

Yamamoto (2015) provides indicators to GSN nodes but the approach is to further refine the GSN notation. Unlike the DSF, the Yamamoto (2015) approach does not review the relationship between nodes and does not provide a method to measure the evidence, e.g. there is no concept of *quality*. Zhao et al. (2012) provides an approach which is based upon BBN and Toulmin arguments, however the *quantitative* results from the Zhao et al. (2012) case studies indicates that *quality* and *contribution* factors are not sufficiently considered[20]. Whereas, the DSF uses these attributes as a fundamental method to derive the confidence values.

Littlewood and Wright (2007) has a very specific BBN model which has simplified variables to capture observations from multi-legged arguments. The Littlewood and Wright (2007) approach has a focus on gaining greater insight to the multiple legs of evidence rather than how BBN is normally adopted for confidence assessment. Due to this the approach does not have the richness to explicitly consider the *sufficiency* or *independence* of the variables, unlike the DSF.

Other research contains attributes of relevance to the DSF, e.g. considering *independence* (Yuan and Kelly, 2011); however, many are based upon *qualitative* judgements, e.g. Hawkins and Kelly (2009). There is an inability to combine such evidence which is captured using a *qualitative* approach. Such research lack the richness that the DSF provides in adopting numerical confidence values which can consider factors such as *change overheads*.

The DSF offers a unique perspective as it provides a mechanism to capture a range of confidence factors via attributes which span a number of evidence relationships, e.g. to the *parent* and *siblings* (both extant and potential[21]). Patterns are devised for the DSF to *build* the structure which is repeatable for complicated evidence sets which will be encountered for the wider system solution spaces[22]. The captured judgements can be suitably reasoned upon with the DSF providing the facility to determine feasible steps based upon a perceived stakeholder *reality*, e.g. via *change overheads*, and the ability to arrive at an *optimised* and satisfactory set of evidence.

[19] See sub-section 9.6 for further information.
[20] See sub-section 4.2 for further information.
[21] See sub-section 8.2.3 for further information.
[22] As illustrated simplistically within Figure 1.1.

9.8 Summary: Case Studies and Exploratory Testing

A number of case studies and exploratory tests were devised which aimed to exploit the features of the DSF to understand the relationship of the evidence *attributes*.

The case studies and exploratory tests were devised in order to respond partially (or in full) to a number of findings within the *Background and the Problem of Interest* (chapter 3) which highlighted why further work was required in this area. In addition, the exercises assisted in exploring the benefits of the DSF features (e.g. visualisation and optimisation) to assist in achieving suitable diverse evidence 'solutions'. Linked to this is the demonstration of the value of optimisation itself and the reasoning under uncertainty approach being fit for purpose.

A range of DALs were selected for the systems under review as part of the case studies and the exploratory testing. This highlighted the levels of *confidence* that are required for the various system criticality values and also the expectations for the evidence for these systems.

The exercises centred on the notion of: altering *independent* and *dependent* variables (nodes and attributes); and understanding the relationships between the highlighted node attributes values (e.g. siblings *quality*). A potential initial flow for the assessment was devised to assist in the early exploration of approaches to any given diverse evidence problem. The case studies and exploratory testing purposefully deviated from the initial flow to understand the attribute relationships in greater detail.

The results of the exercises were extremely positive with lessons identified from each of the case studies and the exploratory tests. The exercises indicated that there is value in the incremental adoption of diverse evidence; however it was shown that there are limits to the confidence 'which can be gained if the evidence is not supportive.

A range of observations have been made with regard to the relationships between the attributes of the evidence. These results have been interesting and surprising to a degree in that attributes can have a greater impact on confidence than originally devised, e.g. *contribution*. An understanding has been gained on how any attributes changes are propagated to impact the *confidence* of other nodes (and therefore the overall DAL claimed for an evidential set[23]). Also, there have been observations which can influence scenarios where evidence shortfalls need to be mitigated. In essence, the observations can act to *guide* stakeholders in potential next steps to mitigate or accept evidential shortfalls, i.e. the lessons and the DSF can *target* interventions.

[23]The representation of confidence in this way also assists with outlining the *value* of the evidence and assists with identifying a stopping point for evidence gathering/generation.

Chapter 9 has informed two research sub-questions:

- Sub-sections 9.4 and 9.5 have partly responded to the sub-question: *What is a suitable structure for software safety assurance evidence and can mathematically derived approaches inform how judgements are made on the evidence and for proposing alternative/optimised solutions?*.
- Sub-section 9.6 has partly responded to the sub-question: *What observations and recommendations can be made on how to implement a software safety assurance evidence argument and how to inform a UK defence software safety assurance strategy?*.

Chapter 10

Recommendations to Enhance Current Software Safety Assurance Processes

Due to the range and breadth of research activities conducted a number of enhancements have been identified to current MOD practice. These enhancements may allow the MOD to move towards an environment which allows diverse evidence to be captured and judged. This can be conducted in a more efficient and effective manner for systems which reside within the *expanded* solution space[1].

The purpose of the EngD is to produce "industry value-adding research" with an "application in an industrial or commercial context" (IDC in Systems, 2013). The enhancements described in this chapter help to meet the requirements stated within IDC in Systems (2013).

This chapter will examine:

- *Methods to Enhance MOD Software Assurance.* Details of a number of enhancements that could be made to the currently defined permissible software-related evidence and the subsequent safety assessment process for MOD.
- *Suggested Approach to Adopt Diverse Evidence within a Software Assurance Qualification Strategy.* Summary of the guidance written[2] for DE&S DT Desk Officers which can assist DTs with their procurement approaches to gain diverse evidence.
- *Challenges to the Adoption of the Methods.* For the enhancements to be implemented there will need to be changes to current assurance practice. With any change there

[1] See Figure 1.1.

[2] Written by the RE and the EngD Industrial Supervisor (Dr Mark Hadley - Dstl Senior Principal Scientist in Software Systems).

will be barriers encountered.

10.1 Methods to Enhance MOD Software Assurance

The activities within this thesis have allowed the RE to construct a set of robust and defensible enhancements to the methods in which the MOD conducts software safety assurance. The enhancements are based upon findings from *all* elements of the thesis from Chapter 3 (*Background and the Problem of Interest*) through to Chapter 9 (*Case Studies, Exploratory Testing, and Evaluation of the DSF*). The enhancements have a number of themes, as shown in Figure 10.1.

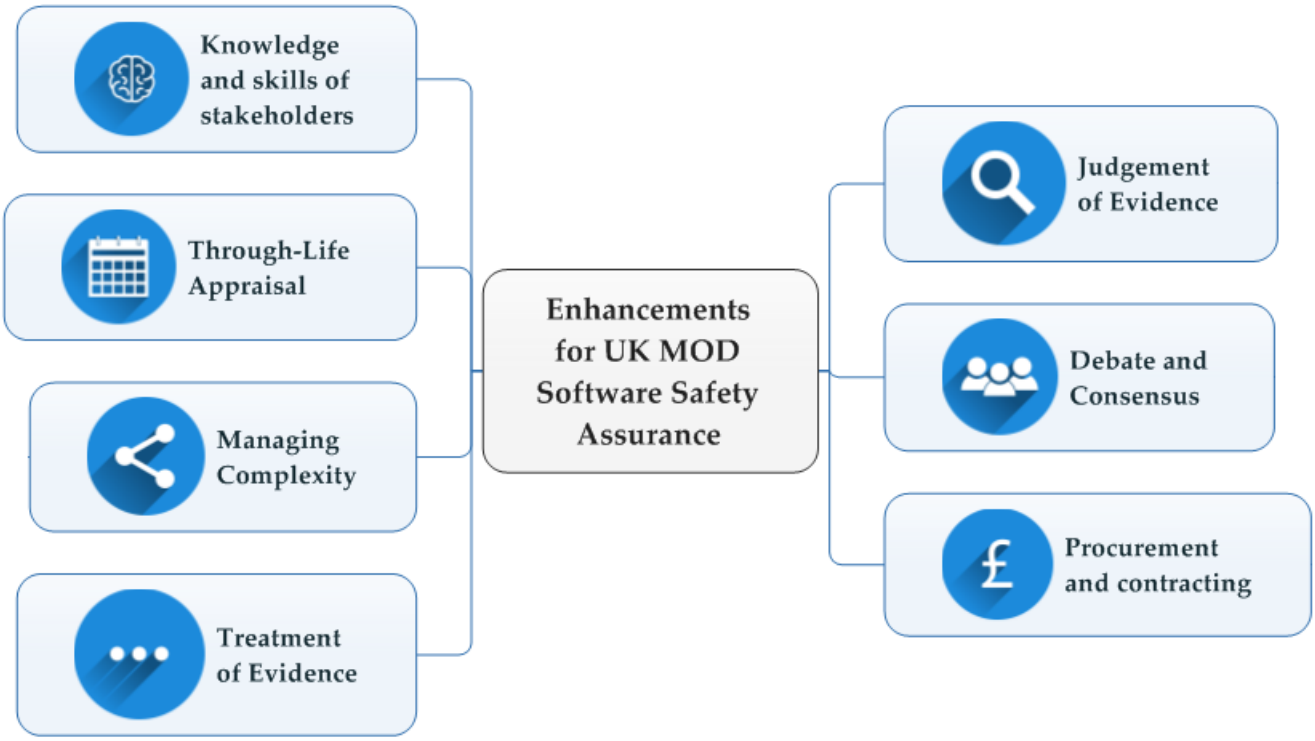


Figure 10.1: Themes to the Enhancements to MOD Software Safety Assurance

1. Knowledge and Skills of Stakeholders.

   (a) *SQEP requirements and skill sets to alter to adopt diverse evidence*[3]. The ability to review and judge a wider range of diverse evidence requires a revised skill set

[3]Within an safety assurance domain which is not focussed on process-based evidence the potential supporting evidence will be wider and more complicated. This reflects the fact that the solution space is comprised of a wider set of systems and procurement types. In an assurance environment which embraces

compared to a pure process-based approach. These include, but are not limited to: the procurement environment, system and platform interaction (including SoS), and effective industry/customer engagement[4].

(b) *Principles of evidence theory need to be restated.* The enhanced approach requires evidence shortfalls to be mitigated and holistically assessed and 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[5], and also the *theory of evidence*. The ability to apply, for example, *backing* or *reinforcement* evidence for software needs to be commonly adopted.

2. Through-Life Appraisal.

(a) *Judgements should be captured for the life of the software/system.* There are distinct phases of any LRU such as the development and in-service periods. Depending on the phase at which the initial judgements were captured there will, most likely, be opportunities to provide regular updates on judgements of the software. Standards and guidelines evolve, albeit slowly, but the attributes associated with any evidence can remain constant. Therefore, a framework such as the DSF can act to capture judgements through-life as 'snap-shots' to reflect the evidence characteristics. This is particularly relevant as a LRU and its software moves towards being classed as *legacy*[6]. The DSF can be a consistent method used throughout the distinct phases of a system.

(b) *Capture the confidence being built rather than only ongoing problems.* In-service evidence should be a method to *build confidence* rather than to 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 maintain, or *improve*, confidence[7].

(c) *Actively gather metrics to inform diverse evidence.* A premise for diverse evidence is that *all* evidence is of relevance but the contributions vary. In addition to

diverse evidence there is a need to gain and comprehend evidence which is pre-existing and in varying formats.

[4]Understanding system interactions and behaviours can also be assisted via adopting such techniques as the SLF as outlined in sub-section 3.3.2 and the SLF conference paper research outputs

[5]Such as understanding the observations of attributes/relationships outlined within sub-section 9.6.

[6]As outlined in the tenets within sub-section 7.2, for example.

[7]As illustrated within sub-section 7.1.11.

capturing evidence to validate any prior-belief in the process evidence, e.g. in-service data, 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 are a myriad of metrics which can be obtained to support or refute the confidence in a system, e.g. technical suggestions raised in TIMs which lead to actual changes. Opportunities to gather such data should be encouraged. It should be noted that any *quantification* of confidence and the metrics gathered are to *inform* SME judgement and not to replace it[8].

(d) *Ensure the right metrics are informing the right judgements.* There is a need to be mindful of the use of metrics and how they shape the safety arguments and how they alter the confidence which is being gained. Measuring the 'wrong' thing can have unintended consequences, for example via future decisions taken or by the introduction of inherent weaknesses in the confidence argument[9]. There are other dangers with metrics such as *gaming*[10] a quantification-based approach and the difficulties with SMEs agreeing metric values.

3. Managing Evidence Complexity[11].

(a) *Gain proportionate information via system interfaces.* There is merit to adopting an approach which allows the behaviour of systems and the underpinning software to be understood via interfaces. This approach allows behaviour to be understood at a system level with deeper analysis being instigated at the sub-system and software level. This understanding can be gained via models. The use of such frameworks as the SLF can assist with gaining proportionate information and assist with stakeholder dialogue[12].

(b) *Management and stakeholder comprehension of an increased range and depth of underpinning evidence.* Process-based compliance which is benchmarked against standards/guidelines results 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 the SMEs. Due to this there is a

[8] As outlined in the tenets within sub-section 7.2, for example.

[9] As outlined within sub-section 7.3.

[10] See sub-section 7.3.1 for further information on the concept of *gaming*.

[11] The *complexity* in this instance is that created due to the increased level of relevant admissible evidence to inform a software assurance argument. The issue of *software complexity* is a separate topic but there is overlap as diverse evidence can mitigate the lack of non-traditional evidence sources.

[12] As outlined via the SLF in sub-section 3.3.2 and the SLF conference paper research outputs.

requirement to have the *judgements managed/captured*[13]. A framework such as the DSF can assist with this[14].

(c) *Growth in the potential 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 and will come under *potential* review. Stakeholders can select a myriad of evidence strands and therefore the potential solution space grows significantly. This larger solution space would be complicated for stakeholders to assess and determine actions. There is a need to establish a method/framework to allow next steps to be considered via *what-if* analysis and to optimise potential solutions. A framework such as the DSF can assist with this[15].

(d) *Communication of evidence to stakeholders is key.* Allowing direct 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. There is also a requirement to allow wider stakeholders to comprehend the decisions evidence status at a more *abstract* level. Not all stakeholders with 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* devised as part of this thesis could assist with this[16]. In essence, the *complexity* of the qualification approach is simplified into an elegant representation[17].

4. Treatment of Evidence.

(a) *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 which 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. A preferred option which embraces the use of diverse evidence

[13]The judgements will state the *acceptability* of the evidence, for example.
[14]As outlined within sub-sections 8.2.5 and 8.2.6, for example.
[15]As outlined within sub-sections 8.2.8 and 8.2.9, for example.
[16]As outlined within sub-section 7.4 and Standish and Hadley (2018).
[17]This is akin to the *brontosaurus of complexity* (Holt, 2007) where rather than producing an elegant *solution* to a complex problem the communication allows an elegant *representation* of a complex problem.

is to allow weightings, e.g. *contribution*, to be formally established for objectives and to allow these to be managed and reasoned upon[18].

(b) *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[19]. The range and depth of evidence can be significantly increased. Collating evidence will allow it to be judged consistently and to establish the level of influence which one set of evidence has over another.

(c) *What does 'good' look like?* With the adoption of diverse evidence there are a number of standards/guidelines which can inform a view of what an ideal scenario would look like for any given evidence strand. 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 allow a measure to be gained of the shortfalls and therefore, the confidence which can be assigned to such evidence. A framework such as the DSF can allow these types of standards/guidelines to be included and judged[20]. However, diverse and *radical* evidence which does not have a precedence will lack a supporting framework. Such evidence will be significantly reliant on SME judgement.

(d) *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. The current guidelines/standards lack benchmarks to build a consistently judged diverse evidence assurance argument. Therefore, 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[21].

(e) *Diverse evidence associated with a LRU 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 process-based approach which applies *labels* to the

[18] Using such attributes as those defined within sub-section 8.2.2.

[19] As described within sub-section 6.1.1.

[20] As stated within Chapter 8, the research output seminar presentation titled *The Cake of Alternative Software Safety Evidence: Getting the Ingredients Right and Modifying the Recipe*, and illustrated within the DSF case studies in Chapter 9.

[21] As outlined within sub-sections 8.2.5, 8.2.6, and the safety-critical systems club seminar research output titled *Use of Service History and Field Data - In Support of Safety Justifications.*

software, e.g. "DO-178B DAL B compliant", which doesn't account for the wider supporting evidence. Therefore, there is a requirement to treat LRUs as bespoke items with judgements on the evidence being captured[22].

(f) *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 which is assigned to systems will be derived from non-process evidence. Therefore, the systems and software may not receive a label which succinctly states a process-based standard compliance. This is due to there being a wider and diverse set of evidence providing the *confidence*. Therefore, there should be consideration to remove the labels which 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[23].

5. Judgement of Evidence.

(a) *Judgements must be captured with consistent attributes.* An assurance approach which uses wider evidence must have the ability to allow *priorities and the purpose of evidence to be managed/captured.* Attributes of evidence are *essential* to allow the detail of the judgements to be captured and to maintain a consistent review of 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. The DSF has proposed a number of potential attributes to assist with evidence judgement[24].

(b) *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. The concept of being reliant on SME judgement is common within other domains and there is acceptance that there will be a variance of opinion and judgement. The MOD assurance stance in relation to software should also take an approach to provide credence to SME judgement. This would allow judgement to be made from a *defensible* position rather than one which is based upon known

[22] As outlined within sub-section 8.2.5.
[23] As supported by sub-sections 7.1 and 8.2.5.
[24] As outlined within sub-sections 8.2.2 and 8.2.7, for example.

and accepted benchmarks, i.e. that which is *repeatable*[25]. A *defensible* claim by a stakeholder would be one which is justified with an opinion which can be argued to be *good*[26] (Collins Dictionary, 1995*f*). This would be a change to the current safety assurance paradigm and may have legal ramifications in relation to individual liabilities. There would need to be a shift in the world-views of the stakeholders for any increased risk that would need to be accepted.

6. Debate and Consensus.

   (a) *Collective judgements need to be captured.* An approach underpinned by judgement rather than process conformance requires a number of stakeholders to alter the approaches adopted for evidence measurement. A number of subjective stakeholder opinions and their statements of evidence acceptance needs to be captured. This information should be able to be debated and reasoned upon via a suitable structure and method (such as the DSF)[27].

   (b) *Move to a more consensus based approach for evidence judgement and acceptance.* There would be greater emphasis on stakeholder judgement if a DSF is adopted. 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 would be a *joint acceptance* by the stakeholders of the diverse evidence. This could then inform any decisions to taken by Duty Holders, for example. The DSF can act as as method to allow reasoning and contested matters to be stated and to assist with the generation of formal outputs[28].

   (c) *Move to a more consensus based approach for evidence judgement and acceptance.* There would be greater emphasis on stakeholder judgement if a DSF is adopted. 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 would be a *joint acceptance* by the stakeholders of the diverse evidence. This could then inform any decisions to taken by Duty Holders, for example. The DSF can act as as method to allow reasoning and contested matters to be stated and to assist with the generation of formal outputs[29].

[25] As supported by sub-section 6.2.

[26] Based upon the decisions made to structure and judge the supporting evidence. See sub-section 9.7.1 for a perspective on *why* the DSF can provide a *defensible* claim.

[27] As that outlined within sub-section 8.2.

[28] Akin to joint reports/submissions regarding differing expert witness perspectives within the criminal justice domain. Also, as supported by sub-sections 6.2 and 8.2.

[29] Akin to joint reports/submissions regarding differing expert witness perspectives within the criminal justice domain. Also, as supported by sub-sections 6.2 and 8.2.

7. Procurement and Contracting.

   (a) *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 which 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 to be received[30].

   (b) *Method to contract for software assurance evidence will need to reflect diverse evidence approach.* If LRUs 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 SDP. However, with the approach devised by the RE 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[31].

10.1.1 Timeframes for Implementing the Methods

Due to the nature of the safety assurance domain any changes to an established method requires a robust and patient approach to fully adopt the benefits which diverse evidence can provide. Changes to MOD strategy will require a prolonged period to gain traction and acceptance; therefore, there is a need to have a staged approach to influencing MOD policy.

In the near term the MOD policy regarding the use of diverse evidence can be influenced, and has been as a result of this thesis, to enable a wider range of evidence to form part of a robust assurance approach. This extends to the methods to communicate evidence shortfalls and plans with stakeholders, e.g. via the *Wheel of Qualification*. Another near-term aim is to influence the *knowledge and skills of stakeholders* who reason with and form judgements on the diverse evidence. Enhancements of the methods for the *judgement of evidence* can also be near-term aspirations, even from a *qualitative* basis.

The adoption of the enhancements which relate to *through-life appraisal* can also be implemented in the near/medium term. Enhancements regarding *procurement and contracting* are also near/medium-term aims.

[30] As outlined within sub-section 7.1.

[31] As outlined within the research output titled *Use of Diverse Software Evidence within a Safety-Critical Software Airborne Qualification Strategy.*

In the medium/long-term the introduction of methods to *manage complexity* can allow the techniques and processing to be reviewed and embedded. This will allow the wider benefits of a method such as the DSF to be adopted. The DSF can capture the information generated as a result of *debate and consensus* amongst stakeholders.

10.2 Suggested Approach to Adopt Diverse Evidence within a Software Assurance Qualification Strategy

To utilise diverse evidence within a MOD platform/system qualification strategy there are a number of key points that should be understood. There is a requirement to comprehend *why* diverse evidence is suitable for certain mitigations and there is a requirement to articulate a *justification* for such an approach.

The information that follows is an extract from guidance for DT Desk Officers written by the RE and the EngD Industrial Supervisor[32]. The guidance has been published within DE&S to assist DTs with their diverse evidence approaches. The full guidance document can be found within the research output white paper titled *Use of Diverse Software Evidence within a Safety-Critical Software Airborne Qualification Strategy.*

The points below are not exhaustive but may act as a method to inform a qualification strategy. Relevant to all of the following points is that support and/or direction should be gained from suitable SMEs to assist the DT to understand and/or deliver the diverse evidence strategy. SME input should be sought for any of the following points if there is insufficient knowledge within the DT.

1. Engage with LRU vendors (and/or sub-vendors) to articulate the MOD evidential requirements and to understand the level of conformity (and the ability for MOD to access such data).

2. Explore options/feasibility of closing any process evidence divergences (if they exist).

3. Gather information on the availability of wider, diverse, evidence to support a particular safety claim. Again, gather an understanding of the ability for the MOD to access such data.

4. Understand what level of confidence can be gained from the available process and non-process evidence. Understand how the wider evidence mitigates, either partially

[32]Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems.

or fully, the shortfalls with the process evidence and how such an approach would be reasoned.

5. Engage with the ITE and MAA in order to articulate and justify the approach which uses diverse evidence. Generate the relevant documentation which formalises the planned method, e.g. Type Certificate Baseline (TCB), MCRI, or Special Condition.

6. Gather the evidence in support of the planned method, e.g. TCB, MCRI, or Special Condition. Contracting for evidence with relevant vendors will need careful management if the type of evidence being requested is not part of the regular information exchange between the MOD and the vendor. In addition, a Request For Information (RFI) with lead procurement nations will potentially need to be articulated for in-service information.

Assistance should be sought from SMEs if there is not the necessary awareness within the DT to perform any of the above points and/or to construct the diverse evidence argument.

The use of diverse evidence is certainly not an easier route than the use of process-based claims. However, it does provide an alternative means to gaining a suitable level of confidence, allows an understanding to be gained of associated risks, and assists with the delivery of capability.

10.3 Challenges to the Adoption of the Methods

It is recognised that the enhancements which have been defined will require amendments to the way in which assurance is currently adopted from a policy and technique perspective. Therefore, there are naturally going to be challenges in the adoption of the enhancements due to the breadth and depth of the findings. Not all of the challenges will be equal in terms of the level of disruption to the current approaches and some advocate a more *evolutionary* change rather than *revolutionary*.

- At present SMEs make judgements against a set of pre-defined process objectives. If diverse evidence is adopted then the judgements will be based upon the SMEs own opinions and the 'objectives' which are to be reached are formed by the SME themselves. This may place additional risk on the SME and so a collective responsibility from the stakeholders could share the judgements, and therefore the risks. There will also be costs associated with any activities needed to train and up-skill staff to be able to provide defensible *opinions*.

- Traditionally the safety assurance domain is very much risk averse and places confidence in pedigree and known evidence. There is a need for further research to devise further robust arguments to adopt a wider diverse approach to ensure that suitable confidence and momentum can be built within the software safety communities.

- There are understandably standard/guideline and policy considerations when instigating changes. However, there are also the less tangible factors. e.g. influencing the mindsets of individuals to be more favourable to the diverse evidence concept. The further implementation of the findings within this thesis will assist with this activity.

- At present the process-based approach to assessment is the default position for the standards/guidelines which are adopted. There needs to be a change to adopting diverse evidence as a *first port of call* via continued use of the thesis outcomes within a wider range of contexts. There is an industry based upon supporting the objectives within guidelines such as DO-178C (e.g. training, consultancy, tools etc) and there would need to be a change in how the concepts of software safety assurance are articulated to those within the industry.

Chapter 10 has partly responded to the research sub-question: *What observations and recommendations can be made on how to implement a software safety assurance evidence argument and how to inform a UK defence software safety assurance strategy?.*

Chapter 11

Research Review and Contributions to Knowledge

This chapter will revisit the research questions, assess progress against these questions, state the contributions to knowledge of the research, the impact, and reflect on the efficacy of the research approach. This chapter contains three broad sections:

- *Research Requirements: A Review.* States the original argument for why the research was needed (i.e. why the intervention was required). Also, provides the progress made against the research questions.
- *Contributions to Knowledge and Research Impact.* States the variety of research outputs which have provided novel contributions to knowledge in the software safety assurance *academic* domain and states the *industrial* impact of the research.
- *Research Legitimacy and Reflections.* Examines the *quality* and *validity* of the work, limitations and potential further work to the research, and autobiographical reflections.

11.1 Research Requirements: A Review

11.1.1 Restatement of the Argument for Intervention

As safety-related systems will contain increasingly more software and are to become ever more reliant on this software, it is imperative that the software can be *assured*. This allows those that regulate, procure, and operate the software to have confidence that any software failures which lead to *damage* only occur at an acceptable rate. *What* different types of evidence support this confidence and *how* should this evidence be structured, judged, and combined?

If suitable approaches are defined then *how* should any identified military software assurance domain *enhancements* be implemented?

11.1.2 Research *Grand Tour* and Sub-Questions: Progress

The thesis has been founded upon a robust research strategy which allowed a set of research questions to be devised. These add value to the software safety assurance domain. The structure of the research strategy has allowed the questions to be responded to in a systematic way using a variety of research paradigms. The activities answered the research *grand tour* question:

What enhancements can be made to the current UK defence domain's software safety assurance approaches for capturing and judging supporting evidence?

For each of the research sub-questions there has been significant progress made[1]:

1. *What is the current approach to system safety assurance within the UK defence domain and are there alternative system-level approaches?*[2] The current MOD approach to safety management has been explored by the RE with an understanding gained of the context and adoption of safety case practice within MOD. The RE has also explored the current practice to how PEs are assessed. A new framework was developed, termed the SLF, which provided a potential solution to constructing safety arguments from a SoS approach.

2. *What is the current permissible software safety assurance evidence within the UK defence domain and related domains?*[3] Lessons were identified from the methods adopted by safety-critical (e.g. civil nuclear) and non-safety critical (e.g. criminal justice) domains which make decisions strongly underpinned by evidence. These lessons were used by the RE to inform the implementation of the DSF. The REs work partly identified a potential solution to a staged adoption of MC processors within the safety-critical domain. The solution implements technical design features which are favourable in terms of obtaining wider diverse evidence.

3. *What software safety assurance evidence is relevant/admissible and what are the underpinning principles for the use of such evidence?*[4] Potential permissible evidence

[1]Sub-section 11.2 within this chapter contains more detailed information on the *contributions to knowledge* in relation to the research sub-questions.

[2]Sub-question responded to within Chapter 3.

[3]Sub-question responded to within sub-sections 4.2, 5.2, 5.3, and Chapter 6.

[4]Sub-question responded to within sub-sections 7.1 and 7.2.

was identified which could inform a software safety assurance argument in the context of military airborne platforms. Underpinning principles were determined[5] for the permissible evidence which state how such evidence should be gathered and used.

4. *What are the unintended consequences of adopting incorrect metrics when forming decisions and how can system/evidence relationships be communicated to stakeholders?*[6] The measurement of data leads to the generation and management of metrics. Lessons were stated regarding the unintended consequences of metrics. In addition, a model was devised to conceptualise elements within an assurance approach; i.e. the systems, the evidence, and the relationships between the two. This is captured in a *Wheel of Qualification*[7].

5. *What is a suitable structure for software safety assurance evidence and can mathematically derived approaches inform how judgements are made on the evidence and for proposing alternative/optimised solutions?*[8] A framework was developed, termed the DSF. This provided a means to structure the evidence under review and to allow judgements on confidence to be formed on the evidence. This was achieved via the selection of a number of key attributes, e.g. *contribution*[9], which can be assessed within a number of mathematically calculated FISs. Visualisations of the evidence *and* the subsequent judgements were also implemented. The DSF also allowed judgements to be explored via the use of *what-if* analysis[10]. It ensured efficient evidence collection via mathmatical optimisation techniques, i.e. GAs. An initial process flow was devised to prompt action for the gathering of evidence.

6. *What observations and recommendations can be made on how to implement a software safety assurance evidence argument and how to inform a UK defence software*

[5]For example, the need to conduct continual monitoring of a system through-life to maintain or improve the prior belief gained in the process-based evidence.

[6]Sub-question responded to within sub-sections 7.3 and 7.4.

[7]The *Wheel of Qualification* is a visualisation/model which allows stakeholders to comprehend and debate a varied set of evidential data/sources with a view to drive improved decision making.

[8]Sub-question responded to within sub-section 4.3, Chapter 8, sub-sections 9.4 and 9.5.

[9]The notion of *contribution* captures the level of influence, or the level of the direct bearing, that the specific evidence has on its broader evidence; e.g. MC/DC as child evidence may feed into the broader parent category of coverage testing.

[10]There are a range of definitions for the terms *what-if* and *sensitivity* analysis with some sources using the terms interchangeably, e.g. Bujoreanu (2011). However, for the purpose of this research *what-if* analysis is defined as changing variable values and/or changing the relationships among the variables to allow the results to be observed (Bagad, 2009). *Sensitivity* analysis is defined as a special case of *what-if* analysis in that with *sensitivity* analysis only *one* variable is to be changed at any time to observe the impact of *small* changes (Bagad, 2009). The research adopted *what-if* analysis methods as this allowed greater scope to assess changes to *multiple* forms of evidence and the supporting variables.

safety assurance strategy?[11] As a result of the DSF development, case studies, and exploratory testing a number of observations were made on the relationships between the attributes. The observations can also act as guidance for deriving software safety assurance from diverse evidence. From the overall set of research activities a set of evidence-based enhancements were generated for the military software safety assurance domain. The research outcomes were informed by the creation of a range of supporting concepts, such as frameworks (e.g. SLF and DSF) and models (e.g. *Wheel of Qualification*).

Figure 11.1 shows the mapping between the research sub-questions and the relevant Chapters and sub-sections of this thesis.

11.2 Contributions to Knowledge and Research Impact

The thesis has provided methods and success factors to *enhance* the safety assurance process[12]. This concept is important to note as the context of any suggested enhancements are to expand upon an existing safety assurance process[13]. The research has investigated how diverse evidence can be gathered, judged, and implemented within the military software assurance domain.

The variety of research outputs has provided novel contributions to knowledge in the software safety assurance *academic* domain. They have also been implemented in an *industrial* setting within a military software assurance context. They have led to realisable and tangible benefits and in the immediate term the concepts from the research have already *directly* informed the procurement strategy for a number of military systems and the research has provided *confidence* to those that adopt such a diverse approach[14].

The research findings advocate quite fundamental changes to how evidence is treated and judged. These changes may require a more gradual introduction, most likely supported by further studies and research. The research has not only clearly articulated why there is a need to change but has also *provided solutions* on how to enact such change with the tools to support the adoption of diverse evidence. A key impact of the research is that the premise of adopting diverse evidence can be taken forward to inform customer qualification assessments.

[11] Sub-question responded to within sub-section 9.6 and Chapter 10.

[12] The use of the term *enhance* is defined as to "*further* improve" (OED, 2018*c*).

[13] Although the current research on the *quantification* of assurance confidence has a number of identified weaknesses. See sub-sections 4.2 and 9.7.2.

[14] Supported by technical reports such as Standish, Hadley and Lennon (2017).

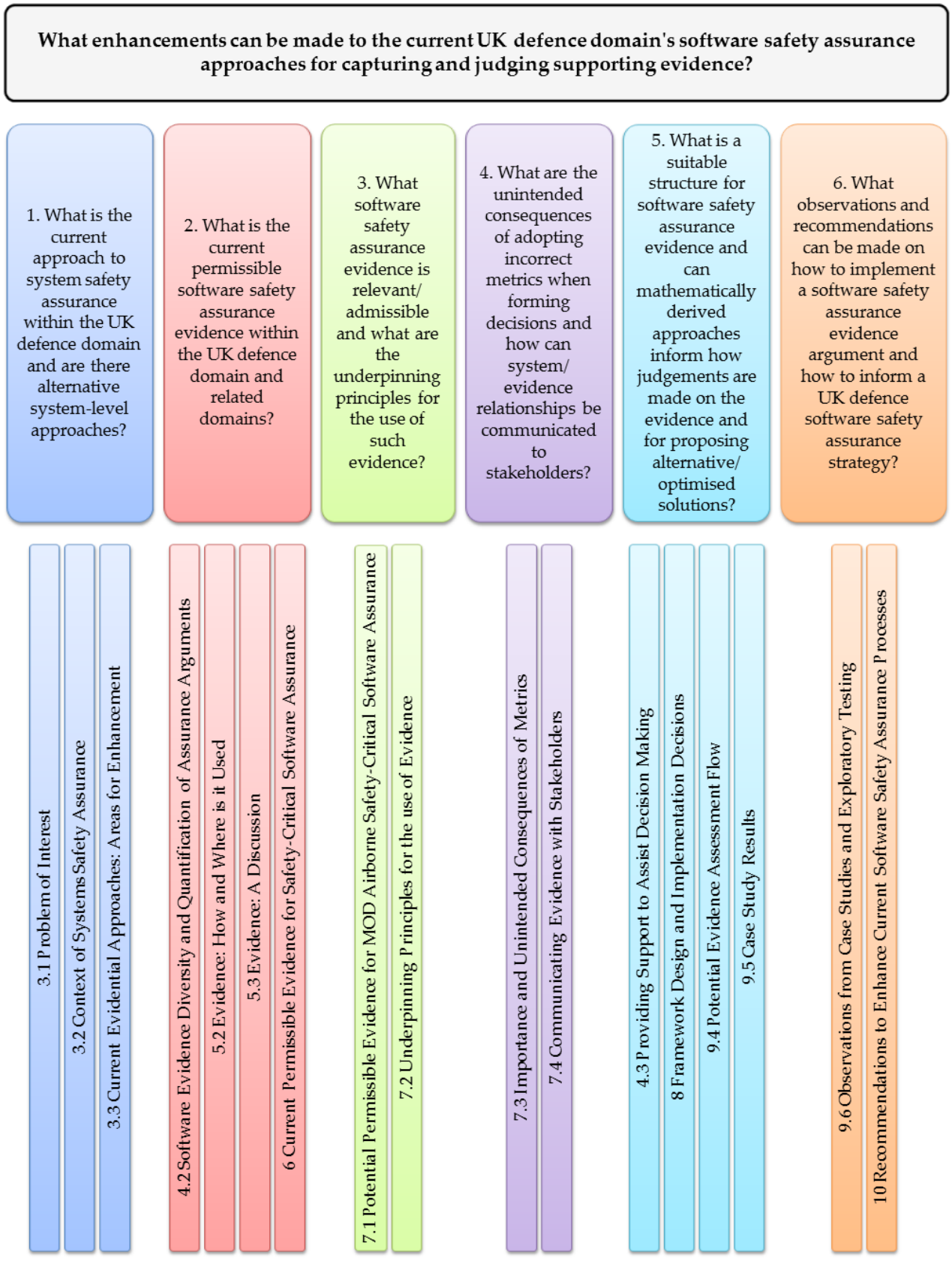


Figure 11.1: Research Sub-Questions Mapped to Thesis Chapters and Sub-Sections

The types of contributions that the research has generated are shown in Figure 11.2[15].

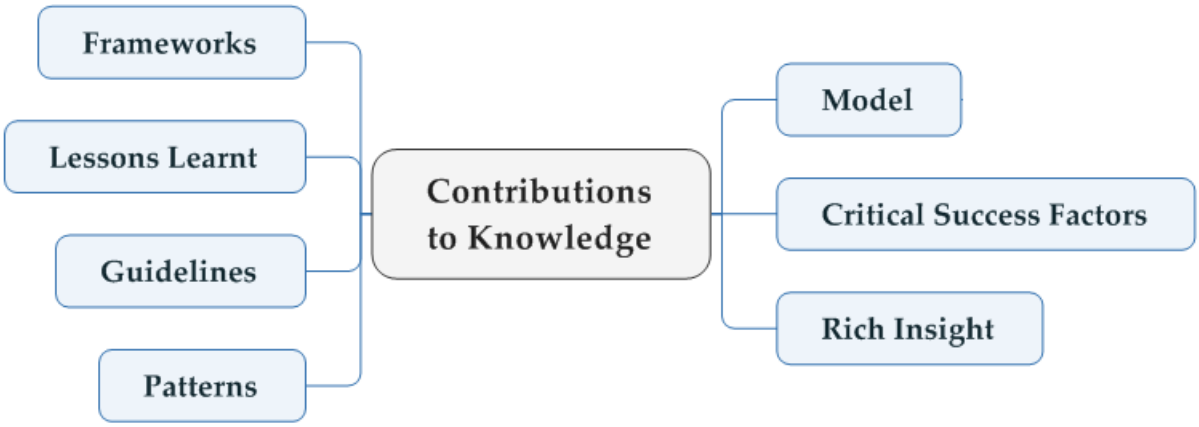


Figure 11.2: Types of Contributions to Knowledge Generated by the Research

All of the contributions to knowledge that the research has generated have value, however they do not all have an equal impact. The key contributions from the research are:

- A pattern, or *blueprint*, has been developed with a set of attributes (e.g. *sufficiency*) which captures the key properties of any evidence that forms part of a software safety assurance approach. The choice of attributes and the method of combination has not been adopted within the software safety assurance domain prior to this research.
- A DSF has been implemented which provides a usable and practical tool for decision makers to construct diverse assurance arguments based upon *quantitative* judgements on the evidence characteristics. Such a tool with such a richness of features has not been implemented within the software safety assurance domain prior to this research.
- All of the research activities have allowed the RE to develop a set of clear and concise recommendations. These will allow MOD and other safety-critical domains to adopt diverse software evidence in order to implement systems which reside within a non-process based solution space[16]. Such a breadth of considered recommendations have not been developed for the software safety assurance domain prior to this research.

[15]The types of contributions have been derived from Presthus and Munkvold (2016).
[16]See sub-section 1.1 for further information.

11.2.1 Frameworks: Decision Support Framework (DSF) and Safety three-Layered Framework (SLF)

A framework can act as a conceptual entity to serve to support analysis or discussion (Presthus and Munkvold, 2016). The frameworks developed for this thesis[17] have enabled observations to be made which has generated enhancements to the use of diverse evidence to support software safety arguments.

11.2.1.1 DSF

The DSF contains a range of features to allow stakeholders to gather and judge evidence. It provides support to allow the solution space to be explored. The DSF contributions include the non-trivial attribute selection and relationships process, methods to structure and reason with the evidence, and to allow *what-if* analysis and optimisation to be conducted. A key aspect of the DSF is that it provides an *intuitive* method to capture evidence judgements and the approach allows confidence to be propagated in a *consistent* way. The DSF was implemented by the RE and has a number of key stages.

1. *Evidence identification.* A review of how evidence is adopted within related safety domains and non-safety domains[18] allowed an initial set of permissible evidence[19] to be included within the DSF. Importantly the framework then also allows for additional evidence to be defined and captured by stakeholders.

2. *Evidence attributes.* Suitable metrics were determined[20] which allowed judgements to be formed on the evidence. The chosen attributes reflected the features of the evidence and also determined how the attributes are combined to derive the *confidence* in multiple and diverse evidence. The chosen attributes were also a driver to *how* diversity can be achieved[21]. The right balance and combination of *all* attributes can provide a satisfactory level of evidence diversity.

3. *Evidence data states.* Not all evidence is equal in supporting a qualification argument and different data states[22] were devised. The data/evidence states were successfully

[17] See sub-section 3.3.2 and Chapter 8.

[18] See sub-sections 5.2, 3.2.1, 3.2.2, 3.2.3, 6.1, 6.2, and 6.3.

[19] Evidence included, but was not limited to: PSH, staff competencies, and software architecture complexity. See sub-section 7.1.

[20] The attributes are: *confidence, quality, contribution, sufficiency*, and *independence.* See sub-section 8.2.2.

[21] See sub-section 8.2.2.6.

[22] The data states are: *extant, obligatory*, and *ancillary.* See sub-section 8.2.3.

used within the DSF to determine the evidence to be gathered, the value of the evidence, and how persuasive the evidence could be to stakeholders for the assurance argument.

4. *Methods to reason under uncertainty.* There are a range of approaches to assess evidence and to allow judgements to be formed. The causes of uncertainty were investigated by the research[23]. The DSF adopts an approach based upon Fuzzy Logic and FISs which are proportionate to the problem. They are suited in situations which evidence is itself fuzzy in nature, they allow decision making with estimated values under incomplete or uncertain information, and they can inherently account for noise in the data[24].

5. *Structure of the reasoning approach.* A number of FISs[25] were developed to *combine* the evidence and the evidence attributes which were defined (e.g. *sufficiency*)[26]. The FISs are repeated for all of the relevant evidence within the structure. The FIS calculations implemented by the DSF were deliberately designed to allow for more simplified and structured assessments to be conducted by stakeholders. This had benefits in terms of allowing stakeholders to traverse the data tree and the node values using logical steps which assists with the comprehension of the evidence and the structure.

6. *Visualisation approach.* The DSF provides informative data to the decision maker via a Linkage Diagram[27]. The Linkage Diagram is suitable as it allows stakeholders to capture and structure evidence in a logical manner and allows methodical analysis of the data. In addition, the Linkage Diagrams allow a substantial level of information to be viewed. This allows a holistic perspective to be gained as the supporting assurance evidence is being assessed. Novel visual indicators developed as part of the research allow decision makers to determine the quantified values of the evidence characteristics[28]. The visualisation showed the level of confidence in the evidence at the node level. It also demonstrated how evidence attributes propagate through the tree which, importantly, identifies particular causes of concern or areas of strength.

7. *Overheads associated with evidence changes.* Any evidence may have *theoretical* value to generate *confidence*. However, there are also *practical* aspects to obtaining such

[23]This includes, but is not limited to, having a *lack of information, an abundance of information,* and a *vagueness in language.* See sub-section 8.2.4.1.

[24]Other methods considered included: *Bayesian Theory* and *DST.* See sub-section 8.2.4.4.

[25]The FISs included: Child Node Confidence, Sibling Nodes Assessment, and Parent Node Quality.

[26]See sub-section 8.2.5.

[27]Other visualisation methods considered included, but were not limited to: *arc diagrams*, *chord diagrams*, and *network diagrams.* See sub-section 8.2.6.

[28]See sub-section 8.2.6.2.

evidence. This includes the *time* taken to generate the evidence, the *cost* to obtain such evidence, and the *quality* that can be achieved with any change to the evidence[29]. These factors were classed as *change overheads* and featured within the DSF optimisation calculations to inform *realistic* and *practical* solutions for the decision maker.

8. *Implementation of an optimisation method.* The process of optimisation is to select the best element, or set of elements, from a set of available alternatives. The DSF implemented a process to devise optimal sets of evidence in a novel approach. There are a number of data optimisation approaches to achieve this process[30]. A GA approach was determined to be of most value as it has a greater success of finding the global optimal.

9. *Data interrogation options.* A key element to the DSF is the requirement to allow decision makers to conduct *what-if* analysis and to explore scenarios. A rich set of options to perform such analysis were devised. The options were based upon realistic outcomes to ensure that the solutions were viable[31].

Such an end-to-end framework has not been devised previously for the software safety domain. In relation to other concepts which have been proposed in this area, sub-section 4.2.3 contains a substantial review of the existing research. Sub-section 9.7.2 details how the selection and combination of attributes within the DSF differs from existing methods.

11.2.1.2 SLF

Modular safety cases provide a number of benefits over traditional monolithic safety cases, e.g. ease of construction and a focus on integration boundaries (IAWG, 2010). Previous research in the area of modular safety cases attempted to provide more structured and efficient methods but there were still limitations, e.g. arguments were not supported by evidence. The SLF[32] provides a contribution to addressing the limitations of existing methods by applying a modularised approach at a systems level. This allows a greater level of detail to be exposed for the assurance of safety-critical system components. The SLF consists of a flow of information which is fed from the top level down to populate each stage: modular safety cases, engineering models, and detailed analysis such as formal models.

[29] See sub-section 8.2.7.

[30] Other optimisation techniques which were considered included, but were not limited to, *SA*, *PSO*, and *Harmony Searches*. See sub-section 8.2.8.

[31] See sub-section 8.2.9.

[32] The approach was published in two papers. The papers were written in collaboration between the RE, two Dstl colleagues (Paul Caseley [Dstl, Senior Fellow] and Dr Mark Hadley [Dstl, Senior Principal Scientist in Software Systems]), and a colleague from AWE (Helen Auld).

The SLF helps to form a judgement on the interface interactions, how the relationships occur, the formal proof of the relationships, and how the relationships can be met. It enables dependency relationships to be defined at appropriate supply interfaces and at differing levels of the SLF. The SLF supports the principle of adopting wider diverse evidence to achieve a suitable level of safety assurance for a SoS; this is also of relevance to the lower-level software safety assurance domain[33]. The concept of the SLF illustrates that safety processes adopted by the MOD are subject to ongoing challenges. Issues such as multi-national procurements and the use of legacy elements need to be considered as part of the methods to develop the wider SMS. These challenges are also prevalent in the evidence which is required for software and CEH assurance.

11.2.2 Lessons Learnt

An important element to the thesis is establishing lessons which can be used within a number of domains which use judgements on evidence to be a key element in decision making. This includes the MOD assurance domain, the wider safety-critical domains, and a number of non-safety critical domains. The following thesis activities have contributed to these goals.

1. *Adoption of Diverse Evidence.* The research has outlined a number of factors which drive the requirement for diverse evidence within the software safety assurance domain. Such factors include: the continued and necessary adoption of novel technologies, continuing procurements via international partners and vendors, adoption of COTS and MOTS equipment, and the desire within wider domains to utilise pre-existing qualification evidence[34].

2. *Adoption of Evidence Within Non-Safety Related Domains.* Observations have been made regarding the use and types of evidence within *non-safety related domains* and how these relate to MOD software safety assurance practice. A number of domains adopt an evidence-based approach within a variety of contexts and the findings from the non-safety domain evidence attributes were utilised in the research. This is in addition to taking account of how other domains assess evidence *confidence*. Lessons have been learnt from understanding evidence use within the criminal justice system, healthcare and medicine, and government policy and strategy[35].

3. *Adoption of Evidence Within Safety-Related Domains.* Similarly to the lessons learnt from the non-safety domain there are also observations made within the thesis regarding

[33]See sub-section 3.3.2.
[34]See sub-sections 3.1.1 and 3.3.
[35]See sub-section 5.2.

the use and types of evidence within *safety-related domains*. Domains such as: the civil air traffic services, civil aviation, automotive, health information technology, and rail (signals and rolling stock) have all garnered points relevant to MOD software safety assurance practice. The use, type, and hierarchy of attributes and evidence differs between domains and these insights have informed the thesis development and contain lessons for wider research activities[36].

11.2.3 Guidelines

The thesis has provided informative results not withstanding the constraints related to commercial sensitivities and IP considerations. The outputs have directly informed DTs, e.g. via directing qualification strategies, and also the wider software assurance community within the MOD, e.g. via informative white papers. In the case of the MC processor research, which adopts a staged assurance solution with wider more diverse supporting evidence, the information has been shared with a range of safety domains.

1. *Types and Benefits of Diverse Evidence.* The research has generated guidance on the types and benefits of diverse evidence for a software safety assurance approach. Importantly, the guidance included a number of considerations to ensure that diverse evidence can be contracted for appropriately and to allow direct informed discussions with suppliers[37].

2. *Staged Approach to MC Processor Qualification.* Guidance was generated on a potential solution for the assurance of a novel technology, that of MC processors[38]. The staged approach adopts the use of diverse evidence to provide confidence from non-standard sources. The findings were shared with a safety domain audience[39]. The guidance has also influenced further research activities[40] and has directly informed discussions with suppliers to ascertain how their system architecture could be developed to support a qualification argument.

3. *Underpinning Principles to Adopt Diverse Evidence.* Guidance was generated on a number of underpinning principles which must be factored into any use of diverse

[36]See sub-section 6.2.

[37]See the research output titled *Use of Diverse Software Evidence within a Safety-Critical Software Airborne Qualification Strategy* and the research informed customer deliverable Standish, Hadley and Lennon (2017).

[38]Guidance written in collaboration between the RE and the EngD Industrial Supervisor (Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems).

[39]Illustrated within the research output titled *Multi-Core (MC) Processor Qualification for Safety Critical Systems.*

[40]For example, Imperial College London's MC test harness which was funded by Dstl. See the following for more information: https://github.com/mc-imperial/multicore-test-harness.

evidence. This ensures that the evidence is implemented correctly, consistently, and appropriately. The principles also provide guidance on the practical use of diverse evidence[41]

11.2.4 Pattern: Combination and Relationships of Evidence Attributes

Patterns, or *blueprints*, allow pre-existing methods to be re-used within frequently encountered problems. They can have many benefits to a range of stakeholders as they allow consistent and repeatable solutions to be developed. An example in the software engineering domain is with design patterns[42].

The research devised a reusable pattern which outlined the evidence attributes which are agnostic to the form of evidence/structure. The chosen attributes allow the characteristics of the evidence to be understood as well as the relationship of the evidence to other forms of evidence. The pattern also drives how the *confidence* in the evidence is judged as it dictates how the attributes are combined[43].

11.2.5 Model: *Wheel of Qualification*

Within this research a model is defined as a construct which expresses relationships among concepts (Presthus and Munkvold, 2016). In this context, a model can allow complicated concepts to be articulated.

The adoption of diverse evidence provides an increased solution space. However, these can traditionally not have full process-based qualification evidence. Therefore, further forms of evidence are required to gain a suitable level of software safety assurance confidence. This could lead to issues for the *perception*, *interpretation* and *comprehension* of the evidence, the source of the evidence, and how it relates to specific LRUs.

To address this issue a visualisation was created which was termed the *Wheel of Qualification*[44]. The visualisation facilitates informed dialogue with a number of stakeholders (e.g. MAA, DT etc.) who require an insight into the software/CEH evidence for the individual LRUs of a platform. The model shows the relationship between the various evidence strands, in relation to an ITE, supplier, and LRU. The model can allow areas of particular strength or weakness to be shown both in respect to the LRUs of interest and the associated evidence

[41] See sub-section 7.2.

[42] See, for example, Helm and Johnson (2015).

[43] See sub-sections 8.2.2 and 8.2.5.

[44] Model developed in collaboration between the RE and the EngD Industrial Supervisor (Dr Mark Hadley - Dstl, Senior Principal Scientist in Software Systems).

which is being judged[45].

The *Wheel of Qualification* has driven the requirements discussion at a number of project workshops and has allowed the RE to articulate where further evidence is required. The *Wheel of Qualification* is the cornerstone of an existing project qualification strategy. There is no concept like this in current use within the software safety assurance domain.

11.2.6 Critical Success Factors (CSFs): Recommendations to Enhance Software Safety Assurance Processes

Critical Success Factors (CSFs) describe activities which are necessary to ensure that a positive outcome is reached. The term is based upon business and management principles; however, the term is apt for the adoption of diverse evidence for software assurance.

The devised evidence-based CSFs (such as revised skill sets of SQEP staff) are based upon 7 key themes (such as how evidence complexity is managed). Over 20 recommendations are made to allow diverse evidence to be fully exploited within a software safety assurance strategy[46].

Non-process based software safety assurance evidence can be more difficult to measure which results in a number of challenges to the adoption of such an approach. The research has demonstrated the benefits of a diverse evidence strategy and it has also provided numerous solutions to allow it's adoption, e.g. the DSF. The recommendations, or CSFs, indicate *how* the practical solutions provided by the research can be implemented and *why* they should be adopted.

11.2.7 Rich Insight: Importance and Unintended Consequences of Metrics

Metrics can provide valuable information to decision makers and those that wish to draw conclusions from the data. However, metrics that are not chosen well can have negative consequences if they lead to incorrect decisions. They can also lead to incorrect interpretations, especially when they form a safety assurance argument.

An insight has been provided on the concept of 'technical debt' and it has illustrated how the measurement of performance data can lead to unintended consequences[47]. The sub-section regarding the unintended consequences of metrics has a focus on software maintenance, however the principle of being cautious with the use of metrics still holds. The

[45] See sub-section 7.4.

[46] See sub-section 9.6 and Chapter 10.

[47] Written in collaboration between the RE and a Dstl colleague (Rob Ashmore - Dstl, Senior Fellow).

research provides support for ensuring that any measure of assurance must take into account the unintended consequences which could arise. Metrics adopted to make judgements must be used with an understanding of their limitations[48].

11.3 Research Legitimacy and Reflections

11.3.1 Research Quality

There are a number of measures which can provide an indication of research *quality* and it is important for REs to make an appraisal of the measurements to refine any points within the thesis itself. A common set of measures for *quality* are *validity*, *reliability*, and *generalisability* (Leung, 2015, Ali and Yusof, 2011, Heale and Twycross, 2015).

- *Validity.* The concept of *validity* spans a number of areas, with the focus on the *appropriateness* of the tools, processes, and the data. Is the research question valid for the desired outcome? Is the choice of methodology appropriate for answering the research question? Is the design valid for the methodology? Is the data analysis appropriate? Are the results and conclusions valid for the context? (Leung, 2015).

  The research *grand tour* and sub-questions were developed via an iterative process which was refined as more knowledge became available via document analysis and further qualitative methods. The structured research strategy generated a gradual and stepped flow of information and salient observations. These fed into a number of enhancements which could be made to the defence software safety assurance process. The research question was appropriate for the desired and obtained outcomes. The research methodology was also exploited to develop a theory which was then tested via observations and case studies. The staged outputs from the activities, and their variety, were proportionate and purposeful to acquire a suitable level of data to inform the research outcomes. The research was centred on a military assurance domain with applicable findings from other domains taken into account to apply to the military context.

- *Reliability.* Refers to the exact *replicability* of the processes and the results. Such a definition for reliability is challenging and counter-intuitive but the key is *consistency.* A margin of variability can be tolerated as long as the methodology consistently yields data which is ontologically similar but may differ in richness and ambience within similar dimensions (Leung, 2015). This is the nature of research. Research decisions

[48] See sub-section 7.3 and the output regarding the unintended consequences of metrics.

based upon subjective feedback, e.g. from SMEs, will influence the direction of the research.

Within a research process there are varying types of analysis which can flow from the development of creative concepts. These can be based upon insights to learning via data analysis or observations. A key element to any research approach is the ability for the RE to develop concepts with *imagination and insight* seen as a researcher quality (Cauvery et al., 2003)[49]. Aspects of the approach taken to respond to a research *grand tour* or sub-questions may not be repeatable (this is, to a degree, based upon the *generic* and *specific* skills of the researcher) but they should certainly be *defensible* (Rolfe, 2006). The repeatability element is in relation to the results obtained from any data and the ability to infer findings from any quantitative/qualitative analysis methods. Importantly, the observations, enhancements, and contributions to knowledge from the research are traceable and so the inferences are repeatable.

- *Generalisability.* The concept of *generalisability* may not be a key driver for some research if it focusses on a specific issue or phenomenon (Leung, 2015). However, there is still value in ascertaining the extent that the research results can be applied to cases, settings, or situations beyond those examined in the study. The refined *grand tour* and sub-questions for the research were based upon safety assurance within a specific domain and for particular platforms, i.e. military airborne. This was a deliberate and conscious decision in order to gain robust findings which were not *diluted* due to expanding the findings to a wider setting. There a various layers to how *generalisable* the study findings can be. The findings and outputs from the study can certainly be used within the software safety assurance domain and there are elements which are valid for the land and maritime domains, e.g. in the treatment and judgement of evidence. Other findings are more domain specific due to them being based upon particular airborne platform practices, e.g. DAOS.

  The concept of *generalisability* is separate to that of *transferability*. *Generalisability* is concerned with the explicit extension of research findings to other domains/settings. Whereas, *transferability* is applied by the *readers* of the research with salient observations being applied to their problem areas to varying degrees (Given, 2008). The outputs and findings from the research will have *transferability* as the contexts within a number of assurance domains may have similar challenges and opportunities to embed

[49]Cauvery et al. (2003) state five *general* qualities which a researcher needs: scientific attitude, imagination and insight, perseverance, quick grasping power, and clarity of thinking. Cauvery et al. (2003) states five further *specific* qualities that a researcher needs in relation to the research itself: knowledge of the subject, knowledge of the technique of research, personal taste in the study, familiarity about the information, and an unbiased attitude.

the use of diverse evidence within their assurance processes. The utility scale (from specific, to general, to transferable) certainly applies to this research; however, there are pertinent findings applicable for each of the scales.

11.3.2 Validity of the Research Implementation

The issue of *validity* is one which is worthy of further exploration as although the overall research methodology and design are claimed to be sound, as proposed above, there are a number of potential threats to the validity.

- There is debate, depending on the research paradigm, on the impact that the researchers *own* philosophical stance has on the research process and outcomes. Within a *phenomenological* paradigm the researcher interacts with that which is being researched. There is also perceived to be benefits to having researcher involvement; with it claimed to be *essential* (Leung, 2015). The experience and role of the RE within the sponsoring organisation meant that there is a form of *validity* to the research observations and from the wider analysis, e.g. via document analysis.

- A number of domains and research approaches, e.g. document analysis, were used to gain information on how safety and non-safety domains conduct evidence gathering and judgement. The lessons and observations were pertinent and allowed salient observations to be made but there could be a query regarding the sufficiency of the data which was gathered. The domains which were not reviewed, e.g. teaching within the non-safety domain, were scoped to assess if significant observations could be gained in relation to the information already collected. The same is also true of the safety domains which had a greater number of findings. It is believed that the domains reviewed provided a *sufficient* level of observations to support the research[50].

- A concept proposed by this thesis is that, to a degree, all evidence is of *relevance* and that there is a large solution space to gain evidence to support a level of safety assurance. The evidence proposed in support of this premise was not exhaustive. The initial set of permissible evidence was gained via document reviews and other qualitative factors to understand the evidence adopted within the wider safety-domains. In addition, the evidence was also relevant to the case studies which were conducted. The DSF provides the ability to add/amend evidence strands which can be assessed via a proposed pattern. Therefore, the theory is scalable to use with wider evidence sets, with this being a decision to attempt to counter this limitation.

[50] In essence, *generalisations* were made from other domains to support the thesis research.

- A consideration for any research is the level of supporting data which is assessed to gain confidence in the research outcomes. Depending on the types of study there may be a large volume of quantitative data to statistically assess, whereas other types will have smaller sample sizes. These variances are understood within the research field, i.e. it is dependent on the *methodological* philosophical assumption. The cases studies implemented and the supporting exploratory testing are deemed *proportionate* for the derived observations and findings. It is believed that additional analysis would have gained further *supporting* evidence rather than *contradictory* or additional observations.

- Intentionally the research approach and implementations allowed a dynamic set of evidence and attributes to be captured for a given system. This approach leads to having a *defensible* position rather than a *repeatable* one as the choices are made using subjective judgement. It is quite possible that given the same sets of evidence that differing attributes will be applied by different SMEs. However, this observation is true for other domains and there is acceptance that there will be a variance of opinion and judgement. The fact that differing opinions will be captured which will influence the results does not invalidate the approach.

11.3.3 Limitations of the Research

An important element to the research is to understand the limitations of the activities and the outputs. The research has resulted in a number of valid and defensible potential enhancements to the software safety assurance domain. However, there are factors to take into account which provides context to the findings. The limitations to the research below are in addition to the *evaluation* of the DSF contained within sub-section 9.7.

1. *Benefit of a wider set of participants.* An aspect which the RE was cautious of was the risk of *group think*[51] and the impact that this may have on the research outcomes. This risk was lessened as the people involved in this research[52] all have varying degrees of experience and by the nature of their roles have expectations in terms of professional conduct and integrity. These principles should hold for when they are providing feedback to those involved in assessing their domains as they should not be easily influenced by others by the nature of their roles. Also, one-to-one interviews and smaller workshops were designed in order to allow more open discussions. However, a wider

[51] A psychological phenomenon where people set aside their own beliefs or adopt the opinion of the rest of the group (Cherry, 2019).

[52] As an example, as part of semi-structured interviews and workshops etc.

set of participants to the research may have provided greater support to the findings and further reduced the risk of *group think*.

2. *Nuances between software design and assurance diversity.* There are well stated benefits of adopting diversity within the *design* of safety-critical systems. Some of the benefits to software design have been *transferred* within this thesis to assuring software safety evidence. However, there may be subtle nuances in the applications and contexts between the two. This may limit the benefits which can be transferred.

3. *Non-exhaustive review of reasoning methods.* The reasoning methods selected (fuzzy logic supported by FISs) were declared suitable within the design decisions[53]. This proved to be the case as valid observations were made on the DSF behaviour and outputs. In addition, suitable enhancements were supported by the identified node/attribute behaviours within the DSF. A range of reasoning approaches were identified and assessed as part of the design decision process, e.g. BBN, DST, etc. However, this review was not exhaustive with some known approaches not under assessment, such as hill-climbing[54]. There are obvious limitations to the review process as other suitable methods may have been established. However, the premise of adopting a *fit for purpose* reasoning method resulted in a suitable choice.

4. *Reliance on expert judgement.* The DSF is reliant on user judgements to capture suitable evidence and attributes. The results of the DSF have been cross-referenced with the findings from SMEs, e.g. via life-cycle process assessment reports. Therefore, the DSF does represent a suitable reflection of SME judgements given the case studies which were implemented. The stakeholders involved in the use of the DSF are SQEP SMEs and therefore some risks are lessened, for example, the integrity of any feedback. However, some of the fundamental issues with capturing expert judgements are still present, e.g. evidence bias. The DSF does, however, attempt to mitigate this by introducing the ability for consensus to be gained by the stakeholders.

5. *Limited data interrogation options for greenfield projects.* The focus of the DSF activities has been on *brownfield* projects and mitigating shortfalls to widen the system solution space. Further data interrogation options could have been provided for further support for evidence generation for *greenfield* projects. The DSF currently supports decision maker(s) to perform data interrogation options for *greenfield* projects; however, tailored options could have been considered further. Nevertheless, there is a need

[53] See sub-section 8.2.4.4.

[54] See the following for further information: https://www.sciencedirect.com/topics/computer-science/hill-climbing.

to balance the DSF being to *support* decisions or as a method to *drive* decisions.

6. *Non-time optimised GA calculations.* From a practical perspective the optimisation operations within the DSF using the R GA package[55] are not optimal. The processing of the GA calculations needs to be refined to ensure a more timely execution. The current processing time did inhibit some of the interactions with stakeholders and it may result in fewer iterations of *what-if* activities being performed if it is not resolved.

11.3.3.1 Potential Limitations with Practical use of the Research

As well as understanding the threats to validity and limitations of the research a number of limitations have been identified regarding the longer-term adoption of the research philosophy and outputs. There may be limitations with putting the research into practice.

1. *Regulators maintaining the same risk appetite.* The premise for the thesis research is underpinned in the fact that the risk appetite within the software safety assurance domain for determining the *validity* of evidence will remain constant or become stronger. At present the regulatory process uses MCRIs and Special Conditions in the event of alternative evidence being adopted for a software assurance safety argument. The theories within this thesis adopts these approaches. If the appetite to accept such evidence changes, e.g. by regulators, then the thesis principles do not hold in full. The DSF/principles are still valid for defining *process* evidence but the wider benefits of exploiting *diverse* evidence would not be realised.

2. *Determining counter-evidence risk via evidence branch reviews.* A key concept with any safety assurance process is how counter-evidence is treated. The degree to which counter-evidence can be mitigated via other forms of evidence is very much open to debate. The current DSF allows counter-evidence to be mitigated via other forms of evidence if the attributes values are configured appropriately by the decision makers. This is a deliberate feature of the DSF. However, this means that the use of the DSF output, the DAL MFs, by stakeholders must take into account the need for them to review evidence branches. This will ensure that where counter-evidence is mitigated that it has consensus. The use of the DSF and the evidence capturing mechanism through-life for a system, as recommended within this thesis, must assess the underpinning evidence and not just the top-level DAL.

[55] See the following for further information: https://cran.r-project.org/web/packages/GA/GA.pdf/.

11.3.4 Further Work for the Research

The research conducted has provided responses to the *grand tour* and sub-questions; however, a number of opportunities have been identified to progress the research findings.

1. *Guidance on what evidence can mitigate shortfalls.* Within the software safety assurance domain a benefit would be an evidence *cookbook* which provides guidance on *what* evidence can explicitly mitigate other evidence shortfalls. There is a need for the varying forms of evidence to be empirically studied to ascertain their effectiveness. As an example, at a software testing level there is evidence that it is the *combination* of approaches which increases testing effectiveness (Hadley, 2013). Such a *cookbook* does not exist and the debate on its content would continue *infinitum*. The DSF and the research findings does not explicitly provide further guidance on this but it does allow acceptable alternative strategies to be considered and analysed[56]. The DSF and the stated enhancements act as a method to allow these concepts to be explored further.

2. *Further review of non-safety domains.* Within sub-section 5.2 there were three non-safety domains subject to review; although the criminal justice system was subject to a review in three sub-areas. A number of further domains to those which were reviewed combine and assess diverse forms of evidence. There is a risk that relevant information to the research may have been excluded. It is a belief that the domains subject to the review were sufficient to gain suitable lessons; however, there is a risk that an expanded review of the non-safety domains may have provided additional information which could have informed the thesis research.

3. *Generate patterns to reduce attribute/relationship selection.* The DSF has deliberately been designed to act as a *scalable* approach to account for numerous evidence strands which may form part of a wider solution space for a platform. However, possible further work could research the ability to apply patterns which *reduce* the attribute/relationship selection process for the parent-child relationships. Such a method would allow a greater number of evidential strands to be considered within any given framework. Obviously, the results of the scaled framework would require suitable validation.

4. *Refine the overhead concept.* As discussed in section 9.7, the use of the evidence *overhead* to shape the reasoning was a greater advantage than originally anticipated. Further activities could be conducted to refine the *overhead* approach to account for a greater nuance of the current contributing factors, e.g. technical risk. A more

[56] In terms of the theoretical and practical implications.

refined approach to the concept *could* significantly influence the reasoning results based upon the interpretation of the evidential changes. Refinement could be to develop further FISs or to conduct further expert analysis to gather metrics for the *overheads* of relevance.

5. *Gain confidence in the inputs to the DSF.* The DSF acts as a method to *record* expert judgement and it allows the processes and techniques to derive this information to be user-defined. Further work could establish suitable methods to gain confidence in the *input* values into the DSF. This would be *prior* to the data entering the framework. Due to the way in which the information is adopted to inform expert judgement the process to gain consensus and group input could be via a formally rigorous process[57].

6. *Capture judgements on extreme evidence types.* The permissible evidence considered within the initial DSF and part of the domain reviews considered evidence pertinent to *existing* assurance claims[58]. This was deliberate as the focus was on allowing the MOD software assurance process to learn from how evidence is gathered and treated within other domains. The focus of the thesis was to develop *practical* enhancements. However, a future approach could be to gain SME feedback on the *extreme* types of diverse evidence and how this may negate shortfalls in process evidence. As an example, the use of High Performance Computing (HPC) capabilities to undertake a substantial and rigorous testing regime to negate risks with an in-house development process. These would create *theoretical* proposals.

7. *Refine how the Wheel of Qualification is reflected in the DSF.* The DSF to date has captured third-party oversight and review activities as separate strands within the evidence structure. A method to reflect the *Wheel of Qualification* more accurately may be to assign any positive inferences, e.g. from oversight activities, *directly* to the evidence under review. As an example, if a third-party review has been conducted on any V&V activities then the V&V strand itself could reflect this increase in confidence. This type of information could be captured by refining the *quality* of the evidence. At present the *Wheel of Qualification* principles are captured in the DSF; however a more efficient and effective mechanism may be possible.

8. *Implement further case studies.* There is scope for further case studies to be implemented to provide further confidence in the research findings and outcomes. The number of cases studies are already stated as a *threat to validity.* Although the cases

[57] Via techniques such as Dynamic Hesitant Fuzzy Linguistic Group Decision-Making (DHFLGDM) which can also capture degrees of certainty in any group decisions (Zhenzhen et al., 2018).

[58] See sub-section 7.1.

studies and exploratory testing were proportionate there could be opportunities to exploit additional feedback. The concept of implementing further case studies is also valid for the SLF; e.g account for humans and their influence of the depth and breadth considerations of SoS assurance.

9. *Alternative overall output types for the DSF.* Another consideration for the DSFs overall output may be that of a reliability figure (or a probability of failure) which would allow direct input into a loss model (or fault tree), for example. Such an output would require less interpretation than a DAL. Although it is recognised that DALs are a recognised measure of confidence for software.

10. *Refine the method to display DSF results.* An accepted and utilised DSF may warrant an improved data entry and visualisation approach. The current method to capture and display information to inform decision making is fit for purpose; however, additional features to assist with the underpinning DSF concepts may be of benefit[59].

11. *Expand the research to wider domains.* The research has, purposefully, focused on the military airborne domain; however, there would be benefit to expanding the research to wider domains to gain relevant feedback. This would also allow those domains to take lessons from the research, in essence this would be the principle of *transferability*. Indeed, the design of the DSF, the *Wheel of Qualification*[60], and the identified enhancements can relate to other domains quite purposefully, e.g. with the method in which the DSF output can inform a traditional Safety Case structure. Within many safety domains the requirements and potential supporting evidence may have overlap, e.g. staff competencies is a cross-domain consideration (if not a formal requirement). Patterns could be developed to capture the evidence strands to account for a pan-safety domain structure which may be validated via expert *consensus*. Non-UK domains could also be explored as differing safety cultures, risk appetites, and tools/techniques can lead to varying methods to address similar safety challenges.

11.3.5 Autobiographical Reflections

The activities conducted to support this research has allowed a number of observations to be made in terms of research approaches (in the generic sense) in addition to observations related directly to the research execution.

[59]R packages such as Shiny would be of value, see the following for further information: https://shiny.rstudio.com/.

[60]The benefit of the model has been recognised by another research programme within MOD which is to further investigate the utility of visualising the qualification status of LRUs.

The structure which a research strategy provides can allow the research to be conducted with a solid premise with the research findings generated with a robust underpinning. Research strategy theory is based upon, it can be argued, quite nuanced and subtle language which requires those within the field (or researchers who are embarking on a substantial research task) to become adept at identifying the position of the research within the various paradigms and associated assumptions. Traditionally, the level of granularity required for the decision analysis for the research strategy can vary between domains and paradigms. However, it is recognised that there is fundamental requirement to implement robust strategies at all levels of analysis. The research process certainly does not fit a linear waterfall model with continuous iteration required. This was, at times, a frustrating task but certainly a rewarding one.

The importance of expert judgement was another element of the research which was a key finding. Any judgement needs to be underpinned by solid evidence and a position of experience/qualification. The ability to shape judgements from a position of knowledge is crucial and is applied to *all* aspects of an assurance argument. This philosophy can inform how the RE interacts with other stakeholders and how information can be reported/presented. Related to this is the concept of experts being *persuasive* in their arguments and using their style to reflect such an approach[61].

There are continuous efforts to assist with *measuring* compliance to forms of evidence. The evidence that is sometimes adopted is not always the most suitable for gaining confidence but can be the most suitable due to its ease of measurement. This allows a commonality of language and, in theory, a relatively straightforward process to gain confidence. However, the technology landscape is becoming very complicated, e.g. due to digital twins, sophisticated System-on-Chip (SOC) architectures etc. New verification techniques are required and due to the maturity of these technologies there is a real need to ensure that SME judgements are fully considered. However, experts do not always agree and this requires a consensus approach. A new reliance on SME judgement may also result in wider opinions being sought, and therefore more judgements being captured[62].

The whole research process has been exceptionally rewarding and the process has altered how the RE approaches complicated problems. The impact of the research has also been to alter how the RE interacts with stakeholders to express judgements on evidence.

[61]Spriggs (2019) provides some interesting thoughts on the ability to make persuasive arguments in the context of safety assurance via *modes of persuasion* (*Logos*, *Pathos*, and *Ethos*).

[62]This is where the DSF and supporting concepts such as the *Wheel of Qualification* may assist.

References

Abdmouleh, Zeineb, Adel Gastli, Lazhar Ben-Brahim, Mohamed Haouari and Nasser Ahmed Al-Emadi. 2017. "Review of optimization techniques applied for the integration of distributed generation from renewable energy sources." *Renewable Energy* 113:266 – 280.
URL: *http://www.sciencedirect.com/science/article/pii/S0960148117304822*

Acuna, S. T., N. Juristo and A. M. Moreno. 2006. "Emphasizing human capabilities in software development." *IEEE Software* 23(2):94–101.
URL: *https://ieeexplore.ieee.org/document/1605185*

AEngD. 2018. "Engineering Doctorate Programmes." WWW.
URL: *http://www.aengd.org.uk/programmes/*

AESMS. 2017. "Acquisition Safety and Environmental Management System: Safety Case and Safety Case Report." WWW.
URL: *https://www.asems.mod.uk/guidance/posms/smp12*

Ali, A. M. and H. Yusof. 2011. "Quality in Qualitative Studies: The Case of Validity, Reliability and Generalizability." *Issues in Social and Environmental Accounting* 5(1):25–64.
URL: *https://www.semanticscholar.org/paper/Quality-in-Qualitative-Studies%3A-The-Case-of-and-Ali-Yusof/54dc92c79a4a3e8cee5c18067f88f405a69c2751*

Alnafjan, K, T Hussain, Gul Khan, Hanif Ullah and Abdullah Alghamdi. 2012. "Evaluating software security risks using fuzzy rule based expert system." *Proceedings of the 21st International Conference on Software Engineering and Data Engineering, SEDE 2012* 0:31–36.
URL: *https://www.researchgate.net/publication/289699206_Evaluating_software_security_risks_using_fuzzy_rule_based_expert_system*

Ammann, Paul and Jeff Offutt. 2008. *Introduction to Software Testing*. 1 ed. New York, NY, USA: Cambridge University Press.

Apple, J. G. and R. P. Deyling. 2012. A Primer on the Civil-Law System. Technical report US Federal Judicial Center.
URL: *https://www.fjc.gov/content/primer-civil-law-system-0*

Ashmore, R. and M. Standish. 2017. "S&SD Team Overview and Potential Engagement." Presentation to SECT-AIR Technical Day.

Atkinson, Roger. 1999. "Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria." *International Journal of Project Management* 17(6):337 – 342.
URL: *http://www.sciencedirect.com/science/article/pii/S0263786398000696*

ATSB. 2011. ATSB TRANSPORT SAFETY REPORT. In-flight upset, 154 km west of Learmonth (WA), 7 October 2008, VH-QPA, Airbus A330-303. Technical report Australian Transport Safety Bureua.
URL: *http://www.atsb.gov.au/media/3532398/ao2008070.pdf*

Avizienis, A., M. R. Lyu and W. Schutz. 1995. IN SEARCH OF EFFECTIVE DIVERSITY: A SIX-LANGUAGE STUDY OF FAULT-TOLERANT FLIGHT CONTROL SOFTWARE. In *Twenty-Fifth International Symposium on Fault-Tolerant Computing, 1995, Highlights from Twenty-Five Years.* pp. 136–.
URL: *https://ieeexplore.ieee.org/document/532625/*

AVSI. 2011. "Exponential Growth of System Complexity." WWW.
URL: *https://savi.avsi.aero/about-savi/savi-motivation/exponential-system-complexity/*

Ayoub, A., B. Kim, I. Lee and O. Sokolsky. 2012. A Systematic Approach to Justifying Sufficient Confidence in Software Safety Argument. In *31st International Conference, SAFECOMP 2012, Magdeburg, Germany, September 25-28, 2012.*
URL: *http://repository.upenn.edu/cis_papers/741/*

Ayoub, A., J. Chang, O. Sokolsky and I. Lee. 2013. Assessing the Overall Sufficiency of Safety Arguments. In *21st Safety-critical Systems Symposium (SSS 13), Bristol, United Kingdom.*
URL: *https://repository.upenn.edu/cis_papers/744/*

Bagad, V. S. 2009. *Management Information Systems.* Technical Publications.

Bagshaw, S. and R. Bellomo. 2008. "The need to reform our assessment of evidence from clinical trials - A commentary." *Philosophy, Ethics, and Humanities in Medicine* 3:23:0.
URL: *https://www.ncbi.nlm.nih.gov/pubmed/18826605*

Baley, K. and D. Belcham. 2010. *Brownfield Application Development in .NET*. Manning.

Baraldi, Piero, Luca Podofillini, Lusine Mkrtchyan, Enrico Zio and Vinh N. Dang. 2015. "Comparing the treatment of uncertainty in Bayesian networks and fuzzy expert systems used for a human reliability analysis application." *Reliability Engineering & System Safety* 138:176 – 193.
URL: *http://www.sciencedirect.com/science/article/pii/S0951832015000265*

Barker, S. 2018. Call that an argument? In *SCSC Seminar: COTS, Legacy, and Reuse*.
URL: *https://scsc.uk/file/576/04---Stephen-Barker---Call-that-an-argument-1.pptx*

Baudry, Benoit and Martin Monperrus. 2015. "The Multiple Facets of Software Diversity: Recent Developments in Year 2000 and Beyond." *ACM Comput. Surv.* 48(1):16:1–16:26.
URL: *https://arxiv.org/pdf/1409.7324.pdf*

Bishop, P. 1995. Review of Software Design Diversity. Technical report Adelard.
URL: *https://www.adelard.com/assets/files/docs/divchap.pdf*

Bishop, P. and R. Bloomfield. 1998. A Methodology for Safety Case Development. Technical report Adelard.
URL: *https://pdfs.semanticscholar.org/2fec/ac8b034d23a3c66bdf3e0b86739dbea378d4.pdf*

Blockley, D and P Godfrey. 2000. *Doing It Differently: Systems for Rethinking Construction*. ICE Publishing.

Bloomfield, Robin and Bev Littlewood. 2006. *On the use of diverse arguments to increase confidence in dependability claims*. London: Springer London pp. 254–268.
URL: *http://openaccess.city.ac.uk/1601/1/INUCE_multi-leg_v15.2.pdf*

BMJ. 2013. "An introduction to patient decision aids." *BMJ* 347:0.
URL: *https://www.bmj.com/content/347/bmj.f4147*

Bond. 2018. "An introduction to the principles for assessing the quality of evidence." WWW.
URL: *https://www.bond.org.uk/data/files/Effectiveness_Programme/120828Full_Bond_checklist_and_guide.pdf*

Botzoris, George, Kyriakos Papadopoulos and Basil Papadopoulos. 2015. "A method for the evaluation and selection of an appropriate fuzzy implication by using statistical data." *Fuzzy Economic Review* 20:19–29.

URL: *https://www.researchgate.net/publication/316216654_A_method_for_the_evaluation_and_selection_of_an_appropriate_fuzzy_implication_by_using_statistical_data*

Bouissou, M., F. Martin and A. Ourghanlian. 1999. Assessment of a safety-critical system including software: a Bayesian belief network for evidence sources. In *Annual Reliability and Maintainability. Symposium. 1999 Proceedings (Cat. No.99CH36283)*. pp. 142–150.
URL: *https://ieeexplore.ieee.org/abstract/document/744110/*

Box, G.E.P. 1979. Robustness in the Strategy of Scientific Model Building. In *Robustness in Statistics*, ed. ROBERT L. LAUNER and GRAHAM N. WILKINSON. Academic Press pp. 201 – 236.
URL: *http://www.sciencedirect.com/science/article/pii/B9780124381506500182*

Brito, M. 2009. Safety Critical Software Process Improvement by Multi-Objetive Optimisation Over BBN Contraints PhD thesis University of Bristol.

Brosgol, B. and D. Smith. 2018. "Toward safety and security in FACE components: High assurance with portability." WWW.
URL: *http://mil-embedded.com/articles/toward-components-high-assurance-portability/*

Bujoreanu, I. N. 2011. "WHAT IF (Sensitivity Analysis)." *Journal of Defense Resources Management* 2(1):45–50.
URL: *http://journal.dresmara.ro/issues/volume2_issue1/05_bujoreanu.pdf*

Burns Statistics. 2018. "An Introduction to Genetic Algorithms." WWW.
URL: *https://www.burns-stat.com/documents/tutorials/an-introduction-to-genetic-algorithms/*

Burton, W. 2007. *Burton's Legal Thesaurus*. 4th ed. McGraw-Hill Professional.

Butler, L. 2016. "Legal implications of Brexit on UK Defence Procurement." WWW.
URL: *https://legalresearch.blogs.bris.ac.uk/2016/06/legal-implications-of-brexit-on-uk-defence-procurement/*

CAA. 2010. "Acceptable Means of Compliance to CAP670 SW01. Guidance for Producing SW01 Safety Arguments for COTS Equipment." WWW.
URL: *https://publicapps.caa.co.uk/docs/33/SW01COTSGuidanceIssue03.pdf*

CAA. 2018. "Advice to the aviation industry on a no deal EU exit." WWW.
URL: *https://www.caa.co.uk/Our-work/About-us/EU-exit/*

CAA. 2019*a*. "CAP670: Air Traffic Services Safety Requirements." WWW.
URL: *https://www.caa.co.uk/CAP670*

CAA. 2019*b*. "Our Role." WWW.
URL: *https://www.caa.co.uk/Our-work/About-us/Our-role/*

Cairney, P. 2017. "Evidence." WWW (Blog).
URL: *https://paulcairney.wordpress.com/evidence/*

Carlson, T. 2016. "FACE. Welcome and Overview." WWW.
URL: *https://slideplayer.com/slide/9782469/*

Carnegie, D. 1982. *How to Win Friends and Influence People*. Simon and Schuster.

Caseley, PR, N Tudor and C O'Halloran. 2003. "MOD Equipment Safety Assurance - The Case for an Evidence Based Approach to Software Certification." *Safety Standards Review Committee* 0:17.

CAST. 1998. "Certification Authorities Software Team (CAST) Position Paper (CAST-1) - Guidance for Assessing the Software Aspects of Product Service History of Airborne Systems and Equipment." WWW.
URL: *https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast_papers/media/cast-1.pdf*

Catch-22. 2018. "The Confidence Framework." WWW.
URL: *https://www.theconfidenceframework.org.uk/*

Cauvery, R., U. K. Sudha-Nayak, M. Girija and R. Meenakshi. 2003. *Research Methodology*. S. Chand Publishing.

Cerny, V. 1985. "Thermodynamical approach to the Traveling Salesman Problem: An efficient simulation algorithm." *Journal of Optimization Theory and Applications* 45:41–51.
URL: *https://link.springer.com/article/10.1007/BF00940812*

Chen, S., E. Nikolaidis and H. H. Cudney. 1999. Comparison of Probabilistic and Fuzzy Set Methods for Designing under Uncertainty. Technical Report AIAA-99-1579 American Institute of Aeronautics and Astronautics.
URL: *https://www.semanticscholar.org/paper/Comparison-of-Probabilistic-and-Fuzzy-Set-Methods-Chen-Nikolaidis/281ef57071405b2c37318007a2958b26e2f8de25*

Cherry, K. 2019. "How to Recognize and Avoid Groupthink." WWW.
URL: *https://www.verywellmind.com/what-is-groupthink-2795213*

Cho, J. and J. Kim. 2018. Performance Comparison of Heuristic Algorithms for UAV Deployment with Low Power Consumption. In *2018 International Conference on Information and Communication Technology Convergence (ICTC)*. pp. 1067–1069.
URL: *https://www.researchgate.net/publication/329490200_Performance_Comparison_of_Heuristic_Algorithms_for_UAV_Deployment_with_Low_Power_Consumption*

Choudhary, V. 2018. "Introduction to Greedy Algorithms." WWW.
URL: *https://developerinsider.co/introduction-to-greedy-algorithms/*

Churchman, C.W. 1959. *Measurement: Definitions and Theories*. Wiley.

Cleland, D. I. 2004. *Field Guide to Project Management*. 2 ed. Wiley.

Clements, P., R. Kazman and M. Klein. 2001. *Evaluating Software Architectures: Methods and Case Studies*. 1 ed. Addison Wesley.

Collins Dictionary. 1995*a*. "Compliance.".

Collins Dictionary. 1995*b*. "Confidence.".

Collins Dictionary. 1995*c*. "Confident.".

Collins Dictionary. 1995*d*. "Contribute.".

Collins Dictionary. 1995*e*. "Contribution.".

Collins Dictionary. 1995*f*. "Defensible.".

Collins Dictionary. 1995*g*. "Distinct.".

Collins Dictionary. 1995*h*. "Diverge.".

Collins Dictionary. 1995*i*. "Diverse.".

Collins Dictionary. 1995*j*. "Diversify.".

Collins Dictionary. 1995*k*. "Diversity.".

Collins Dictionary. 1995*l*. "Evidence.".

Collins Dictionary. 1995*m*. "Independence.".

Collins Dictionary. 1995*n*. "Independent.".

Collins Dictionary. 1995*o*. "Mutual.".

Collins Dictionary. 1995*p*. "Quality.".

Collins Dictionary. 1995*q*. "Relevant.".

Collins Dictionary. 1995*r*. "Sufficiency.".

Collins Dictionary. 1995*s*. "Sufficient.".

Collis, J. and R. Hussey. 2009. *Business Research: A Practical Guide for Undergraduate and Postgraduate Students*. 3 ed. Palgrave McMillan.

Colyvan, M. 2008. "Is Probability the Only Coherent Approach to Uncertainty." *Risk Analysis* 28(3):0.
URL: *http://www.colyvan.com/papers/ipocatu.pdf*

Compound Interest. 2015. "A Rough Guide to Types of Scientific Evidence." WWW.
URL: *http://www.compoundchem.com/2015/04/09/scientific-evidence/*

CPS. 2015*a*. "Domestic Abuse Charging Advice Sheet." WWW.
URL: *https://www.cps.gov.uk/sites/default/files/documents/publications/domestic_abuse_charging_advice_sheet_2015.pdf*

CPS. 2015*b*. "Expert Evidence - Legal Guidance." WWW.
URL: *https://www.cps.gov.uk/legal-guidance/expert-evidence*

CPS. 2018*a*. "About CPS." WWW.
URL: *https://www.cps.gov.uk/about-cps*

CPS. 2018*b*. "Bail." WWW.
URL: *https://www.cps.gov.uk/legal-guidance/bail*

CPS. 2018*c*. "The Full Code Test." WWW.
URL: *https://www.cps.gov.uk/publication/full-code-test*

Cronin, S. and B. Butka. 2018. Self-Optimizing Image Processing Algorithm for Safety Critical Systems. In *SoutheastCon 2018*. pp. 1–5.
URL: *https://ieeexplore.ieee.org/document/8479179*

Cupillari, A. 2012. *The Nuts and Bolts of Proofs: An Introduction to Mathematical Proofs*. 4th ed. Academic Press.

Cwik, C.H. and J.L. North. 2003. *Scientific Evidence Review: Admissibility of Expert Evidence (Issue 6)*. American Bar Association.

Cyra, L. and J. Gorski. 2008. Supporting Expert Assessment of Argument Structures in Trust Cases. In *The Proceedings of the 9th International Probabilistic Safety Assessment and Management Conference PSAM, Hong Kong, China.*
URL: *http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.163.1409&rep=rep1&type=pdf*

Dahll, Gustav. 2000. "Combining disparate sources of information in the safety assessment of software-based systems." *Nuclear Engineering and Design* 195(3):307 – 319.
URL: *http://www.sciencedirect.com/science/article/pii/S0029549399002137*

Daniels, D. 2018. COTS, legacy and reuse from a DO-178C/ED-12C and DO-278A/ED-109A perspective. In *SCSC Seminar: COTS, Legacy, and Reuse.*
URL: *https://scsc.uk/file/576/06---Dewi-Daniels---COTS-Legagy-and-Reuse-3.pptx*

Data Visualisation Catalogue. 2017. "Data Visualisation Catalogue." WWW.
URL: *https://datavizcatalogue.com/home_list.html*

Davis, M. 2018. "Addiction, Criminalization, and Character Evidence." *Texas Law Review* 96(3):619 – 653.
URL: *http://search.ebscohost.com/login.aspx?direct=true&db=plh&AN=127975877&site=ehost-live*

Delic, K., F. Mazzanti and L. Strigini. 1995. Formalising a Software Safety Case via Belief Networks. Technical report IEI-CNR, Pisa, Italy; Centre for Software Reliability, City University, London, U.K.
URL: *https://pdfs.semanticscholar.org/4237/1966b4f1dcea120df933189a978113b0adf4.pdf*

Denney, E., G. Pai and I. Habli. 2011. Towards Measurement of Confidence in Safety Cases. In *2011 International Symposium on Empirical Software Engineering and Measurement.* pp. 380–383.
URL: *https://ieeexplore.ieee.org/document/6092593/*

DE&S. 2017. "Defence Equipment & Support. The DE&S Way. Equipping and Supporting our Armed Forces. 2016/2017." WWW.
URL: *https://des.mod.uk/wp-content/uploads/2017/12/The_DES_Way.pdf*

DeWitt, R. 2018. *Worldviews - An Introduction to the History and Philosophy of Science.* 3 ed. John Wiley & Sons Ltd.

DfT. 2018. "Aviation safety if there's no Brexit deal." WWW.
URL: *https://www.gov.uk/government/publications/aviation-safety-if-theres-no-brexit-deal/aviation-safety-if-theres-no-brexit-deal*

DiFate, V. 2017. "Evidence." WWW - Encyclopedia of Philosophy.
URL: *https://www.iep.utm.edu/evidence/*

DMR. 2016*a*. "DSA02-DMR - MOD Shipping Regulation for Safety and Environmental Protection.".
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/766631/20181218-MOD_Shipping_Regulations_Archive_Version_PDF-O.pdf*

DMR. 2016*b*. "Naval Authority Notice (NAN) 02/2016, Software Integrity Policy.".

Dong, G., S. Wu, G. Wang, T. Guo and Y. Huang. 2010. Security Assurance with Metamorphic Testing and Genetic Algorithm. In *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*. Vol. 3 pp. 397–401.
URL: *https://ieeexplore.ieee.org/document/5614395*

Dong, Yuehua and Jidong Peng. 2011. Automatic generation of software test cases based on improved genetic algorithm. In *2011 International Conference on Multimedia Technology*. pp. 227–230.
URL: *https://ieeexplore.ieee.org/abstract/document/6002999*

Dror, I. and S. A. Cole. 2010. "The vision in 'blind' justice: Expert perception, judgment, and visual cognition in forensic pattern recognition." *Psychonomic bulletin & review* 17:161–7.
URL: *https://link.springer.com/article/10.3758/PBR.17.2.161*

Drown, D. J., T. M. Khoshgoftaar and N. Seliya. 2009. "Evolutionary Sampling and Software Quality Modeling of High-Assurance Systems." *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 39(5):1097–1107.
URL: *https://ieeexplore.ieee.org/document/4967988*

DSA. 2018*a*. "About Us." WWW.
URL: *https://www.gov.uk/government/organisations/defence-safety-authority/about*

DSA. 2018*b*. "Defence Safety Authority." WWW.
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/517494/20151020-DSAuthority-Amendments_v1_2.pdf*

DSIWG. 2018. *Data Safety Guidance*. 3.0 ed. SCSC.
URL: *https://scsc.uk/r127C:1*

Duan, Lian, Sanjai Rayadurgam, Mats P E Heimdahl, Oleg Sokolsky and Insup Lee. 2015. Representing confidence in assurance case evidence. In *Computer Safety, Reliability, and Security - AFECOMP 2015 Workshops ASSURE, DECSoS, ISSE, ReSA4CI, and SASSUR, Proceedings*. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) Springer- Verlag pp. 15–26.
URL: *https://www.researchgate.net/publication/300145635_Representing_Confidence_in_Assurance_Case_Evidence*

Dupay, A. and N. Leveson. 2000. An Empirical Evaluation of the MC/DC Coverage Criterion on the HETE-2 Satellite Software. In *Digital Aviation Systems Conference*.
URL: *http://sunnyday.mit.edu/papers/dupuy.pdf*

EASA. 2008. "EU 482/2008: Establishing a Software Safety Assurance System to be Implemented by Air Navigation Service Providers.".
URL: *https://op.europa.eu/en/publication-detail/-/publication/5806e462-0226-4a4f-93b8-a13f89e0b5e8/language-en*

EASA. 2012. "EASA CM - SWCEH - 002. Software Aspects of Certification.".
URL: *https://www.easa.europa.eu/document-library/product-certification-consultations/easa-cm-swceh-002*

EASA. 2017*a*. Notice of Proposed Amendment 2017-10. Software assurance level requirements for safety assessment of changes to air traffic management/air navigation services functional systems. Technical report EASA.
URL: *https://www.easa.europa.eu/document-library/notices-of-proposed-amendment/npa-2017-10*

EASA. 2017*b*. "Regulation (EU) 2017 373 - Air Traffic Management Common Requirements Implementing Regulation (ATM IR).".
URL: *https://www.easa.europa.eu/document-library/regulations/commission-implementing-regulation-eu-2017373*

EDA MAWA Forum. 2014. EMAR 147 Aircraft Maintenance Training Organisations. Technical Report 1 EDA.
URL: *https://www.eda.europa.eu/docs/default-source/documents/emar-147-edition-1-1-(23-sep-2014)approved-(forms-removed).pdf*

EDA MAWA Forum. 2015. EMAR Continuing Airworthiness Requirements. Technical Report 1 EDA.
URL: *https://www.eda.europa.eu/docs/default-source/documents/emar-m-edition-1-0-(12-oct-2015)---approved.pdf*

EDA MAWA Forum. 2016*a*. EMAD Recognition Process. Technical Report 1 EDA.
URL: *https://www.eda.europa.eu/docs/default-source/documents/emad-r-edition-2-0-(3-feb-2016)---approved.pdf*

EDA MAWA Forum. 2016*b*. EMAR MFTP Military Flight Test Permit Procedure. Technical Report 1 EDA.
URL: *https://www.eda.europa.eu/docs/default-source/documents/emad-mftp-edition-1-0-(4-oct-2016)-approved.pdf*

EDA MAWA Forum. 2018. "Approved MAWA Documents." WWW.
URL: *https://www.eda.europa.eu/experts/airworthiness/mawa-documents*

Education Endowment Foundation. 2018. "Teaching & Learning Toolkit." WWW.
URL: *https://educationendowmentfoundation.org.uk/evidence-summaries/teaching-learning-toolkit*

Eiben, A.E. and J. E. Smith. 2003. *Introduction to Evolutionary Computing*. Spinger.

Evergreen, S.D.H. 2017. *Effective Data Visualization: The Right Chart for the Right Data*. SAGE Publications.

FAA. 2000. FAA System Safety Handbook, Chapter 3: Principles of System Safety. Technical report FAA.
URL: *https://www.faa.gov/regulations_policies/handbooks_manuals/aviation/risk_management/ss_handbook/*

FAA. 2017. Assurance of Multicore Processors in Airborne Systems. Technical Report DOT/FAA/TC-16/51 FAA.
URL: *https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/media/TC-16-51.pdf*

FAA. 2018. "Technical Standard Orders (TSO)." WWW.
URL: *https://www.faa.gov/aircraft/air_cert/design_approvals/tso/*

Fenn, J. L., R. D. Hawkins, P. J. Williams, T. P. Kelly, M.G. Banner and Y. Oakshott. 2007. "The Who, Where, How, Why and When of Modular and Incremental Certification.".
URL: *https://ieeexplore.ieee.org/document/4399923*

Fenton, N., B. Littlewood, M. Neil, L. Strigini, A. Sutcliffe and D. Wright. 1998. "Assessing dependability of safety critical systems using diverse evidence." *Software, IEE Proceedings* - 145(1):35–39.
URL: *https://ieeexplore.ieee.org/iel4/5658/15162/00689297.pdf*

Fenton, N and M Neil. 2001. "Making decisions: using Bayesian nets and MCDA." *Knowledge-Based Systems* 14(7):307 – 325.
URL: *http://www.sciencedirect.com/science/article/pii/S095070510000071X*

Firesmith, D. 2015. "SEI Blog - A Taxonomy of Testing: What-Based and When-Based Testing Types." WWW.
URL: *https://insights.sei.cmu.edu/sei_blog/2015/09/a-taxonomy-of-testing-what-based-and-when-based-testing-types.html*

FLOC. 2018. "Summit on Machine Learning Meets Formal Methods (13th July 2018)." WWW.
URL: *https://www.floc2018.org/summit-on-machine-learning/*

Fulton, R. 2017. "RTCA DO-254 Training.".

Functionize. 2018. "Types of Software Testing." WWW.
URL: *https://www.functionize.com/blog/types-of-software-testing/*

Gallagher, S. 2015. "Airbus confirms software configuration error caused plane crash." WWW.
URL: *https://arstechnica.com/information-technology/2015/06/airbus-confirms-software-configuration-error-caused-plane-crash/*

Galletta, A. 2013. *Mastering the Semi-Structured Interview and Beyond: From Research Design to Analysis and Publication*. NYU Press.

Garfinkel, S. 2005. "History's Worst Software Bugs." WWW.
URL: *https://www.wired.com/2005/11/historys-worst-software-bugs/?currentPage=all*

Geramian, A., M. R. Mehregan, N. G. Mokhtarzadeh and M. Hemmati. 2017. Fuzzy inference system application for failure analyzing in automobile industry. In *International Journal of Quality & Reliability Management*. Vol. 34 pp. 1493–1507.
URL: *https://www.researchgate.net/publication/319862913_Fuzzy_inference_system_application_for_failure_analyzing_in_automobile_industry*

Gerrard, P. and N. Thompson. 2002. *Risk-based E-business Testing*. Artech House.

Ghanbari, H. 2016. Seeking Technical Debt in Critical Software Development Projects: An Exploratory Field Study. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*. pp. 5407–5416.
URL: *https://ieeexplore.ieee.org/abstract/document/7427856*

Gharehchopogh, F. S., R. Rezaii and B. Arasteh. 2015. A new approach by using Tabu search and genetic algorithms in Software Cost estimation. In *2015 9th International Conference on Application of Information and Communication Technologies (AICT)*. pp. 113–117.
URL: *https://www.researchgate.net/publication/308820357_A_new_approach_by_using_Tabu_search_and_genetic_algorithms_in_Software_Cost_estimation*

Gibbs, S. 2015. "Airbus issues software bug alert after fatal plane crash." WWW.
URL: *https://www.theguardian.com/technology/2015/may/20/airbus-issues-alert-software-bug-fatal-plane-crash*

Gill, J. and P. Johnson. 2014. *Research Methods for Managers*. 4 ed. SAGE.

Given, L. M. 2008. *The Sage Encyclopedia of Qualitative Research Methods*. SAGE.

GO-Science. 2010. The Government Chief Scientific Advisers Guidelines on the Use of Scientific and Engineering Advice in Policy Making. Technical report Department for BIS.
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/293037/10-669-gcsa-guidelines-scientific-engineering-advice-policy-making.pdf*

Graydon, P. and C. M. Holloway. 2016. An Investigation of Proposed Techniques for Quantifying Confidence in Assurance Arguments. Technical Report NASA/TM–2016–219195 NASA.
URL: *https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20160006526.pdf*

Griffin, R. 2007. *Fundamentals of Management*. 5 ed. Cengage Learning.

Grigorova, S. and T. S. E. Maibaum. 2013. Taking a page from the law books: Considering evidence weight in evaluating assurance case confidence. In *2013 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*. pp. 387–390.
URL: *https://ieeexplore.ieee.org/document/6688926/*

Guh, Yuh-Yuan, Rung-Wei Po and E. Stanley Lee. 2008. "The fuzzy weighted average within a generalized means function." *Computers & Mathematics with Applications* 55(12):2699 – 2706.
URL: *http://www.sciencedirect.com/science/article/pii/S0898122107007699*

Guiochet, Jérémie, Quynh Anh Do Hoang and Mohamed Kaâniche. 2015. "A Model for Safety Case Confidence Assessment." *CoRR* abs/1512.04467:0.
URL: *http://arxiv.org/abs/1512.04467*

Guo, B. 2003. Knowledge representation and uncertainty management: applying Bayesian belief networks to a safety assessment expert system. In *International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003.* pp. 114–119.
URL: *https://ieeexplore.ieee.org/document/1275879*

Haasan, R., B. Cohanim and O. De Week. 2005. A Comparison of Particle Swarm Optimisation and the Genetic Algorithm. Technical report MIT.
URL: *https://www.researchgate.net/publication/309901048_A_comparison_of_particle_swarm_optimization_and_the_genetic_algorithm*

Haddon-Cave, C. 2009. The Nimrod Review. An independent review into the broader issues surrounding the loss of the RAF Nimrod MR2 Aircraft XV230 in Afghanistan in 2006. Technical report Ordered by the House of Commons.
URL: *https://www.gov.uk/government/publications/the-nimrod-review*

Hadley, M. J. 2013. Empirical Evaluation of the Effectiveness and Reliability of Software Testing Adequacy Criteria and Reference Test Systems PhD thesis University of York.
URL: *http://etheses.whiterose.ac.uk/5861/*

Hadley, M. J. and M. Standish. 2017. "R-Cloud Invitation to Tender (ITT): Ada Security Vulnerabilities Assessment Tool and Process Development (R-Cloud Task Dstlx-1000119542)." WWW (R-Cloud).

Hadley, M. J. and T. White. 2008. Review of the [Airborne Platform] Software Safety Measures (Volume I - Findings and Recommendations). Issue 1. Technical Report Dstl/CR26280 Dstl.

Hall, J. 2018. Safety engineering with COTS components: a problem-oriented approach. In *SCSC Seminar: COTS, Legacy, and Reuse.*
URL: *https://scsc.uk/e576*

Hall, Patrick, John May, D Nichol, K Czachur and B Kinch. 1992. "Integrity Prediction During Software Development." *IFAC Proceedings Volumes* 25:239–244.
URL: *https://www.researchgate.net/publication/317762790_Integrity_Prediction_During_Software_Development*

Hannibal, M and L Mountford. 2016. *Criminal Litigation 2016-2017*. Oxford University Press.

Hawkins, R. D. and T. P. Kelly. 2009. Software safety assurance - what is sufficient? In *4th IET International Conference on Systems Safety 2009. Incorporating the SaRS Annual Conference*. pp. 1–6.
URL: *https://ieeexplore.ieee.org/document/5513089/*

Hawkins, R., K. Clegg, R. Alexander and T. Kelly. 2011. Using a Software Safety Argument Pattern Catalogue: Two Case Studies. In *Computer Safety, Reliability, and Security. 30th International Conference, SAFECOMP 2011, Naples, Italy, September 19-22, 2011, Proceedings*.
URL: *https://www.researchgate.net/publication/221147542_Using_a_Software_Safety_Argument_Pattern_Catalogue_Two_Case_Studies*

Hawkins, R. and T. Kelly. 2010. A structured approach to selecting and justifying software safety evidence. In *5th IET International Conference on System Safety 2010*. pp. 1–6.
URL: *https://ieeexplore.ieee.org/document/5712329/*

Heale, R. and A. Twycross. 2015. "Validity and reliability in quantitative research." *Evidence-Based Nursing* 18(3):66–67.
URL: *https://ebn.bmj.com/content/18/3/66*

Helm, R. and R. Johnson. 2015. *Design Patterns: Elements of Reusable Object-Oriented Software*. Pearson Education.

HM Government. 2012. The Civil Service Reform Plan. Technical report HM Government.
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/305148/Civil-Service-Reform-Plan-final.pdf*

HM Government. 2013. Policy Skills and Knowledge Framework. Technical report HM Government.
URL: *https://civilservicelearning.civilservice.gov.uk/sites/default/files/final_policy_skills_knowledge_framework_february_2013.pdf*

Hobbs, C. and M. Lloyd. 2012. *The Application of Bayesian Belief Networks to Assurance Case Preparation*. Springer London chapter 12, p. 0.
URL: *https://link.springer.com/chapter/10.1007/978-1-4471-2494-8_12*

Hoffmann, T., S. Bennett and C. Del Mar. 2013. *Evidence-Based Practice Across the Health Professions*. Elsevier Health Sciences.

Holloway, C. M. and P. Graydon. 2018. Explicate '78: Assurance Case Applicability to Digital Systems. Technical report FAA.
URL: *https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/media/TC-17-67.pdf*

Holsapple, C.W. and A.B. Whinston. 1996. *Decision Support Systems: A Knowledge Based Approach*. 10 ed. West Group.

Holt, J. 2007. "The brontosaurus of complexity - teaching systems engineering." IET Seminar on Model Based Systems.
URL: *https://ieeexplore.ieee.org/document/4300933*

Howick, J. 2013. *The Philosophy of Evidence-Based Medicine*. Wiley-Blackwell & BMJ Books.

Howick, J., I. Chalmers, P. Glasziou, T. Greenhalgh, C. Heneghan, A. Liberati, I. Moschetti, B. Phillips and H. Thornton. 2011. "The 2011 Oxford CEBM Evidence Levels of Evidence (Introductory Document)." WWW.
URL: *http://www.cebm.net/index.aspx?o=5653*

Hristakeva, M. and D. Shrestha. 2005. Different Approaches to Solve the 0/1 Knapsack Problem. In *MICS*.
URL: *http://www.micsymposium.org/mics_2005/papers/paper102.pdf*

HSE. 2018*a*. "ALARP "at a glance"." WWW.
URL: *http://www.hse.gov.uk/risk/theory/alarpglance.htm*

HSE. 2018*b*. "Flixborough (Nypro UK) Explosion 1st June 1974." WWW.
URL: *http://www.hse.gov.uk/comah/sragtech/caseflixboroug74.htm*

Hsu, C. and C. Huang. 2010. A Study on the Applicability of Modified Genetic Algorithms for the Parameter Estimation of Software Reliability Modeling. In *2010 IEEE 34th Annual Computer Software and Applications Conference*. pp. 531–540.
URL: *https://ieeexplore.ieee.org/document/5676305*

Huff, L. and G. Novak. 2007. "Performance-Based Software Sustainment for the F-35 Lightning II." CrossTalk - The Journal of Defense Software Engineering. 20 (12) 9-14.
URL: *https://apps.dtic.mil/sti/citations/ADA487067*

IAWG. 2010. "Industrial Avionics Working Group Modular Software Safety Case Process (Suite of Technical Reports).".
URL: *https://www.amsderisc.com/resources/www.capability-agility.co.uk/capability-agility/work-package-3/Modular%20Software%20Safety%20Case%20Process%20Overview.pdf*

Ibanez, O., J. Santos and N. Berreira. 2006. Topological Active Nets Optimization Using Genetic Algorithms. In *Image Analysis and Recognition: Third International Conference, ICIAR 2006*.
URL: *https://link.springer.com/chapter/10.1007/11867586_26*

IDC in Systems. 2013. "Industrial Doctorate Centre in Systems- The University of Bristol/University of Bath. Engineering Doctorate (EngD) Programme in Systems. Handbook 2013-2014." WWW.
URL: *http://www.bristol.ac.uk/media-library/sites/eng-systems-centre/migrated/documents/handbook-2013.pdf*

IEEE. 2018. "Software Engineering Competency Model (SWECOM)." WWW.
URL: *https://www.computer.org/web/peb/swecom*

Imwinkelried, E.J. 2014. *The Methods of Attacking Scientific Evidence*. LexisNexis.

Inge, J. R. 2007. "The Safety Case, its Development and Use in the United Kingdom.".
URL: *http://safety.inge.org.uk/20070625-Inge2007_The_Safety_Case-U.pdf*

International Nuclear Regulators. 2018. Licensing of safety critical software for nuclear reactors. Common position of international nuclear regulators and authorised technical support organisations. Technical Report 2018 Regulator Task Force on Safety Critical Software (TF SCS).
URL: *http://www.onr.org.uk/software.pdf*

Jaiswal, K., A. Al-Mahadin, S. Verma and B. Singh. 2018. Safety culture in aircraft maintenance organizations of United Arab Emirates. In *2018 Advances in Science and Engineering Technology International Conferences (ASET)*. pp. 1–8.
URL: *https://www.researchgate.net/publication/325915468_Safety_culture_in_aircraft_maintenance_organizations_of_United_Arab_Emirates*

Johnson, Jeffrey and Susan Weller. 2002. "Elicitation Techniques for interviewing." *Handbook of Interview Research* 1:1.
URL: *https://www.researchgate.net/publication/239964620_Elicitation_Techniques_for_interviewing*

Kaewyotha, J. and W. Songpan. 2018. A Study on the Optimization Algorithm for Solving the Supermarket Shopping Path Problem. In *2018 3rd International Conference on Computer and Communication Systems (ICCCS)*. pp. 11–15.
URL: *https://www.researchgate.net/publication/327641851_A_Study_on_the_Optimization_Algorithm_for_Solving_the_Supermarket_Shopping_Path_Problem*

Kaisti, M., V. Rantala and T. Mujunen. 2013. "Agile methods for embedded systems development - a literature review and a mapping study." *EURASIP Journal on Embedded Systems* 2013:1–16.
URL: *https://link.springer.com/article/10.1186/1687-3963-2013-15*

Kaner, C., J. Falk and H. Q. Nguyen. 1999. *Testing Computer Software*. 2 ed. Wiley.

Kasauli, Rashidah, Eric Knauss, Benjamin Kanagwa, Agneta Nilsson and Gul Calikli. 2018. Safety-Critical Systems and Agile Development: A Mapping Study. In *44th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. pp. 470–477.
URL: *https://www.researchgate.net/publication/328458183_Safety-Critical_Systems_and_Agile_Development_A_Mapping_Study*

Kelly, T. 2011. "Introduction to Safety Cases." WWW.
URL: *https://warwick.ac.uk/fac/med/staff/sujan/research/safety_case_review/wp3_workshop/kelly_scr.pdf*

Khan, R. and M. Amjad. 2016. Optimize the software testing efficiency using genetic algorithm and mutation analysis. In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*. pp. 1174–1176.
URL: *https://ieeexplore.ieee.org/document/7724450*

Kharchenko, V. 2016. Diversity for safety and security of embedded and cyber physical systems: Fundamentals review and industrial cases. In *2016 15th Biennial Baltic Electronics Conference (BEC)*. pp. 17–26.
URL: *https://ieeexplore.ieee.org/document/7743719/*

Kharchenko, V. and E. Brezhnev. 2015. Diversity for Safety of Systems and Software in Context of the Standard ISO/IEC26262. In *13th WS on Automotive Software and Systems*.
URL: *http://www.automotive-spin.it/uploads/13/13W_Kharchenko.pdf*

Kim, J. 1988. "What is Naturalized Epistemology?" Philosophical Perspectives 2, Epistemology.
URL: *https://philpapers.org/rec/KIMWIN*

Kinni, T. 2017. "The Critical Difference Between Complex and Complicated." WWW.
URL: *https://sloanreview.mit.edu/article/the-critical-difference-between-complex-and-complicated/*

Kirk, A. 2016. *Data Visualisation: A Handbook for Data Driven Design*. SAGE.

Kirkpatrick, S., C.D. Gelatt and M. Vacchi. 1983. "Optimization by simulated annealing." *Science* 220:498–516.
URL: *https://www.researchgate.net/publication/220118677_Optimization_by_Simulated_Annealing*

Klir, G. 2005. *Uncertainty and Information: Foundations of Generalized Information Theory*. John Wiley & Sons.

Klir, G. and B. Yuan. 1995. *Fuzzy Sets and Fuzzy Logic - Theory and Applications*. Prentice Hall Press.

Kozhakhmet, K., G. Bortsova, A. Inoue and L. Atymtayeva. 2012. Expert System for Security Audit Using Fuzzy Logic. In *Proceedings of the 23rd Midwest Artificial Intelligence and Cognitive Science Conference 2012 (MAICS 2012)*, ed. S. Visa, A. Inoue and A. Ralescu. Vol. 841.
URL: *http://ceur-ws.org/Vol-841/submission_35.pdf*

Kramer, R. M. and M. A. Neale. 1998. *Power and Influence in Organizations*. SAGE.

Kritzinger, D. 2017. The Pros and Cons of the DAOS Template. Technical report Baines Simmons.
URL: *https://www.bainessimmons.com/wp-content/uploads/Pros-Cons-DAOS.pdf*

Ledinot, Emmanuel, Jean-Paul Blanquart, Jean Gassino, Bertrand Ricque, Philippe Baufreton, J. Boulanger, Jean-Louis Camus, Cyrille Comar, Hervé Delseny and Philippe Quéré. 2016. Perspectives on Probabilistic Assessment of Systems and Software. In *8th European Congress on Embedded Real Time Software and Systems (ERTS 2016), Jan 2016, TOULOUSE, France*.
URL: *https://www.semanticscholar.org/paper/Perspectives-on-Probabilistic-Assessment-of-Systems-Ledinot-Blanquart/8c254fe747356d288984ba0972b1baab53c13ff6*

Lee, Kwang Y. and Mohamed A. El-Sharkawi. 2008. *Fundamentals of Simulated Annealing.* IEEE chapter 0, p. 0.
URL: *https://ieeexplore.ieee.org/document/5396790*

Lehman, J. and S. Phelps. 2005. *West's Encyclopedia of American Law.* 2nd ed. Thomson/Gale.

Lennon, E., M. Standish and M. J. Hadley. 2018. Interim Report on NIST Risk Management Framework Mapping to RTCA DO-326A. Technical Report TR108269 v1.1 Dstl.

Leung, L. 2015. "Validity, reliability, and generalizability in qualitative research." *Journal of Family Medicine and Primary Care* 4:324–327.
URL: *https://www.researchgate.net/publication/281172234_Validity_reliability_and_generalizability_in_qualitative_research*

Leveson, N. 1995. *Safeware: System Safety and Computers.* Addison-Wesley.

Leveson, N. 2011. "White Paper on the Use of Safety Cases in Certification and Regulation.".
URL: *http://sunnyday.mit.edu/SafetyCases.pdf*

Li, Y. F., M. Xie and T. N. Goh. 2007. A study of genetic algorithm for project selection for analogy based software cost estimation. In *2007 IEEE International Conference on Industrial Engineering and Engineering Management.* pp. 1256–1260.
URL: *https://ieeexplore.ieee.org/document/4419393*

Lijuan, W., Z. Yue and H. Hongfeng. 2012. Genetic Algorithms and Its Application in Software Test Data Generation. In *2012 International Conference on Computer Science and Electronics Engineering.* Vol. 2 pp. 617–620.
URL: *https://www.researchgate.net/publication/254029445_Genetic_Algorithms_and_Its_Application_in_Software_Test_Data_Generation*

Littlewood, B., P. Popov and L. Strigini. 1999. "A note on reliability estimation of functionally diverse systems." *Reliability Engineering & System Safety* 66(1):93 – 95.
URL: *http://www.sciencedirect.com/science/article/pii/S0951832099000149*

Littlewood, B., P. T. Popov, L. Strigini and N. Shryane. 2000. "Modeling the effects of combining diverse software fault detection techniques." *IEEE Transactions on Software Engineering* 26(12):1157–1167.
URL: *https://ieeexplore.ieee.org/abstract/document/888629/*

Littlewood, Bev. 2000. "The use of proof in diversity arguments." *Software Engineering, IEEE Transactions on* 26(10):1022–1023.
URL: *https://dl.acm.org/doi/abs/10.1109/32.879822*

Littlewood, Bev and D. Wright. 2007. "The Use of Multilegged Arguments to Increase Confidence in Safety Claims for Software-Based Systems: A Study Based on a BBN Analysis of an Idealized Example." *Software Engineering, IEEE Transactions on* 33(5):347–365.
URL: *https://ieeexplore.ieee.org/document/4160972*

Littlewood, Bev, Peter Popov and Lorenzo Strigini. 2001. "Modeling Software Design Diversity: A Review." *ACM Comput. Surv.* 33(2):177–208.
URL: *http://doi.acm.org/10.1145/384192.384195*

Liu, Jonhson, J.B. Yang, Jchwang Wang and Slive Sii. 2003. "Review of Uncertainty Reasoning Approaches as Guidance for Maritime and Offshore Safety-Based Assessment:." *Journal of UK Safety and Reliability Society* 23:0.
URL: *https://www.researchgate.net/publication/228864767_Review_of_Uncertainty_Reasoning_Approaches_as_Guidance_for_Maritime_and_Offshore_Safety-Based_Assessment*

LSSR. 2017. "DSA02.DLSR.LSSR Land System Safety and Environmental Protection Directive.".
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/646336/DSA02-Regulations-DLSR-LSSR.pdf*

Luping, Chen and J. May. 2014. A Diversity Model Based on Failure Distribution and Its Application in Safety Cases. In *Software Security and Reliability (SERE), 2014 Eighth International Conference on*. pp. 1–10.
URL: *https://ieeexplore.ieee.org/document/7358170*

Luping, Chen, J. May and G. Hughes. 2001. Estimation of software diversity by fault simulation and failure searching. In *Software Reliability Engineering, 2001. ISSRE 2001. Proceedings. 12th International Symposium on*. pp. 122–131.
URL: *https://dl.acm.org/doi/10.5555/851028.856254*

MAA. 2013. "UK MAA and US Army Airworthiness Authority Sign Mutual Recognition Certificate." WWW.
URL: *http://maa.tools.mod.uk/linkedfiles/20140512-maa_mutual_recognition_amrdec.pdf*

MAA. 2015. "DEF STAN 00-970 NOTICE OF PROPOSED AMENDMENT (Def Stan 00-970-NPA)." WWW.
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/454732/Def_Stan_00-970_NAA_2015-002.pdf*

MAA. 2016*a*. "RA 1220: project team airworthiness and safety." WWW.
URL: *https://www.gov.uk/government/publications/regulatory-article-ra-1220-project-team-airworthiness-and-safety*

MAA. 2016*b*. "RA 1300: release to service (RTS)." WWW.
URL: *https://www.gov.uk/government/publications/regulatory-article-ra-1300-release-to-service-rts*

MAA. 2017*a*. "MAA Recognition." WWW.
URL: *https://www.gov.uk/government/publications/maa-recognition*

MAA. 2017*b*. "MAA01: Military Aviation Authority Regulatory Policy." WWW.
URL: *https://www.gov.uk/government/publications/maa01-military-aviation-authority-maa-regulatory-policy*

MAA. 2017*c*. "RA 1205: air system safety cases." WWW.
URL: *https://www.gov.uk/government/publications/regulatory-article-ra-1205-air-system-safety-cases*

MAA. 2017*d*. "RA 5810 - Military Type Certificate (MRP 21 Subpart B)." WWW.
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/712833/RA5810_Issue_2.pdf*

MAA. 2017*e*. "RA 5820: Changes in Type Design (MRP 21 Subpart D)." WWW.
URL: *https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/712835/RA5820_Issue_2.pdf*

MAA. 2018*a*. "List of MAA approved organizations." WWW.
URL: *https://www.gov.uk/government/publications/list-of-maa-approved-organisations*

MAA. 2018*b*. "Military Aviation Authority certification." WWW.
URL: *https://www.gov.uk/government/collections/military-aviation-authority-certification*

MAA. 2018*c*. "RA 5850: Military Design Approved Organization (MRP 21 Subpart J)." WWW.
URL: *https://www.gov.uk/government/publications/regulatory-article-ra-5850-military-design-approved-organization-mrp-21-subpart-j*

MAA. 2018*d*. "Regulatory Article (RA) 1005: contracting with competent organisations." WWW.
URL: *https://www.gov.uk/government/publications/regulatory-article-ra-1005-competent-organisations-and-responsibilities*

Magoun, A.B. and P. Isreal. 2013. "Did You Know? Edison Coined the Term 'Bug'." WWW.
URL: *http://theinstitute.ieee.org/tech-history/technology-history/did-you-know-edison-coined-the-term-bug*

Marcil, L. 2012. Realizing DO-178C's Value by Using New Technology: OOT, MBDV, TQC & FM. In *31st Digital Avionics Systems Conference (DASC)*.
URL: *https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6383059*

MathWorks. 2014*a*. "Simulink." WWW.
URL: *https://www.mathworks.com/products/simulink.html*

MathWorks. 2014*b*. "Stateflow." WWW.
URL: *https://www.mathworks.com/products/stateflow.html*

MathWorks. 2018*a*. "Comparison of Sugeno and Mamdani Systems." WWW.
URL: *https://uk.mathworks.com/help/fuzzy/comparison-of-sugeno-and-mamdani-systems.html*

MathWorks. 2018*b*. "Foundations of Fuzzy Logic." WWW.
URL: *https://uk.mathworks.com/help/fuzzy/foundations-of-fuzzy-logic.html*

McDermid, J. 1998. A Review of Safety Related Defence Standards. Technical report University of York.

McDermid, J. A. and P. Williams. 2014. Defence standard 00-56 issue 5: concepts, principles and pragmatics. In *9th IET International Conference on System Safety and Cyber Security (2014)*. pp. 1–6.
URL: *https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7111720&tag=1*

Meacham, C. J. G. 2015. "Understanding Conditionalization." *Canadian Journal of Philosophy* 45:767–797.
URL: *http://dx.doi.org/10.1080/00455091.2015.1119611*

Mendelow, A. L. 1981. Environmental Scanning - The Impact of the Stakeholder Concept. In *ICIS 1981 Proceedings*.
URL: *https://aisel.aisnet.org/icis1981/20/*

Menon, C. 2018. COTS, Safety and Customisable Software. In *SCSC Seminar: COTS, Legacy, and Reuse*.
URL: *https://scsc.uk/file/576/05---Catherine-Menon---SCSC_Menon_v4.pptx*

Menon, C., R. Hawkins and J. McDermid. 2009*a*. "Defence Standard 00-56 Issue 4: Towards Evidence-Based Safety Standards.".
URL: *https://link.springer.com/chapter/10.1007%2F978-1-84882-349-5_15*

Menon, C., R. Hawkins and J. McDermid. 2009*b*. Interim Standard of Best Practice on Software in the Context of DS 00-56 Issue 4. Technical report Software Systems Engineering Initiative.
URL: *https://docplayer.net/19286892-Software-systems-engineering-initiative.html*

Merriam-Webster. 2018. "Legacy." WWW.
URL: *https://www.merriam-webster.com/dictionary/legacy*

Mertens, B.G.M. 2004. Reasoning with uncertainty in the situational awareness of air targets. Technical report Delft University of Technology.

Mingers, John. 2011. "Soft OR Comes of Age - But Not Everywhere!" *Omega* 39:729–741.
URL: *https://www.sciencedirect.com/science/article/pii/S0305048311000089*

MISRA. 2012. "MISRA C: 2012 Guidelines for the use of the C language in critical systems.".
URL: *https://www.misra.org.uk/MISRAHome/MISRAC2012/tabid/196/Default.aspx*

Mitchell, M. 1996. *An Introduction to Genetic Algorithms*. MIT Press.

Mitrea, D. 2011. *Software Testing*. Vibrant Publishers.

Musa, J. 1987. *Software Reliability: Measurement, Prediction, Application*. 1 ed. McGraw-Hill.

Myers, G. 2004. *The Art of Software Testing*. 2 ed. John Wiley & Sons Ltd.

NAFEMS. 2018. "What is SQEP." WWW.
URL: *https://www.nafems.org/downloads/resource_center/WT03.pdf/*

Nair, S, N Walkinshaw and T Kelly. 2014. Quantifying Uncertainty in Safety Cases Using Evidential Reasoning. Report Simula Research Laboratory (Norway); Department of Computer Science, University of Leicester, (UK); Department of Computer Science, University of York (UK).
URL: *https://link.springer.com/chapter/10.1007%2F978-3-319-10557-4_45*

Nair, S., N. Walkinshaw, T. Kelly and J. L. de la Vara. 2015. An evidential reasoning approach for assessing confidence in safety evidence. In *2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE)*. pp. 541–552.
URL: *https://ieeexplore.ieee.org/document/7381846/*

NAO. 2015. "A Short Guide to the Ministry of Defence." WWW.
URL: *https://www.nao.org.uk/wp-content/uploads/2015/08/A-Short-Guide-to-the-Ministry-of-Defence1.pdf*

Naseem, Afshan, Syed Tasweer Hussain Shah, Shoab Ahmed Khan and Asad Waqar Malik. 2017. "Decision support system for optimum decision making process in threat evaluation and weapon assignment: Current status, challenges and future directions." *Annual Reviews in Control* 43:169 – 187.
URL: *http://www.sciencedirect.com/science/article/pii/S1367578816300979*

Neil, M. and N. Fenton. 1996. Predicting Software Quality using Bayesian Belief Networks. In *21st Annual Software Engineering Workshop NASA/Goddard Space Flight Centre,*.
URL: *http://www.eecs.qmul.ac.uk/~norman/papers/sel_defects.pdf*

NICE. 2017*a*. "Changes to NICE drug appraisals: what you need to know." WWW.
URL: *https://www.nice.org.uk/news/feature/changes-to-nice-drug-appraisals-what-you-need-to-know*

NICE. 2017*b*. "NICE Evidence search: evidence type filter." WWW.
URL: *https://www.nice.org.uk/Media/Default/About/NICE-Communities/Library-and-knowledge-services-staff/Training-materials/Evidence-type-definitions-Jul17.docx*

NIST. 2013. "NIST Special Publication 800-53. Security and Privacy Controls for Federal Information Systems and Organizations.".
URL: *https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-53r4.pdf*

NIST. 2018. "NIST Special Publication 800-37. Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy.".
URL: *https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r2.pdf*

NRC. 2011. *Reference Manual on Scientific Evidence (Third Edition)*. National Academies Press.

Nussbaumer-Knaflic, C. 2015. *Storytelling with Data: A Data Visualization Guide for Business Professionals*. John Wiley & Sons Ltd.

Nutley, S., A. Powell and H. Davies. 2013. What Counts as Good Evidence? Technical report Alliance for Useful Evidence.
URL: *https:// www.alliance4usefulevidence.org/ assets/ What-Counts-as-Good-Evidence-WEB.pdf*

OED. 2018*a*. "Assurance." WWW.
URL: *https:// en.oxforddictionaries.com/ definition/ assurance*

OED. 2018*b*. "Dispose." WWW.
URL: *https:// en.oxforddictionaries.com/ definition/ dispose*

OED. 2018*c*. "Enhance." WWW.
URL: *https:// en.oxforddictionaries.com/ definition/ enhance*

OED. 2018*d*. "Evidence." WWW.
URL: *https:// en.oxforddictionaries.com/ definition/ evidence*

OED. 2018*e*. "Fail-Safe." WWW.
URL: *https:// en.oxforddictionaries.com/ definition/ fail-safe*

Ogilvie, D., M. Egan, V. Hamilton and M. Petticrew. 2005. "Sytematic review of health effects of social interventions - Best available evidence: how low should you go?" Journal of Epidemiology and Community Health, 59: 886-892.

ONR. 2016. The Purpose, Scope, and Content of Safety Cases. Technical Report NS-TAST-GD-051 Revision 4 ONR.
URL: *http:// www.onr.org.uk/ operational/ tech_asst_guides/ ns-tast-gd-051.pdf*

ONR. 2017. ONR Guide. Computer Based Safety Systems. Nuclear Safety Technical Assessment Guide NS-TAST-GD-046 Revision 4 ONR.
URL: *http:// www.onr.org.uk/ operational/ tech_asst_guides/ ns-tast-gd-046.pdf*

Panjwani, A. 2017. Evidence: Improving the use of evidence in UK government policymaking. Technical report Campaign for Science and Engineering (CaSE).
URL: *http:// www.sciencecampaign.org.uk/ asset/ 016176D1-09BF-4CD9-BB9C27B2D7BC50B4.62F554EC-54A4-430E-8EE849F46DAB988B/*

Parkhurst, J. 2016. *The Politics of Evidence: From evidence-based policy to the good governance of evidence*. Routledge.

Paul, Stéphane, Julien Brunel, L Rioux, Frédérique Vallée, Jaime de Oliveira, Grégory Gailliard, Jean-Louis Gilbert, Timo Wiander, Mohammed El Bakkali, Anthony Faucogney and David Chemouil. 2016. "Recommendations for Security and Safety Co-engineering (Release no 3) - Part A.".
URL: *https://www.researchgate.net/publication/298212533_Recommendations_for_Security_and_Safety_Co-engineering_Release_n3_-_Part_A*

Pawson, R. 2003. "Assessing the quality of evidence in evidence-based policy: when, why, how and when." ESRC Research Methods Programme Conference.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. 1 ed. Morgan Kaufmann.

Perry, A.G., P.A. Potter and W. Ostendorf. 2015. *Nursing Interventions and Clinical Skills*. Elsevier Health Sciences.

Petticrew, M. and H. Roberts. 2003. "Evidence, hierarchies, and typologies - horses for courses." *Journal of Epidemiology and Community Health* 57:527 – 529.
URL: *https://jech.bmj.com/content/57/7/527*

Picca, P. 2018. Use of digital COTS devices in the nuclear industry: successes, challenges and lessons learned. In *SCSC Seminar: COTS, Legacy, and Reuse*.
URL: *https://scsc.uk/file/576/Use-of-digital-COTS-devices-in-the-nuclear-industry---PPicca2.pptx*

Popov, P., A. Povyakalo, V. Stankovic and L. Strigini. 2014. Software Diversity as a Measure for Reducing Development Risk. In *2014 Tenth European Dependable Computing Conference*. pp. 106–117.
URL: *https://ieeexplore.ieee.org/document/6821095/*

Popov, P. and L. Strigini. 2001. The reliability of diverse systems: a contribution using modelling of the fault creation process. In *2001 International Conference on Dependable Systems and Networks*. pp. 5–14.
URL: *https://ieeexplore.ieee.org/document/941385/*

Popov, P., L. Strigini, J. May and S. Kuball. 2003. "Estimating bounds on the reliability of diverse systems." *IEEE Transactions on Software Engineering* 29(4):345–359.
URL: *https://ieeexplore.ieee.org/abstract/document/1191798/*

Popper, K. 2002. *The Logic of Scientific Discovery.* 2 ed. Routledge.

Presthus, W. and B.E. Munkvold. 2016. "How to Frame Your Contribution to Knowledge?".
URL: *https://www.semanticscholar.org/paper/How-to-frame-your-contribution-to-knowledge-A-guide-Presthus-Munkvold/02dedf92548c4194e639690e68688596ec16a1da*

Project Oracle. 2018. "Standards of Evidence." WWW.
URL: *https://project-oracle.com/about-us/validation/*

Puttick, R. 2018. Mapping the Standards of Evidence used in UK social policy. Technical report Alliance for Useful Evidence.
URL: *https://www.nesta.org.uk/documents/768/Mapping_Standards_of_Evidence_A4UE_final.pdf*

Radack, D., H. G. Tiedeman and P. Parkinson. 2019. Civil Certification of Multi-core Processing Systems in Commercial Avionics. In *Engineering Safe Autonomy. Proceedings of the 27th Safety-Critical Systems Symposium, Bristol, UK.*
URL: *https://www.sae.org/publications/technical-papers/content/2019-01-1382/*

Redman, D., D. Ward, J. Chilenski and G. Pollari. 2010. Virtual Integration for Improved System Design. In *AVICPS 2010 Workshop. The First Analytic Virtual Integration of Cyber-Physical Systems Workshop.* pp. 57–64.
URL: *https://www.semanticscholar.org/paper/Virtual-Integration-for-Improved-System-Design-Redman-Ward/d1e647e033c536f304e9e673ce9fff99dc6a5ec7*

Rierson, L. 2017. *Developing Safety-Critical Software: A Practical Guide for Aviation Software and DO-178C Compliance.* CRC Press.

Rittel, H. W. J. and M. M. Webber. 1973. "Dilemmas in a General Theory of Planning." *Policy Sciences* 4:155–169.
URL: *https://link.springer.com/article/10.1007/BF01405730*

Rize, L.S., C. Bergmeir, F. Herrera and J. M. Benitez. 2015. Package 'frbs'. Technical Report 3.1-0 CRAN.
URL: *https://cran.r-project.org/web/packages/frbs/frbs.pdf*

Robinson, P. 2016. Benefits of Mutual Recognition in support of the Typhoon programme. Technical report MAA.
URL: *https://www.eda.europa.eu/docs/default-source/events/2-10-ef-2000-experience-with-recognition---uk-maa.pdf*

Rolfe, G. 2006. "Validity, trustworthiness and rigour: quality and the idea of qualitative research." *JAN* 53:304–310.
URL: *https://www.ncbi.nlm.nih.gov/pubmed/16441535*

Rollenhagen, C. and B. Wahlstrom. 2007. Management systems and safety culture; reflections and suggestions for research. In *2007 IEEE 8th Human Factors and Power Plants and HPRCT 13th Annual Meeting*. pp. 145–148.
URL: *https://www.researchgate.net/publication/4303589_Management_systems_and_safety_culture_reflections_and_suggestions_for_research*

Ross, T.J. 2004. *Fuzzy Logic with Engineering Applications*. John Wiley & Sons.

RSSB. 2017. Guidance on High-Integrity Software-Based Systems for Railway Applications. Technical Report GEGN8650 Rail Safety and Standards Board Limited.
URL: *https://catalogues.rssb.co.uk/rgs/standards/GEGN8650%20Iss%201.pdf*

RTCA. 1992. "DO-178B. Software Considerations in Airborne Systems and Equipment Certification.".

RTCA. 2000. "DO-254. Design Assurance Guidance for Airborne Electronic Hardware.".

RTCA. 2011*a*. "DO-178C. Software Considerations in Airborne Systems and Equipment Certification.".

RTCA. 2011*b*. "DO-248C. Supporting Information for DO-178C and DO-278A.".

RTCA. 2011*c*. "DO-278A. Guidelines for communication, navigation, surveillance and air traffic management (CNS/ATM) systems software integrity assurance.".

RTCA. 2011*d*. "DO-330. Software Tool Qualification Considerations.".

RTCA. 2011*e*. "DO-331. Model-Based Development and Verificaiton Supplement to DO-178C and DO-278A.".

RTCA. 2011*f*. "DO-332. Object-Orientated Technology and Related Techniques Supplement to DO-178C and DO-278A.".

RTCA. 2011*g*. "DO-333. Formal Methods Supplement to DO-178C and DO-278A.".

RTCA. 2014*a*. "DO-326A. Airworthiness Security Process Specification.".

RTCA. 2014*b*. "DO-356. Airworthiness Security Methods and Considerations.".

Rushby, J. 2015. The Interpretation and Evaluation of Assurance Cases. Technical report SRI International.
URL: *https://pdfs.semanticscholar.org/5196/fbfbc98306ad2273fc61d420af8ce1452fe1.pdf*

Rutter, J. 2012. Evidence and Evaluation in Policy Making. Technical report Institute for Government.
URL: *https://www.instituteforgovernment.org.uk/sites/default/files/publications/evidence%20and%20evaluation%20in%20template_final_0.pdf*

Sackett, D. L., W.M.C. Rosenberg, J.A. Muir Gray, R. Brian Haynes and W.S. Richardson. 1996. "Evidence based medicine: what it is and what it isn't." WWW.
URL: *https://www.bmj.com/content/312/7023/71*

SAE. 1996. "Aerospace Recommended Practice 4761. Guidelines and Methods for Conducting the Safety Assessment Process on Civil Airborne Systems and Equipment.".

SAE. 2010. "Aerospace Recommended Practice 4754A. Guidelines for Development of Civil Aircraft and Systems.".

Salkind, N. J. 2010. *Encyclopedia of Research Design*. Vol. 1 SAGE.

Sallum, H. 2015. "Cyber Security Risk Assessment Using Multi Fuzzy Inference System." *International Journal of Engineering and Innovative Technology (IJEIT)* 4(8):0.
URL: *http://www.ijeit.com/Vol%204/Issue%208/IJEIT1412201502_04.pdf*

Sarkar, S., A. Roy and B. S. Purkayastha. 2013. Application of Particle Swarm Optimization in Data Clustering: A Survey. In *International Journal of Computer Applications*. Vol. 65.
URL: *https://www.semanticscholar.org/paper/Application-of-Particle-Swarm-Optimization-in-Data-Sarkar-Roy/7eaee85b0b90864c89fccc3332d8dd9040e29797*

Saunders, M., P. Lewis and A. Thornhill. 2012. *Research methods for business students*. 6 ed. Pearson Education.

Sauro, J. and J. R. Lewis. 2016. *Quantifying the User Experience: Practical Statistics for User Research*. 2 ed. Morgan Kaufmann.

Scapens, Robert W. 1990. "Researching management accounting practice: The role of case study methods." *The British Accounting Review* 22(3):259 – 281.
URL: *http://www.sciencedirect.com/science/article/pii/0890838990900086*

Schaefer, Ina, Rick Rabiser, Dave Clarke, Lorenzo Bettini, David Benavides, Goetz Botterweck, Animesh Pathak, Salvador Trujillo and Karina Villela. 2012. "Software diversity: state of the art and perspectives." *International Journal on Software Tools for Technology Transfer* 14(5):477–495.
URL: *http://www.isa.us.es/sites/default/files/benavides12-ijsttt.pdf*

Schaub, G. and J. Wenzel-Kristoffersen. 2017. In, On, or Out of the Loop? Denmark and Autonomous Weapon Systems. Technical report Centre for Military Studies (University of Copenhagen).
URL: *https://cms.polsci.ku.dk/publikationer/in-on-or-out-of-the-loop/In_On_or_Out_of_the_Loop.pdf*

Scientific American. 2018. "What is 'fuzzy logic'? Are there computers that are inherently fuzzy and do not apply the usual binary logic?" WWW.
URL: *https://www.scientificamerican.com/article/what-is-fuzzy-logic-are-t/*

SEBoK. 2018*a*. "Roles and Competencies - SEBoK.".
URL: *https://www.sebokwiki.org/wiki/Roles_and_Competencies*

SEBoK. 2018*b*. "Why Model? - SEBoK.".
URL: *https://www.sebokwiki.org/wiki/Why_Model%3F*

SEI. 2019. "Software Solutions Division: About." WWW.
URL: *https://www.sei.cmu.edu/about/divisions/software-solutions-division/*

Sense about Science. 2017. "Transparency of evidence: a spot check of government policy proposals July 2016 to July 2017." WWW.
URL: *http://senseaboutscience.org/activities/transparency-evidence-spot-check/*

Sense about Science. 2018. Transparency of evidence: a spot check of government policy proposals July 2016 to July 2017. Technical report Sense about Science.
URL: *http://senseaboutscience.org/wp-content/uploads/2018/01/Transparency-of-evidence-spotcheck.pdf*

Smith, P. 1995. "On the Unintended Consequences of Publishing Performance Data in the Public Sector." International Journal of Public Administration. 18 (2/3) 277310.
URL: *https://www.tandfonline.com/doi/abs/10.1080/01900699508525011*

Sowa, J. F. 1999. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Course Technology.

Spriggs, J. 2018. COTS Assurance - An Overview. In *SCSC Seminar: COTS, Legacy, and Reuse.*
URL: *https://scsc.uk/file/576/02---John-Spriggs---COTS-Assurance-web.pdf*

Spriggs, J. 2019. Sufficient Assurance? In *Engineering Safe Autonomy. Proceedings of the 27th Safety-Critical Systems Symposium, Bristol, UK.*
URL: *https://scsc.uk/rp150.4:1*

Standish, M. and M. J. Hadley. 2014. [Airborne Platform] Software Build: Review of Software Processes and Product Service History. Technical report Dstl.

Standish, M. and M. J. Hadley. 2018. [Airborne Platform] Safety-Related Programmable Element (PE) Qualification Strategy. Technical report Dstl.

Standish, M., M. J. Hadley and E. Lennon. 2017. An Argument for the Adoption of Diverse Software and Complex Electronic Hardware (CEH) Evidence Within a Qualification Strategy. Technical report Dstl.

Stanford Encyclopedia of Philosphy. 2014. "Evidence." WWW.
URL: *https://plato.stanford.edu/entries/evidence/*

Steinzor, R. 2010. "Lessons from the North Sea: Should Safety Cases Come to America.".
URL: *https://www.researchgate.net/publication/48865783_Lessons_from_the_North_Sea_Should_Safety_Cases_Come_to_America*

Stevens, B., R. Ashmore, A. Margheri and V. Sassone. 2019. Developing Critical Software in the Modern Threat Environment. In *Engineering Safe Autonomy. Proceedings of the 27th Safety-Critical Systems Symposium, Bristol, UK.*
URL: *https://scsc.uk/rp150.14:1*

Suresh, Y. 2015. Software quality assurance for object-oriented systems using meta-heuristic search techniques. In *2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT).* pp. 441–448.
URL: *https://ieeexplore.ieee.org/document/7456924*

TechTarget. 2006. "Integer Overflow." WWW.
URL: *https://searchsoftwarequality.techtarget.com/definition/integer-overflow*

TechTarget. 2015. "Race Condition." WWW.
URL: *https://searchstorage.techtarget.com/definition/race-condition*

The Guardian. 2013. "Piper Alpha disaster: how 167 oil rig workers died." WWW.
URL: *https://www.theguardian.com/business/2013/jul/04/piper-alpha-disaster-167-oil-rig*

Tian, P., J. Wang, W. Zhang and J. Liu. 2009. A Fault Tree Analysis Based Software System Reliability Allocation Using Genetic Algorithm Optimization. In *2009 WRI World Congress on Software Engineering*. Vol. 2 pp. 194–198.
URL: *https://ieeexplore.ieee.org/document/5319681*

Toulmin, S.E. 2003. *The Uses of Argument*. Cambridge University Press.

Turban, E., R. Sharda and D. Delen. 2010. *Decision Support and Business Intelligence Systems*. 9th ed. Upper Saddle River, NJ, USA: Prentice Hall Press.

UK MOD. 2001. "JSP 440. The Defence Manual of Security.".

UK MOD. 2008. "Defence Standard 00-250: Human Factors for Designers of Systems. Issue 1.".

UK MOD. 2012. Applied R&M Manual for Defence Systems. Part C - R&M Related Techniques. Chapter 51 - Software Reliability Techniques. Technical Report GR-77 UK MOD.

UK MOD. 2014*a*. "Design and Airworthiness Requirements for Service Aircraft. Defence Standard 00-970 Part 13 (Military Common Fit Equipment), Issue 8.".

UK MOD. 2014*b*. "Requirements for Safety of Programmable Elements (PE) in Defence Systems. Defence Standard 00-55 Part 1 (Requirements and Guidance), Issue 3.".

UK MOD. 2014*c*. "Safety Management Requirements for Defence Systems. Defence Standard 00-56, Part 1 (Requirements and Guidance), Issue 5.".

UK MOD. 2015. Defence Industrial Strategy – White Paper. Technical report UK MOD.

UK MOD. 2018. "An Introduction to System Safety Management in the MOD. Issue 4." WWW.
URL: *https://www.asems.mod.uk/sites/default/files/documents/White%20and%20Green%20Book/SSM%20Whitebook%20PART%201%20v5.pdf*

Utterly, M.R.H. and B. Wilkinson. 2016. *A spin of the wheel? Defence procurement and defence industries in the Brexit debates*. John Wiley & Sons Ltd chapter 3, pp. 569–586.
URL: *https://www.chathamhouse.org/sites/default/files/publications/ia/inta92-3-04-uttleywilkinson.pdf*

Wang, R., J. Guiochet and G. Motet. 2016. A Framework for Assessing Safety Argumentation Confidence. In *Software Engineering for Resilient Systems: 8th International Workshop, SERENE 2016, Gothenburg, Sweden, September 5-6, 2016, Proceedings*. p. 149.
URL: *https://link.springer.com/chapter/10.1007/978-3-319-45892-2_1*

Wang, Rui, Jérémie Guiochet and Gilles Motet. 2017. Confidence Assessment Framework for Safety Arguments. In *Computer Safety, Reliability, and Security: 36th International Conference, SAFECOMP 2017*. pp. 55–68.
URL: *https://www.researchgate.net/publication/319138629_Confidence_Assessment_Framework_for_Safety_Arguments*

Wang, X., X. Z. Gao and K. Zenger. 2015. *An Introduction to Harmony Search Optimization*. Spinger.

Waring, A. 1996. *Practical Systems Thinking*. Cengage Learning.

Watkins, C. B. and R. Walter. 2007. Transitioning from federated avionics architectures to Integrated Modular Avionics. In *2007 IEEE/AIAA 26th Digital Avionics Systems Conference*. pp. 2.A.1–1–2.A.1–10.
URL: *https://ieeexplore.ieee.org/document/4391842*

Weaver, R., T. Kelly and P. Mayo. 2006. Gaining Confidence in Goal-based Safety Cases. In *SSS*.
URL: *https://www-users.cs.york.ac.uk/tpk/sss06.pdf*

Weaver, RA, G Despotou, TP Kelly and JA McDermid. 2005. "Combining Software Evidence - Arguments and Assurance." *REBSE '05* 0:7.
URL: *https://www-users.cs.york.ac.uk/tpk/REBSE05.pdf*

Weaver, Rob, Jane Fenn and Tim Kelly. 2003. A Pragmatic Approach to Reasoning About the Assurance of Safety Arguments. In *Proceedings of the 8th Australian Workshop on Safety Critical Systems and Software - Volume 33*. SCS '03 Darlinghurst, Australia: Australian Computer Society, Inc. pp. 57–67.
URL: *http://crpit.com/confpapers/CRPITV33Weaver.pdf*

Weinstock, D. 2007. "What Is Evidence?: A Philosophical Perspective." Presentation - National Collaborating Centres for Public Health (2007 Summer Institute: Making Sense of it All).
URL: *http://www.ncchpp.ca/docs/Weinstock_Evidence_Ang.pdf*

Weisstein, E. W. 2018*a*. "Global Minimum." WWW.
URL: *http://mathworld.wolfram.com/GlobalMinimum.html*

Weisstein, E. W. 2018*b*. "Local Minimum." WWW.
URL: *http://mathworld.wolfram.com/LocalMinimum.html*

Whittaker, J. 2009. *Exploratory Software Testing*. Pearson Education.

Wilson, B. 1990. *Systems: Concepts, Methodologies, and Applications*. 2 ed. Wiley.

World Nuclear Association. 2018. Defence-in-Depth and Diversity: Challenges Related to I&C Architecture. Technical report World Nuclear Association.
URL: *http://www.world-nuclear.org/getattachment/Our-Association/Publications/Online-Reports/CORDEL-Defence-in-Depth-and-Diversity/CORDEL-Defence-in-Depth-Report-10-April.pdf.aspx*

Wright, D. and K.Y. Cai. 1994. Representing Uncertainty for Safety Critical Systems. Technical report City University, London.
URL: *http://research.cs.ncl.ac.uk/cabernet/www.laas.research.ec.org/pdcs/trs/abstracts/135.html*

Xiao-ping, Z., H. Shi-zhao and D. Xin-wei. 2008. Comparison of Performance between Genetic Algorithm and Breeding Algorithm for Global Optimization of Continuous Functions. In *2008 Fourth International Conference on Natural Computation*. Vol. 1 pp. 294–298.
URL: *https://www.researchgate.net/publication/251863720_Comparison_of_Performance_between_Genetic_Algorithm_and_Breeding_Algorithm_for_Global_Optimization_of_Continuous_Functions*

Yamamoto, S. 2015. Assuring Security through Attribute GSN. In *2015 5th International Conference on IT Convergence and Security (ICITCS)*. pp. 1–5.
URL: *https://www.computer.org/csdl/proceedings-article/icitcs/2015/07292954/12OmNvAiSKw*

Yang, X. S. 2018. *Optimization Techniques and Applications with Examples*. Wiley.

Yearworth, M. 2014*a*. "Introduction to Hierarchical Process Modelling." WWW.
URL: *http://www.smartsteep.eu/wp-content/uploads/2014/06/Chapter3-MY-290114.pdf*

Yearworth, M. 2014*b*. "More on Assessing Performance." WWW.
URL: *http://www.smartsteep.eu/wp-content/uploads/2014/06/Chapter5-MY-290114.ppt.pdf*

Yearworth, Mike, David A Lowe, Daniel Schien and Thomas A Walworth. 2015. From deciding to acting: hierarchical process modelling for problem structuring. In *Calculating and Communicating Uncertainty (CCU 2015)*.
URL: *https://research-information.bris.ac.uk/en/publications/from-deciding-to-acting-hierarchical-process-modelling-for-problem-structuring(43b50dde-2ffa-452c-95b5-446d1c132be2).html*

Yin, R. K. 2003. *Case Study Research: Design and Methods*. SAGE.

Young, K. S., J. T. Wood, G. M. Phillips and D. J. Pedersen. 2006. *Group Discussion: A Practical Guide to Participation and Leadership*. 4 ed. Waveland Press.

Yuan, Chunchun, Ji Wu, Chao Liu and H Yang. 2017. A Subjective Logic-Based Approach for Assessing Confidence in Assurance Case. In *International Journal of Performability Engineering*.
URL: *https://www.researchgate.net/publication/320709440_A_Subjective_Logic-Based_Approach_for_Assessing_Confidence_in_Assurance_Case*

Yuan, T. and T. Kelly. 2011. Argument Schemes in Computer System Safety Engineering. In *Informal Logic*. Vol. 31.
URL: *https://www-users.cs.york.ac.uk/~tommy/Papers/IL-2011.pdf*

Yue, M., B. Guo, T. Hu and X. Guo. 2009. The research of parameters of genetic algorithm and comparison with particle swarm optimization and shuffled frog-leaping algorithm. In *2009 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS)*. Vol. 1 pp. 77–80.
URL: *https://ieeexplore.ieee.org/document/5406960*

Zadeh, L. A. 1965. "Fuzzy sets." *Information and Control* 8(3):338–353.
URL: *http://www.sciencedirect.com/science/article/pii/S001999586590241X*

Zadeh, L. A. 1973. "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes." *Systems, Man and Cybernetics, IEEE Transactions on* SMC-3(1):28–44.
URL: *https://ieeexplore.ieee.org/abstract/document/5408575*

Zeng, F., M. Lu and D. Zhong. 2013. "Using DS Evidence Theory to Evaluation of Confidence in Safety Case." *Journal of Theoretical and Applied Information Technology* 47:0.
URL: *http://www.jatit.org/volumes/Vol47No1/22Vol47No1.pdf*

Zhao, Xingyu, Dajian Zhang, Minyan Lu and Fuping Zeng. 2012. "A New Approach to Assessment of Confidence in Assurance Cases.".
URL: *https://www.researchgate.net/publication/262314006_A_New_Approach_to_Assessment_of_Confidence_in_Assurance_Cases*

Zhenzhen, M. A., P. Kumaraswamy, Z. Jianjun and Z. Shitao. 2018. "Dynamic hesitant fuzzy linguistic group decision-making from a reliability perspective." *Journal of Systems Engineering and Electronics* 29(5):1009–1021.
URL: *http://www.jseepub.com/EN/abstract/abstract6531.shtml*

Zhu, Xiaodong, Zhiqiu Huang, Shuqun Yang and Guohua Shen. 2007. Fuzzy Implication Methods in Fuzzy Logic. In *Proceedings - Fourth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2007*. Vol. 1 pp. 154–158.
URL: *https://dl.acm.org/doi/10.1109/FSKD.2007.327*

Zimmermann, H.-J. 2000. "An application-oriented view of modeling uncertainty." *European Journal of Operational Research* 122(2):190 – 198.
URL: *http://www.sciencedirect.com/science/article/pii/S0377221799002283*

Zolotas, A., D. Kolovas, R. F. Paige, J. McDermid, M. Bennett, S. Hutchesson and A. Hawthorn. 2017. "SECT-AIR. Software Engineering Costs and Timescales - Aerospace Initiative for Reduction." WWW.
URL: *https://www.slideshare.net/astalavistathes/the-sectair-project-staf-2017*

[This page intentionally left blank]

Appendix A

Example of Workshop Discussion Items (MAA)

An example of a workshop designed and delivered for the research is that with the MAA. The workshop explored the use of evidence and how it can be assessed given disparate sources of information. The topics/questions under discussion are below.

1. *The focus.* Outline of the workshop focus and the aims to gain information on the use of evidence by the MAA. Also, to gain thoughts on some initial concepts of the research.

2. *What the law states.* Overview of how the legal domain conceptualises evidence. Use of rules to determine *what* evidence is considered and the *weight* of evidence. Also, the law is concerned with the *quantum (amount)*, *quality*, and *type of proof* needed. The types of evidence were also explored.

3. *The court of law.* What can be learnt from how evidence is presented in court (structure of evidence). The use of a persuasive and logically complete narrative.

4. *Undermining evidence.* Discussion on how evidence can undermine an argument. What happens if a 'pillar' of the software safety argument is 'disproved' or is not robust as first thought?

5. *Other domains.* Discussion on how evidence is treated within other domains, for example within medical research.

6. *Hierarchies of evidence.* Discussion on the structure of evidence and how to determine the importance of evidential types. Is there a clear hierarchy of evidence types?

7. *Air Traffic Controller (ATC) radar example.* Discussion on the evidence for an ATC radar. What software safety assurance evidence would be valid?

8. *Types of evidence.* Discussion on the types of evidence which could support a generic software safety assurance argument.

9. *Weight of evidence.* Discussion on the weighting of evidence. How is it balanced? Does one evidential type gain confidence in the event of a shortfall of another type?

10. *Establishing a stopping point.* Discussion on how much evidence is enough for a software safety assurance argument.

11. *EBPM example.* Discussion on how the EBPM domain treats evidence. Factors include: importance of the decision; expertise of the decision-maker; and the openness of the decision-maker to evidence.

12. *The EngD.* How the MAA can inform the EngD research.

13. *Framework.* Discussion on the use and role of a DSF to assist DTs, for example, with making informed judgements. Could a tool inform DTs *before* they contracted for assurance evidence? Extent of engineering judgement to determine diverse evidence? What evidence would inform a tool? Does not gaining certain types of evidence matter? Is there a weighting to evidence?

Appendix B

Semi-Structured Interview Details

A number of semi-structured interviews were conducted to gather information on the use of software safety assurance evidence within other domains. The interview style was based upon the guidance within Saunders, Lewis and Thornhill (2012), e.g. neutral questioning. The main question types were *open*[1] and *probing*[2]. The topics/questions under discussion are below.

1. *The focus.* Outline of the focus of the semi-structured interview and why the information from the particular domain can help.

2. *The EngD.* Outline of the EngD as a concept and the particular focus of the research.

3. *Treatment of evidence: the regulations.* Discussion on the regulations of the domain and how software safety assurance evidence is gathered.

4. *Treatment of evidence: the reality.* Discussion on the reality of gathering software safety assurance evidence. How do the regulations and the actual arguments differ?

5. *Type of evidence.* Discussion on the types of evidence which support a software safety assurance approach.

6. *Evidence structure.* Discussion on how evidence is presented to stakeholders and how this informs decision making.

7. *Stakeholder structure.* Discussion on the stakeholders/organisations involved in the regulatory process and the roles/remit.

[1] *Open* questions allow participants to define and describe situations/events (Saunders, Lewis and Thornhill, 2012).

[2] *Probing* questions allow significant responses to be explored (Saunders, Lewis and Thornhill, 2012).

8. *Tools to assist.* Discussion on the tools/techniques which could assist SMEs to structure/judge evidence.

9. *Next steps.* Discussion on how the participant may inform the next stages of the research and if there are any sources of information, e.g. third-party research papers, which may inform the participants interest in the assurance domain.

The information from the interviews was recorded in a number of non-published outputs. These outputs are stated below in no particular order.

1. Meeting with Civil Aviation Authority (CAA). 2018.
2. Meeting with Dstl Land Platforms Systems Team Member. 2018.
3. Meeting with Dstl Software and Systems Dependability Team Member. 2018.
4. Teleconference with National Air Traffic Services (NATS). 2015.
5. Teleconference with Consultants to Government and Industries (CGI). 2015.
6. Teleconference with Lloyds Register (Rail). 2015.
7. Meeting with Dstl and Military Aviation Authority (MAA). 2015.
8. Meeting with MAA. 2015.
9. Meeting with Vehicle Certification Agency (VCA). 2015.
10. Teleconference with VCA. 2015.
11. Teleconference with UK MOD Naval Authority Group (NAG). 2015.
12. Meeting with Office for Nuclear Regulation (ONR). 2015.

Appendix C

Sources to Inform Evidence Within the Initial Framework

The evidence to be initially included within the framework is stated below. The evidence has been derived (and is supported) from a number of sources and domains (e.g. automotive) which were subject to discussion within Chapter 6.

[Appendix text redacted]

[This page intentionally left blank]